

Table of Contents

Overview

[Network security](#)

[Database security](#)

[Storage security](#)

[Compute security](#)

[Operational security](#)

[Security management and monitoring](#)

[Service Fabric security](#)

[Identity management](#)

[IoT security](#)

[Azure encryption overview](#)

[Security architecture](#)

[Enabling operational security](#)

[Advanced threat detection](#)

[Logging and auditing](#)

[Isolation in the public cloud](#)

[Security technical capabilities](#)

[Governance in Azure](#)

[Data encryption at rest](#)

Get Started

[Getting started with Azure security](#)

[Security best practices and patterns](#)

[Security services and technologies](#)

[Network security](#)

[Network security best practices](#)

[Azure network security](#)

[Boundary security](#)

[Secure hybrid network architecture](#)

[Storage security](#)

- [Data security and encryption best practices](#)
- [Storage security guide](#)
- [Compute security](#)
 - [Best practices for Azure VMs](#)
 - [Best practices for IaaS workloads](#)
 - [Microsoft Antimalware](#)
 - [Disk encryption for IaaS VMs](#)
 - [Encrypt an Azure VM](#)
- [Operational security](#)
 - [Best practices for operational security](#)
 - [Security management and monitoring](#)
 - [Security management](#)
 - [Azure Security Center](#)
 - [Introduction to Azure log integration](#)
- [Service Fabric](#)
 - [Service Fabric best practices](#)
 - [Service Fabric checklist](#)
- [Identity management](#)
 - [Identity management security best practices](#)
- [PaaS services](#)
 - [Securing PaaS deployments](#)
- [Internet of Things](#)
 - [Secure your IoT deployment](#)
 - [IoT security best practices](#)
- [Security architecture](#)
 - [Data classification](#)
 - [Disaster recovery and high availability for applications built on Azure](#)
- [Related](#)
 - [Trust Center](#)
 - [Microsoft Security Response Center](#)
 - [Pen testing](#)
 - [Security Center](#)

- [Key Vault](#)
- [Log Analytics](#)
- [Multi-Factor Authentication](#)
- [Azure Active Directory](#)
- [Operations Management Suite](#)
- [Resources](#)
 - [Azure Roadmap](#)
 - [Azure security MVP program](#)
 - [Cybersecurity consulting](#)
 - [Pricing calculator](#)
- [Scenarios](#)
 - [Manage personal data in Azure](#)
 - [Discover, identify, and classify personal data in Azure](#)
 - [Protect personal data in Azure](#)
 - [Protect personal data with Security Center](#)
 - [Protect personal data with Application Gateway](#)
 - [Protect personal data by using identity and access controls](#)
 - [Protect personal data at rest by using encryption](#)
 - [Protect personal data in transit by using encryption](#)
 - [Protect personal data by using Azure reporting tools](#)
- [Security and Compliance blog](#)
- [Security courses from Virtual Academy](#)
- [Security videos on Channel 9](#)
- [Threat Modeling Tool](#)
 - [Getting started](#)
 - [Feature overview](#)
 - [Threats](#)
 - [Mitigations](#)

Introduction to Azure Security

9/8/2017 • 30 min to read • [Edit Online](#)

Overview

We know that security is job one in the cloud and how important it is that you find accurate and timely information about Azure security. One of the best reasons to use Azure for your applications and services is to take advantage of its wide array of security tools and capabilities. These tools and capabilities help make it possible to create secure solutions on the secure Azure platform. Microsoft Azure provides confidentiality, integrity, and availability of customer data, while also enabling transparent accountability.

To help you better understand the collection of security controls implemented within Microsoft Azure from both the customer's and Microsoft operations' perspectives, this white paper, "Introduction to Azure Security", is written to provide a comprehensive look at the security available with Microsoft Azure.

Azure Platform

Azure is a public cloud service platform that supports a broad selection of operating systems, programming languages, frameworks, tools, databases, and devices. It can run Linux containers with Docker integration; build apps with JavaScript, Python, .NET, PHP, Java, and Node.js; build back-ends for iOS, Android, and Windows devices.

Azure public cloud services support the same technologies millions of developers and IT professionals already rely on and trust. When you build on, or migrate IT assets to, a public cloud service provider you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security requirements.

In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your organization's deployments. This document helps you understand how Azure security capabilities can help you fulfill these requirements.

NOTE

The primary focus of this document is on customer-facing controls that you can use to customize and increase security for your applications and services.

We do provide some overview information, but for detailed information on how Microsoft secures the Azure platform itself, see information provided in the [Microsoft Trust Center](#).

Abstract

Initially, public cloud migrations were driven by cost savings and agility to innovate. Security was considered a major concern for some time, and even a show stopper, for public cloud migration. However, public cloud security has transitioned from a major concern to one of the drivers for cloud migration. The rationale behind this is the superior ability of large public cloud service providers to protect applications and the data of cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security needs. In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your deployments to meet your IT control policies and adhere to external regulations.

This paper outlines Microsoft's approach to security within the Microsoft Azure cloud platform:

- Security features implemented by Microsoft to secure the Azure infrastructure, customer data, and applications.
- Azure services and security features available to you to manage the Security of the Services and your data within your Azure subscriptions.

Summary Azure Security Capabilities

The table following provide a brief description of the security features implemented by Microsoft to secure the Azure infrastructure, customer data, and secure applications.

Security Features Implemented to Secure the Azure Platform:

The features listed following are capabilities you can review to provide the assurance that the Azure Platform is managed in a secure manner. Links have been provided for further drill-down on how Microsoft addresses customer trust questions in four areas: Secure Platform, Privacy & Controls, Compliance, and Transparency.

SECURE PLATFORM	PRIVACY & CONTROLS	COMPLIANCE	TRANSPARENCY
Security Development Cycle, Internal audits	Manage your data all the time	Trust Center	How Microsoft secures customer data in Azure services
Mandatory Security training, back ground checks	Control on data location	Common Controls Hub	How Microsoft manage data location in Azure services
Penetration testing, intrusion detection, DDoS, Audits & logging	Provide data access on your terms	The Cloud Services Due Diligence Checklist	Who in Microsoft can access your data on what terms
State of art datacentre, physical security, Secure Network	Responding to law enforcement	Compliance by service, location & Industry	How Microsoft secures customer data in Azure services
Security Incident response, Shared Responsibility	Stringent privacy standards		Review certification for Azure services, Transparency hub

Security Features Offered by Azure to Secure Data and Application

Depending on the cloud service model, there is variable responsibility for who is responsible for managing the security of the application or service. There are capabilities available in the Azure Platform to assist you in meeting these responsibilities through built-in features, and through partner solutions that can be deployed into an Azure subscription.

The built-in capabilities are organized in six (6) functional areas: Operations, Applications, Storage, Networking, Compute, and Identity. Additional detail on the features and capabilities available in the Azure Platform in these six (6) areas are provided through summary information.

Operations

This section provides additional information regarding key features in security operations and summary information about these capabilities.

Operations Management Suite Security and Audit Dashboard

The [OMS Security and Audit solution](#) provides a comprehensive view into your organization's IT security posture with [built-in search queries](#) for notable issues that require your attention. The [Security and Audit](#) dashboard is the home screen for everything related to security in OMS. It provides high-level insight into the Security state of your

computers. It also includes the ability to view all events from the past 24 hours, 7 days, or any other custom time frame.

In addition, you can configure OMS Security & Compliance to [automatically carry out specific actions](#) when a specific event is detected.

Azure Resource Manager

[Azure Resource Manager](#) enables you to work with the resources in your solution as a group. You can deploy, update, or delete all the resources for your solution in a single, coordinated operation. You use an [Azure Resource Manager template](#) for deployment and that template can work for different environments such as testing, staging, and production. Resource Manager provides security, auditing, and tagging features to help you manage your resources after deployment.

Azure Resource Manager template-based deployments help improve the security of solutions deployed in Azure because standard security control settings and can be integrated into standardized template-based deployments. This reduces the risk of security configuration errors that might take place during manual deployments.

Application Insights

[Application Insights](#) is an extensible Application Performance Management (APM) service for web developers. With Application Insights, you can monitor your live web applications and automatically detect performance anomalies. It includes powerful analytics tools to help you diagnose issues and to understand what users actually do with your apps. It monitors your application all the time it's running, both during testing and after you've published or deployed it.

Application Insights creates charts and tables that show you, for example, what times of day you get most users, how responsive the app is, and how well it is served by any external services that it depends on.

If there are crashes, failures or performance issues, you can search through the telemetry data in detail to diagnose the cause. And the service sends you emails if there are any changes in the availability and performance of your app. Application Insight thus becomes a valuable security tool because it helps with the availability in the confidentiality, integrity, and availability security triad.

Azure Monitor

[Azure Monitor](#) offers visualization, query, routing, alerting, auto scale, and automation on data both from the Azure infrastructure ([Activity Log](#)) and each individual Azure resource ([Diagnostic Logs](#)). You can use Azure Monitor to alert you on security-related events that are generated in Azure logs.

Log Analytics

[Log Analytics](#) part of [Operations Management Suite](#) – Provides an IT management solution for both on-premises and third-party cloud-based infrastructure (such as AWS) in addition to Azure resources. Data from Azure Monitor can be routed directly to Log Analytics so you can see metrics and logs for your entire environment in one place.

Log Analytics can be a useful tool in forensic and other security analysis, as the tool enables you to quickly search through large amounts of security-related entries with a flexible query approach. In addition, on-premises [firewall and proxy logs can be exported into Azure and made available for analysis using Log Analytics](#).

Azure Advisor

[Azure Advisor](#) is a personalized cloud consultant that helps you to optimize your Azure deployments. It analyzes your resource configuration and usage telemetry. It then recommends solutions to help improve the [performance](#), [security](#), and [high availability](#) of your resources while looking for opportunities to [reduce your overall Azure spend](#). Azure Advisor provides security recommendations, which can significantly improve your overall security posture for solutions you deploy in Azure. These recommendations are drawn from security analysis performed by [Azure Security Center](#).

Azure Security Center

[Azure Security Center](#) helps you prevent, detect, and respond to threats with increased visibility into and control

over the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

In addition, Azure Security Center helps with security operations by providing you a single dashboard that surfaces alerts and recommendations that can be acted upon immediately. Often, you can remediate issues with a single click within the Azure Security Center console.

Applications

The section provides additional information regarding key features in application security and summary information about these capabilities.

Web Application vulnerability scanning

One of the easiest ways to get started with testing for vulnerabilities on your [App Service app](#) is to use the [integration with Tinfoil Security](#) to perform one-click vulnerability scanning on your app. You can view the test results in an easy-to-understand report, and learn how to fix each vulnerability with step-by-step instructions.

Penetration Testing

If you prefer to perform your own penetration tests or want to use another scanner suite or provider, you must follow the [Azure penetration testing approval process](#) and obtain prior approval to perform the desired penetration tests.

Web Application firewall

The web application firewall (WAF) in [Azure Application Gateway](#) helps protect web applications from common web-based attacks like SQL injection, cross-site scripting attacks, and session hijacking. It comes preconfigured with protection from threats identified by the [Open Web Application Security Project \(OWASP\)](#) as the top 10 common vulnerabilities.

Authentication and authorization in Azure App Service

[App Service Authentication / Authorization](#) is a feature that provides a way for your application to sign in users so that you don't have to change code on the app backend. It provides an easy way to protect your application and work with per-user data.

Layered Security Architecture

Since [App Service Environments](#) provide an isolated runtime environment deployed into an [Azure Virtual Network](#), developers can create a layered security architecture providing differing levels of network access for each application tier. A common desire is to hide API back-ends from general Internet access, and only allow APIs to be called by upstream web apps. [Network Security groups \(NSGs\)](#) can be used on Azure Virtual Network subnets containing App Service Environments to restrict public access to API applications.

Web server diagnostics and application diagnostics

App Service web apps provide diagnostic functionality for logging information from both the web server and the web application. These are logically separated into [web server diagnostics](#) and [application diagnostics](#). Web server includes two major advances in diagnosing and troubleshooting sites and applications.

The first new feature is real-time state information about application pools, worker processes, sites, application domains, and running requests. The second new advantages are the detailed trace events that track a request throughout the complete request-and-response process.

To enable the collection of these trace events, IIS 7 can be configured to automatically capture full trace logs, in XML format, for any particular request based on elapsed time or error response codes.

Web server diagnostics

You can enable or disable the following kinds of logs:

- Detailed Error Logging - Detailed error information for HTTP status codes that indicate a failure (status code 400 or greater). This may contain information that can help determine why the server returned the error code.
- Failed Request Tracing - Detailed information on failed requests, including a trace of the IIS components used to process the request and the time taken in each component. This can be useful if you are attempting to increase site performance or isolate what is causing a specific HTTP error to be returned.
- Web Server Logging - Information about HTTP transactions using the W3C extended log file format. This is useful when determining overall site metrics such as the number of requests handled or how many requests are from a specific IP address.

Application diagnostics

[Application diagnostics](#) allows you to capture information produced by a web application. ASP.NET applications can use the [System.Diagnostics.Trace](#) class to log information to the application diagnostics log. In Application Diagnostics, there are two major types of events, those related to application performance and those related to application failures and errors. The failures and errors can be divided further into connectivity, security, and failure issues. Failure issues are typically related to a problem with the application code.

In Application Diagnostics, you can view events grouped in these ways:

- All (displays all events)
- Application Errors (displays exception events)
- Performance (displays performance events)

Storage

The section provides additional information regarding key features in Azure storage security and summary information about these capabilities.

Role-Based Access Control (RBAC)

You can secure your storage account with Role-Based Access Control (RBAC). Restricting access based on the [need to know](#) and [least privilege](#) security principles is imperative for organizations that want to enforce Security policies for data access. These access rights are granted by assigning the appropriate RBAC role to groups and applications at a certain scope. You can use [built-in RBAC roles](#), such as Storage Account Contributor, to assign privileges to users. Access to the storage keys for a storage account using the [Azure Resource Manager](#) model can be controlled through Role-Based Access Control (RBAC).

Shared Access Signature

A [shared access signature \(SAS\)](#) provides delegated access to resources in your storage account. The SAS means that you can grant a client limited permissions to objects in your storage account for a specified period and with a specified set of permissions. You can grant these limited permissions without having to share your account access keys.

Encryption in Transit

Encryption in transit is a mechanism of protecting data when it is transmitted across networks. With Azure Storage, you can secure data using:

- [Transport-level encryption](#), such as HTTPS when you transfer data into or out of Azure Storage.
- [Wire encryption](#), such as [SMB 3.0 encryption](#) for [Azure File shares](#).
- Client-side encryption, to encrypt the data before it is transferred into storage and to decrypt the data after it is transferred out of storage.

Encryption at rest

For many organizations, data encryption at rest is a mandatory step towards data privacy, compliance, and data sovereignty. There are three Azure storage security features that provide encryption of data that is "at rest":

- [Storage Service Encryption](#) allows you to request that the storage service automatically encrypt data when writing it to Azure Storage.
- [Client-side Encryption](#) also provides the feature of encryption at rest.
- [Azure Disk Encryption](#) allows you to encrypt the OS disks and data disks used by an IaaS virtual machine.

Storage Analytics

[Azure Storage Analytics](#) performs logging and provides metrics data for a storage account. You can use this data to trace requests, analyze usage trends, and diagnose issues with your storage account. Storage Analytics logs detailed information about successful and failed requests to a storage service. This information can be used to monitor individual requests and to diagnose issues with a storage service. Requests are logged on a best-effort basis. The following types of authenticated requests are logged:

- Successful requests.
- Failed requests, including timeout, throttling, network, authorization, and other errors.
- Requests using a Shared Access Signature (SAS), including failed and successful requests.
- Requests to analytics data.

Enabling Browser-Based Clients Using CORS

[Cross-Origin Resource Sharing \(CORS\)](#) is a mechanism that allows domains to give each other permission for accessing each other's resources. The User Agent sends extra headers to ensure that the JavaScript code loaded from a certain domain is allowed to access resources located at another domain. The latter domain then replies with extra headers allowing or denying the original domain access to its resources.

Azure storage services now support CORS so that once you set the CORS rules for the service, a properly authenticated request made against the service from a different domain is evaluated to determine whether it is allowed according to the rules you have specified.

Networking

The section provides additional information regarding key features in Azure network security and summary information about these capabilities.

Network Layer Controls

Network access control is the act of limiting connectivity to and from specific devices or subnets and represents the core of network security. The goal of network access control is to make sure that your virtual machines and services are accessible to only users and devices to which you want them accessible.

Network Security Groups

A [Network Security Group \(NSG\)](#) is a basic stateful packet filtering firewall and it enables you to control access based on a [5-tuple](#). NSGs do not provide application layer inspection or authenticated access controls. They can be used to control traffic moving between subnets within an Azure Virtual Network and traffic between an Azure Virtual Network and the Internet.

Route Control and Forced Tunneling

The ability to control routing behavior on your Azure Virtual Networks is a critical network security and access control capability. For example, if you want to make sure that all traffic to and from your Azure Virtual Network goes through that virtual security appliance, you need to be able to control and customize routing behavior. You can do this by configuring User-Defined Routes in Azure.

[User-Defined Routes](#) allow you to customize inbound and outbound paths for traffic moving into and out of

individual virtual machines or subnets to insure the most secure route possible. [Forced tunneling](#) is a mechanism you can use to ensure that your services are not allowed to initiate a connection to devices on the Internet.

This is different from being able to accept incoming connections and then responding to them. Front-end web servers need to respond to requests from Internet hosts, and so Internet-sourced traffic is allowed inbound to these web servers and the web servers can respond.

Forced tunneling is commonly used to force outbound traffic to the Internet to go through on-premises security proxies and firewalls.

Virtual Network Security Appliances

While Network Security Groups, User-Defined Routes, and forced tunneling provide you a level of security at the network and transport layers of the [OSI model](#), there may be times when you want to enable security at higher levels of the stack. You can access these enhanced network security features by using an Azure partner network security appliance solution. You can find the most current Azure partner network security solutions by visiting the [Azure Marketplace](#) and searching for "security" and "network security."

Azure Virtual Network

An Azure virtual network (VNet) is a representation of your own network in the cloud. It is a logical isolation of the Azure network fabric dedicated to your subscription. You can fully control the IP address blocks, DNS settings, security policies, and route tables within this network. You can segment your VNet into subnets and place Azure IaaS virtual machines (VMs) and/or [Cloud services \(PaaS role instances\)](#) on Azure Virtual Networks.

Additionally, you can connect the virtual network to your on-premises network using one of the [connectivity options](#) available in Azure. In essence, you can expand your network to Azure, with complete control on IP address blocks with the benefit of enterprise scale Azure provides.

Azure networking supports various secure remote access scenarios. Some of these include:

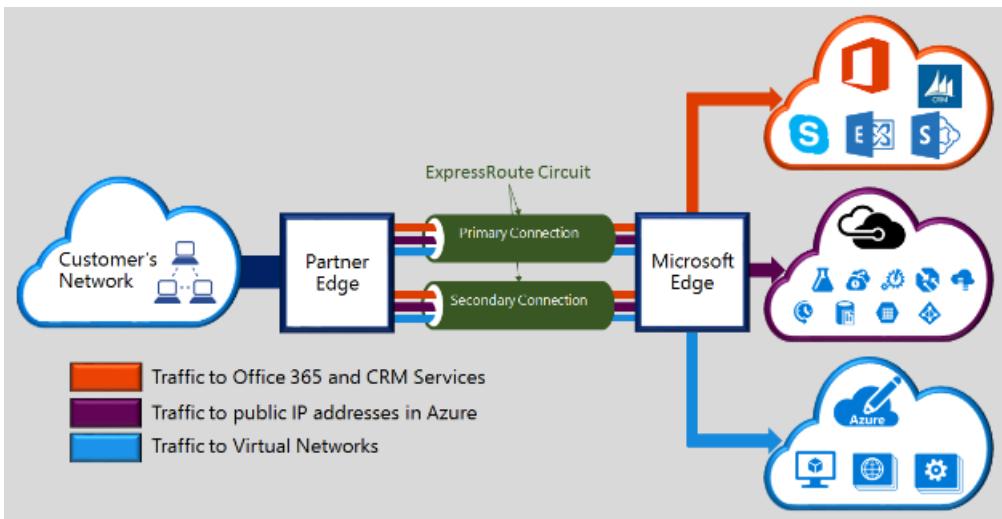
- [Connect individual workstations to an Azure Virtual Network](#)
- [Connect on-premises network to an Azure Virtual Network with a VPN](#)
- [Connect on-premises network to an Azure Virtual Network with a dedicated WAN link](#)
- [Connect Azure Virtual Networks to each other](#)

VPN Gateway

To send network traffic between your Azure Virtual Network and your on-premises site, you must create a VPN gateway for your Azure Virtual Network. A [VPN gateway](#) is a type of virtual network gateway that sends encrypted traffic across a public connection. You can also use VPN gateways to send traffic between Azure Virtual Networks over the Azure network fabric.

Express Route

Microsoft Azure [ExpressRoute](#) is a dedicated WAN link that lets you extend your on-premises networks into the Microsoft cloud over a dedicated private connection facilitated by a connectivity provider.

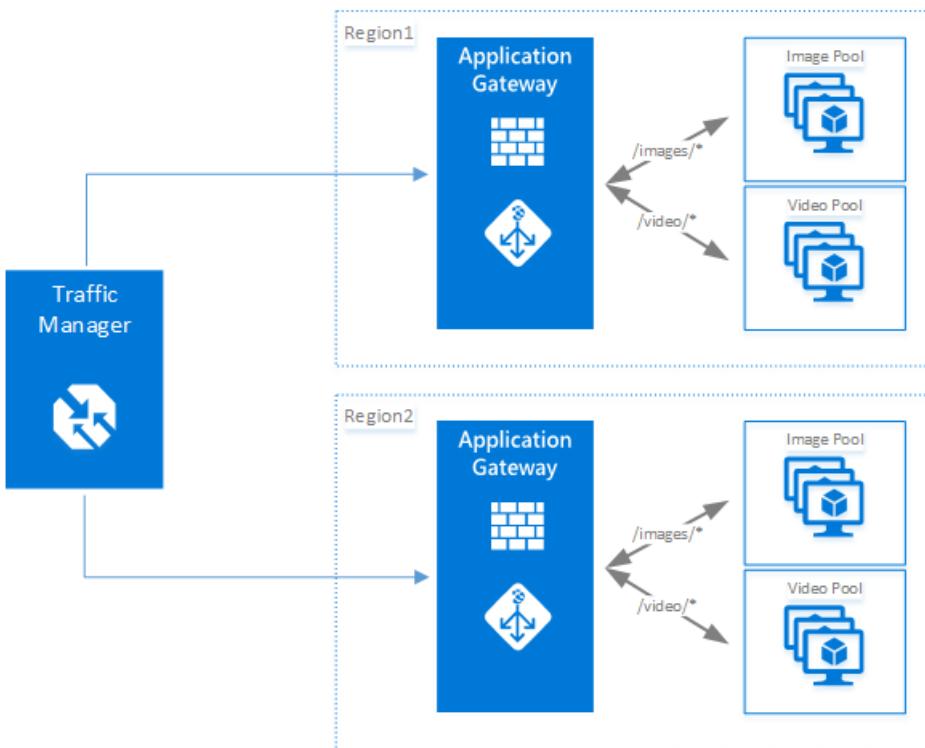


With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure, Office 365, and CRM Online. Connectivity can be from an any-to-any (IP VPN) network, a point-to-point Ethernet network, or a virtual cross-connection through a connectivity provider at a co-location facility.

ExpressRoute connections do not go over the public Internet and thus can be considered more secure than VPN-based solutions. This allows ExpressRoute connections to offer more reliability, faster speeds, lower latencies, and higher security than typical connections over the Internet.

Application Gateway

Microsoft [Azure Application Gateway](#) provides an [Application Delivery Controller \(ADC\)](#) as a service, offering various layer 7 load balancing capabilities for your application.



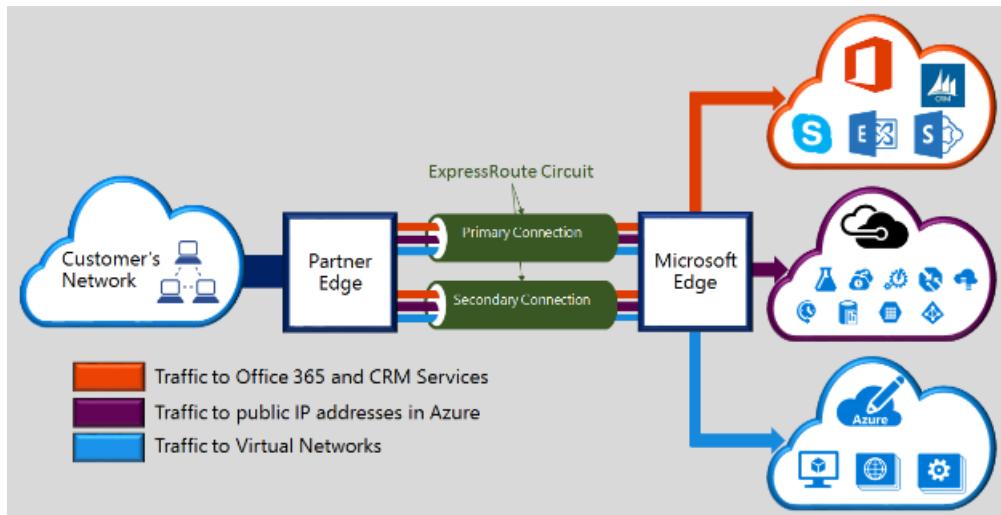
It allows you to optimize web farm productivity by offloading CPU intensive SSL termination to the Application Gateway (also known as "SSL offload" or "SSL bridging"). It also provides other Layer 7 routing capabilities including round-robin distribution of incoming traffic, cookie-based session affinity, URL path-based routing, and the ability to host multiple websites behind a single Application Gateway. Azure Application Gateway is a layer-7 load balancer.

It provides failover, performance-routing HTTP requests between different servers, whether they are on the cloud or on-premises.

Application provides many Application Delivery Controller (ADC) features including HTTP load balancing, cookie-based session affinity, [Secure Sockets Layer \(SSL\)](#) offload, custom health probes, support for multi-site, and many others.

Web Application Firewall

Web Application Firewall is a feature of [Azure Application Gateway](#) that provides protection to web applications that use application gateway for standard Application Delivery Control (ADC) functions. Web application firewall does this by protecting them against most of the OWASP top 10 common web vulnerabilities.



- SQL injection protection
- Common Web Attacks Protection such as command injection, HTTP request smuggling, HTTP response splitting, and remote file inclusion attack
- Protection against HTTP protocol violations
- Protection against HTTP protocol anomalies such as missing host user-agent and accept headers
- Prevention against bots, crawlers, and scanners
- Detection of common application misconfigurations (that is, Apache, IIS, etc.)

A centralized web application firewall to protect against web attacks makes security management much simpler and gives better assurance to the application against the threats of intrusions. A WAF solution can also react to a security threat faster by patching a known vulnerability at a central location versus securing each of individual web applications. Existing application gateways can be converted to an application gateway with web application firewall easily.

Traffic Manager

Microsoft [Azure Traffic Manager](#) allows you to control the distribution of user traffic for service endpoints in different data centers. Service endpoints supported by Traffic Manager include Azure VMs, Web Apps, and Cloud services. You can also use Traffic Manager with external, non-Azure endpoints. Traffic Manager uses the Domain Name System (DNS) to direct client requests to the most appropriate endpoint based on a [traffic-routing method](#) and the health of the endpoints.

Traffic Manager provides a range of traffic-routing methods to suit different application needs, endpoint health [monitoring](#), and automatic failover. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

Azure Load Balancer

[Azure Load Balancer](#) delivers high availability and network performance to your applications. It is a Layer 4 (TCP, UDP) load balancer that distributes incoming traffic among healthy instances of services defined in a load-balanced set. Azure Load Balancer can be configured to:

- Load balance incoming Internet traffic to virtual machines. This configuration is known as [Internet-facing load balancing](#).
- Load balance traffic between virtual machines in a virtual network, between virtual machines in cloud services, or between on-premises computers and virtual machines in a cross-premises virtual network. This configuration is known as [internal load balancing](#).
- Forward external traffic to a specific virtual machine

Internal DNS

You can manage the list of DNS servers used in a VNet in the Management Portal, or in the network configuration file. Customer can add up to 12 DNS servers for each VNet. When specifying DNS servers, it's important to verify that you list customer's DNS servers in the correct order for customer's environment. DNS server lists do not work round-robin. They are used in the order that they are specified. If the first DNS server on the list is able to be reached, the client uses that DNS server regardless of whether the DNS server is functioning properly or not. To change the DNS server order for customer's virtual network, remove the DNS servers from the list and add them back in the order that customer wants. DNS supports the availability aspect of the "CIA" security triad.

Azure DNS

The [Domain Name System](#), or DNS, is responsible for translating (or resolving) a website or service name to its IP address. [Azure DNS](#) is a hosting service for DNS domains, providing name resolution using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records using the same credentials, APIs, tools, and billing as your other Azure services. DNS supports the availability aspect of the "CIA" security triad.

Log Analytics NSGs

You can enable the following diagnostic log categories for NSGs:

- Event: Contains entries for which NSG rules are applied to VMs and instance roles based on MAC address. The status for these rules is collected every 60 seconds.
- Rules counter: Contains entries for how many times each NSG rule is applied to deny or allow traffic.

Azure Security Center

Security Center helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the Security of your Azure resources. It provides integrated Security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of Security solutions. Network recommendations center around firewalls, Network Security Groups, configuring inbound traffic rules, and more.

Available network recommendations are as follows:

- [Add a Next Generation Firewall](#) Recommends that you add a Next Generation Firewall (NGFW) from a Microsoft partner to increase your security protections
- [Route traffic through NGFW only](#) Recommends that you configure network security group (NSG) rules that force inbound traffic to your VM through your NGFW.
- [Enable Network Security Groups on subnets or virtual machines](#) Recommends that you enable NSGs on subnets or VMs.
- [Restrict access through Internet facing endpoint](#) Recommends that you configure inbound traffic rules for NSGs.

Compute

The section provides additional information regarding key features in this area and summary information about

these capabilities.

Antimalware & Antivirus

With Azure IaaS, you can use antimalware software from security vendors such as Microsoft, Symantec, Trend Micro, McAfee, and Kaspersky to protect your virtual machines from malicious files, adware, and other threats.

[Microsoft Antimalware](#) for Azure Cloud Services and Virtual Machines is a protection capability that helps identify and remove viruses, spyware, and other malicious software. Microsoft Antimalware provides configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems. Microsoft Antimalware can also be deployed using Azure Security Center

Hardware Security Module

Encryption and authentication do not improve security unless the keys themselves are protected. You can simplify the management and security of your critical secrets and keys by storing them in [Azure Key Vault](#). Key Vault provides the option to store your keys in hardware Security modules (HSMs) certified to FIPS 140-2 Level 2 standards. Your SQL Server encryption keys for backup or [transparent data encryption](#) can all be stored in Key Vault with any keys or secrets from your applications. Permissions and access to these protected items are managed through [Azure Active Directory](#).

Virtual machine backup

[Azure Backup](#) is a solution that protects your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications that can lead to security issues. With Azure Backup, your virtual machines running Windows and Linux are protected.

Azure Site Recovery

An important part of your organization's [business continuity/disaster recovery \(BCDR\)](#) strategy is figuring out how to keep corporate workloads and apps up and running when planned and unplanned outages occur. [Azure Site Recovery](#) helps orchestrate replication, failover, and recovery of workloads and apps so that they are available from a secondary location if your primary location goes down.

SQL VM TDE

[Transparent data encryption \(TDE\)](#) and column level encryption (CLE) are SQL server encryption features. This form of encryption requires customers to manage and store the cryptographic keys you use for encryption.

The Azure Key Vault (AKV) service is designed to improve the security and management of these keys in a secure and highly available location. The SQL Server Connector enables SQL Server to use these keys from Azure Key Vault.

If you are running SQL Server with on-premises machines, there are steps you can follow to access Azure Key Vault from your on-premises SQL Server machine. But for SQL Server in Azure VMs, you can save time by using the Azure Key Vault Integration feature. With a few Azure PowerShell cmdlets to enable this feature, you can automate the configuration necessary for a SQL VM to access your key vault.

VM Disk Encryption

[Azure Disk Encryption](#) is a new capability that helps you encrypt your Windows and Linux IaaS virtual machine disks. It applies the industry standard BitLocker feature of Windows and the DM-Crypt feature of Linux to provide volume encryption for the OS and the data disks. The solution is integrated with Azure Key Vault to help you control and manage the disk-encryption keys and secrets in your Key Vault subscription. The solution also ensures that all data on the virtual machine disks are encrypted at rest in your Azure storage.

Virtual networking

Virtual machines need network connectivity. To support that requirement, Azure requires virtual machines to be connected to an Azure Virtual Network. An Azure Virtual Network is a logical construct built on top of the physical Azure network fabric. Each logical [Azure Virtual Network](#) is isolated from all other Azure Virtual Networks. This isolation helps insure that network traffic in your deployments is not accessible to other Microsoft Azure customers.

Patch Updates

Patch Updates provide the basis for finding and fixing potential problems and simplify the software update management process, both by reducing the number of software updates you must deploy in your enterprise and by increasing your ability to monitor compliance.

Security policy management and reporting

[Azure Security Center](#) helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated Security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Azure Security Center

Security Center helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Identify and access management

Securing systems, applications, and data begins with identity-based access controls. The identity and access management features that are built into Microsoft business products and services help protect your organizational and personal information from unauthorized access while making it available to legitimate users whenever and wherever they need it.

Secure Identity

Microsoft uses multiple security practices and technologies across its products and services to manage identity and access.

- [Multi-Factor Authentication](#) requires users to use multiple methods for access, on-premises and in the cloud. It provides strong authentication with a range of easy verification options, while accommodating users with a simple sign-in process.
- [Microsoft Authenticator](#) provides a user-friendly Multi-Factor Authentication experience that works with both Microsoft Azure Active Directory and Microsoft accounts, and includes support for wearables and fingerprint-based approvals.
- [Password policy enforcement](#) increases the security of traditional passwords by imposing length and complexity requirements, forced periodic rotation, and account lockout after failed authentication attempts.
- [Token-based authentication](#) enables authentication via Active Directory Federation Services (AD FS) or third-party secure token systems.
- [Role-based access control \(RBAC\)](#) enables you to grant access based on the user's assigned role, making it easy to give users only the amount of access they need to perform their job duties. You can customize RBAC per your organization's business model and risk tolerance.
- [Integrated identity management \(hybrid identity\)](#) enables you to maintain control of users' access across internal datacenters and cloud platforms, creating a single user identity for authentication and authorization to all resources.

Secure Apps and data

[Azure Active Directory](#), a comprehensive identity and access management cloud solution, helps secure access to data in applications on site and in the cloud, and simplifies the management of users and groups. It combines core directory services, advanced identity governance, security, and application access management, and makes it easy for developers to build policy-based identity management into their apps. To enhance your Azure Active Directory, you can add paid capabilities using the Azure Active Directory Basic, Premium P1, and Premium P2 editions.

FREE / COMMON FEATURES	BASIC FEATURES	PREMIUM P1 FEATURES	PREMIUM P2 FEATURES	AZURE ACTIVE DIRECTORY JOIN – WINDOWS 10 ONLY RELATED FEATURES
Directory Objects, User/Group Management (add/update/delete)/ User-based provisioning, Device registration, Single Sign-On (SSO), Self-Service Password Change for cloud users, Connect (Sync engine that extends on-premises directories to Azure Active Directory), Security / Usage Reports	Group-based access management / provisioning, Self-Service Password Reset for cloud users, Company Branding (Logon Pages/Access Panel customization), Application Proxy, SLA 99.9%	Self-Service Group and app Management/Self-Service application additions/Dynamic Groups, Self-Service Password Reset/Change/Unlock with on-premises write-back, Multi-Factor Authentication (Cloud and On-premises (MFA Server)), MIM CAL + MIM Server, Cloud App Discovery, Connect Health, Automatic password rollover for group accounts	Identity Protection, Privileged Identity Management	Join a device to Azure AD, Desktop SSO, Microsoft Passport for Azure AD, Administrator Bitlocker recovery, MDM auto-enrollment, Self-Service Bitlocker recovery, Additional local administrators to Windows 10 devices via Azure AD Join

- [Cloud App Discovery](#) is a premium feature of Azure Active Directory that enables you to identify cloud applications that are used by the employees in your organization.
- [Azure Active Directory Identity Protection](#) is a security service that uses Azure Active Directory anomaly detection capabilities to provide a consolidated view into risk events and potential vulnerabilities that could affect your organization's identities.
- [Azure Active Directory Domain Services](#) enables you to join Azure VMs to a domain without the need to deploy domain controllers. Users sign in to these VMs by using their corporate Active Directory credentials, and can seamlessly access resources.
- [Azure Active Directory B2C](#) is a highly available, global identity management service for consumer-facing apps that can scale to hundreds of millions of identities and integrate across mobile and web platforms. Your customers can sign in to all your apps through customizable experiences that use existing social media accounts, or you can create new standalone credentials.
- [Azure Active Directory B2B Collaboration](#) is a secure partner integration solution that supports your cross-company relationships by enabling partners to access your corporate applications and data selectively by using their self-managed identities.
- [Azure Active Directory Join](#) enables you to extend cloud capabilities to Windows 10 devices for centralized management. It makes it possible for users to connect to the corporate or organizational cloud through Azure Active Directory and simplifies access to apps and resources.
- [Azure Active Directory Application Proxy](#) provides SSO and secure remote access for web applications hosted on-premises.

Next Steps

- [Getting started with Microsoft Azure Security](#)

Azure services and features you can use to help secure your services and data within Azure

- [Azure Security Center](#)

Prevent, detect, and respond to threats with increased visibility and control over the security of your Azure resources

- [Security health monitoring in Azure Security Center](#)

The monitoring capabilities in Azure Security Center to monitor compliance with policies.

Azure Network Security Overview

6/27/2017 • 17 min to read • [Edit Online](#)

Microsoft Azure includes a robust networking infrastructure to support your application and service connectivity requirements. Network connectivity is possible between resources located in Azure, between on-premises and Azure hosted resources, and to and from the Internet and Azure.

The goal of this article is to make it easier for you to understand what Microsoft Azure has to offer in the area of network security. Here we provide basic explanations for core network security concepts and requirements. We also provide you information on what Azure has to offer in each of these areas as well as links to help you gain a deeper understanding of interesting areas.

This Azure Network Security Overview article focuses on the following areas:

- Azure networking
- Network access control
- Secure remote access and cross-premises connectivity
- Availability
- Name resolution
- DMZ architecture
- Monitoring and threat detection

Azure Networking

Virtual machines need network connectivity. To support that requirement, Azure requires virtual machines to be connected to an Azure Virtual Network. An Azure Virtual Network is a logical construct built on top of the physical Azure network fabric. Each logical Azure Virtual Network is isolated from all other Azure Virtual Networks. This helps insure that network traffic in your deployments is not accessible to other Microsoft Azure customers.

Learn more:

- [Virtual Network Overview](#)

Network Access Control

Network access control is the act of limiting connectivity to and from specific devices or subnets within an Azure Virtual Network. The goal of network access control is to limit access to your virtual machines and services to approved users and devices. Access controls are based on allow or deny decisions for connections to and from your virtual machine or service.

Azure supports several types of network access control such as:

- Network layer control
- Route control and forced tunneling
- Virtual network security appliances

Network Layer Control

Any secure deployment requires some measure of network access control. The goal of network access control is to restrict virtual machine communication to the necessary systems and that other communication attempts are blocked.

If you need basic network level access control (based on IP address and the TCP or UDP protocols), then you can

use Network Security Groups. A Network Security Group (NSG) is a basic stateful packet filtering firewall and it enables you to control access based on a [5-tuple](#). NSGs do not provide application layer inspection or authenticated access controls.

Learn more:

- [Network Security Groups](#)

Route Control and Forced Tunneling

The ability to control routing behavior on your Azure Virtual Networks is a critical network security and access control capability. If routing is configured incorrectly, applications and services hosted on your virtual machine may connect to unauthorized devices including systems owned and operated by potential attackers.

Azure networking supports the ability to customize the routing behavior for network traffic on your Azure Virtual Networks. This enables you to alter the default routing table entries in your Azure Virtual Network. Control of routing behavior helps you make sure that all traffic from a certain device or group of devices enters or leaves your virtual Network through a specific location.

For example, you might have a virtual network security appliance on your Azure Virtual Network. You want to make sure that all traffic to and from your Azure Virtual Network goes through that virtual security appliance. You can do this by configuring [User Defined Routes](#) in Azure.

[Forced tunneling](#) is a mechanism you can use to ensure that your services are not allowed to initiate a connection to devices on the Internet. Note that this is different from accepting incoming connections and then responding to them. Front-end web servers need to respond to requests from Internet hosts, and so Internet-sourced traffic is allowed inbound to these web servers and the web servers are allowed to respond.

What you don't want to allow is a front-end web server to initiate an outbound request. Such requests may represent a security risk because these connections could be used to download malware. Even if you do wish these front-end servers to initiate outbound requests to the Internet, you might want to force them to go through your on-premises web proxies so that you can take advantage of URL filtering and logging.

Instead, you would want to use forced tunneling to prevent this. When you enable forced tunneling, all connections to the Internet are forced through your on-premises gateway. You can configure forced tunneling by taking advantage of User Defined Routes.

Learn more:

- [What are User Defined Routes and IP Forwarding](#)

Virtual Network Security Appliances

While Network Security Groups, User Defined Routes, and forced tunneling provide you a level of security at the network and transport layers of the [OSI model](#), there may be times when you want to enable security at levels higher than the network.

For example, your security requirements might include:

- Authentication and authorization before allowing access to your application
- Intrusion detection and intrusion response
- Application layer inspection for high-level protocols
- URL filtering
- Network level antivirus and antimalware
- Anti-bot protection
- Application access control
- Additional DDoS protection (above the DDoS protection provided the Azure fabric itself)

You can access these enhanced network security features by using an Azure partner solution. You can find the

most current Azure partner network security solutions by visiting the [Azure Marketplace](#) and searching for "security" and "network security."

Secure Remote Access and Cross Premises Connectivity

Setup, configuration, and management of your Azure resources needs to be done remotely. In addition, you may want to deploy [hybrid IT](#) solutions that have components on-premises and in the Azure public cloud. These scenarios require secure remote access.

Azure networking supports the following secure remote access scenarios:

- Connect individual workstations to an Azure Virtual Network
- Connect your on-premises network to an Azure Virtual Network with a VPN
- Connect your on-premises network to an Azure Virtual Network with a dedicated WAN link
- Connect Azure Virtual Networks to each other

Connect Individual Workstations to an Azure Virtual Network

There may be times when you want to enable individual developers or operations personnel to manage virtual machines and services in Azure. For example, you need access to a virtual machine on an Azure Virtual Network and your security policy does not allow RDP or SSH remote access to individual virtual machines. In this case, you can use a point-to-site VPN connection.

The point-to-site VPN connection uses the [SSTP VPN](#) protocol to enable you to set up a private and secure connection between the user and the Azure Virtual Network. Once the VPN connection is established, the user will be able to RDP or SSH over the VPN link into any virtual machine on the Azure Virtual Network (assuming that the user can authenticate and is authorized).

Learn more:

- [Configure a Point-to-Site Connection to a Virtual Network using PowerShell](#)

Connect Your On-Premises Network to an Azure Virtual Network with a VPN

You may want to connect your entire corporate network, or portions of it, to an Azure Virtual Network. This is common in hybrid IT scenarios where companies [extend their on-premises datacenter into Azure](#). In many cases companies will host parts of a service in Azure and parts on-premises, such as when a solution includes front-end web servers in Azure and back-end databases on-premises. These types of "cross-premises" connections also make management of Azure located resources more secure and enable scenarios such as extending Active Directory domain controllers into Azure.

One way to accomplish this is to use a [site-to-site VPN](#). The difference between a site-to-site VPN and a point-to-site VPN is that a point-to-site VPN connects a single device to an Azure Virtual Network, while a site-to-site VPN connects an entire network (such as your on-premises network) to an Azure Virtual Network. Site-to-site VPNs to an Azure Virtual Network use the highly secure IPsec tunnel mode VPN protocol.

Learn more:

- [Create a Resource Manager VNet with a site-to-site VPN connection using the Azure Portal](#)
- [Planning and design for VPN gateway](#)

Connect Your On-premises Network to an Azure Virtual Network with a Dedicated WAN Link

Point-to-site and site-to-site VPN connections are effective for enabling cross-premises connectivity. However, some organizations consider them to have the following drawbacks:

- VPN connections move data over the Internet – this exposes these connections to potential security issues involved with moving data over a public network. In addition, reliability and availability for Internet connections cannot be guaranteed.

- VPN connections to Azure Virtual Networks may be considered bandwidth constrained for some applications and purposes, as they max out at around 200 Mbps.

Organizations that need the highest level of security and availability for their cross-premises connections typically use dedicated WAN links to connect to remote sites. Azure provides you the ability to use a dedicated WAN link that you can use to connect your on-premises network to an Azure Virtual Network. This is enabled through Azure ExpressRoute.

Learn more:

- [ExpressRoute technical overview](#)

Connect Azure Virtual Networks to Each Other

It is possible for you to use many Azure Virtual Networks for your deployments. There are many reasons why you might do this. One of the reasons might be to simplify management; another might be for security reasons.

Regardless of the motivation or rationale for putting resources on different Azure Virtual Networks, there may be times when you want resources on each of the networks to connect with one another.

One option would be for services on one Azure Virtual Network to connect to services on another Azure Virtual Network by “looping back” through the Internet. The connection would start on one Azure Virtual Network, go through the Internet, and then come back to the destination Azure Virtual Network. This option exposes the connection to the security issues inherent to any Internet-based communication.

A better option might be to create an Azure Virtual Network-to-Azure Virtual Network site-to-site VPN. This Azure Virtual Network-to-Azure Virtual Network site-to-site VPN uses the same [IPsec tunnel mode](#) protocol as the cross-premises site-to-site VPN connection mentioned above.

The advantage of using an Azure Virtual Network-to-Azure Virtual Network site-to-site VPN is that the VPN connection is established over the Azure network fabric instead of connecting over the Internet. This provides you an extra layer of security compared to site-to-site VPNs that connect over the Internet.

Learn more:

- [Configure a VNet-to-VNet Connection by using Azure Resource Manager and PowerShell](#)

Availability

Availability is a key component of any security program. If your users and systems can't access what they need to access over the network, the service can be considered compromised. Azure has networking technologies that support the following high-availability mechanisms:

- HTTP-based load balancing
- Network level load balancing
- Global load balancing

Load balancing is a mechanism designed to equally distribute connections among multiple devices. The goals of load balancing are:

- Increase availability – when you load balance connections across multiple devices, one or more of the devices can become unavailable and the services running on the remaining online devices can continue to serve the content from the service
- Increase performance – when you load balance connections across multiple devices, a single device doesn't have to take the processor hit. Instead, the processing and memory demands for serving the content is spread across multiple devices.

HTTP-based Load Balancing

Organizations that run web-based services often desire to have an HTTP-based load balancer in front of those web

services to help insure adequate levels of performance and high availability. In contrast to traditional network-based load balancers, the load balancing decisions made by HTTP-based load balancers are based on characteristics of the HTTP protocol, not on the network and transport layer protocols.

To provide you HTTP-based load balancing for your web-based services, Azure provides you the Azure Application Gateway. The Azure Application Gateway supports:

- HTTP-based load balancing – load balancing decisions are made based on characteristic special to the HTTP protocol
- Cookie-based session affinity – this capability makes sure that connections established to one of the servers behind that load balancer stays intact between the client and server. This insures stability of transactions.
- SSL offload – when a client connection is established with the load balancer, that session between the client and the load balancer is encrypted using the HTTPS (SSL/) protocol. However, in order to increase performance, you have the option to have the connection between the load balancer and the web server behind the load balancer use the HTTP (unencrypted) protocol. This is referred to as "SSL offload" because the web servers behind the load balancer don't experience the processor overhead involved with encryption, and therefore should be able to service requests more quickly.
- URL-based content routing – this feature makes it possible for the load balancer to make decisions on where to forward connections based on the target URL. This provides a lot more flexibility than solutions that make load balancing decisions based on IP addresses.

Learn more:

- [Application Gateway Overview](#)

Network Level Load Balancing

In contrast to HTTP-based load balancing, network level load balancing makes load balancing decisions based on IP address and port (TCP or UDP) numbers. You can gain the benefits of network level load balancing in Azure by using the Azure Load Balancer. Some key characteristics of the Azure Load Balancer include:

- Network level load balancing based on IP address and port numbers
- Support for any application layer protocol
- Load balances to Azure virtual machines and cloud services role instances
- Can be used for both Internet-facing (external load balancing) and non-Internet facing (internal load balancing) applications and virtual machines
- Endpoint monitoring, which is used to determine if any of the services behind the load balancer have become unavailable

Learn more:

- [Internet Facing load balancer between multiple Virtual Machines or services](#)
- [Internal Load Balancer Overview](#)

Global Load Balancing

Some organizations will want the highest level of availability possible. One way to reach this goal is to host applications in globally distributed datacenters. When an application is hosted in data centers located throughout the world, it's possible for an entire geopolitical region to become unavailable and still have the application up and running.

In addition to the availability advantages you get by hosting applications in globally distributed datacenters, you also can get performance benefits. These performance benefits can be obtained by using a mechanism that directs requests for the service to the datacenter that is nearest to the device that is making the request.

Global load balancing can provide you both of these benefits. In Azure, you can gain the benefits of global load balancing by using Azure Traffic Manager.

Learn more:

- [What is Traffic Manager?](#)

Name Resolution

Name resolution is a critical function for all services you host in Azure. From a security perspective, compromise of the name resolution function can lead to an attacker redirecting requests from your sites to an attacker's site.

Secure name resolution is a requirement for all your cloud hosted services.

There are two types of name resolution you need to address:

- Internal name resolution – internal name resolution is used by services on your Azure Virtual Networks, your on-premises networks, or both. Names used for internal name resolution are not accessible over the Internet. For optimal security, it's important that your internal name resolution scheme is not accessible to external users.
- External name resolution – external name resolution is used by people and devices outside of your on-premises and Azure Virtual Networks. These are the names that are visible to the Internet and are used to direct connection to your cloud-based services.

For internal name resolution, you have two options:

- An Azure Virtual Network DNS server – when you create a new Azure Virtual Network, a DNS server is created for you. This DNS server can resolve the names of the machines located on that Azure Virtual Network. This DNS server is not configurable and is managed by the Azure fabric manager, thus making it a secure name resolution solution.
- Bring your own DNS server – you have the option of putting a DNS server of your own choosing on your Azure Virtual Network. This DNS server could be an Active Directory integrated DNS server, or a dedicated DNS server solution provided by an Azure partner, which you can obtain from the Azure Marketplace.

Learn more:

- [Virtual Network Overview](#)
- [Manage DNS Servers used by a Virtual Network \(VNet\)](#)

For external DNS resolution, you have two options:

- Host your own external DNS server on-premises
- Host your own external DNS server with a service provider

Many large organizations will host their own DNS servers on-premises. They can do this because they have the networking expertise and global presence to do so.

In most cases, it's better to host your DNS name resolution services with a service provider. These service providers have the network expertise and global presence to ensure very high availability for your name resolution services. Availability is essential for DNS services because if your name resolution services fail, no one will be able to reach your Internet facing services.

Azure provides you a highly available and performant external DNS solution in the form of Azure DNS. This external name resolution solution takes advantage of the worldwide Azure DNS infrastructure. It allows you to host your domain in Azure using the same credentials, APIs, tools, and billing as your other Azure services. As part of Azure, it also inherits the strong security controls built into the platform.

Learn more:

- [Azure DNS Overview](#)

DMZ Architecture

Many enterprise organizations use DMZs to segment their networks to create a buffer-zone between the Internet and their services. The DMZ portion of the network is considered a low-security zone and no high-value assets are placed in that network segment. You'll typically see network security devices that have a network interface on the DMZ segment and another network interface connected to a network that has virtual machines and services that accept inbound connections from the Internet.

There are a number of variations of DMZ design and the decision to deploy a DMZ, and then what type of DMZ to use if you decide to use one, is based on your network security requirements.

Learn more:

- [Microsoft Cloud Services and Network Security](#)

Monitoring and threat detection

Azure provides capabilities to help you in this key area with early detection, monitoring and the ability to collect and review network traffic.

Azure Network Watcher

Azure Network Watcher includes a large number of capabilities that help with troubleshooting as well as provide a whole new set of tools to assist with the identification of security issues.

[Security Group View](#) helps with auditing and security compliance of Virtual Machines and can be used to perform programmatic audits comparing the baselines policies defined by your organization to effective rules for each of your VMs. This can help you identify any configuration drift.

[Packet capture](#) allows you to capture network traffic to and from the virtual machine. Besides helping by allowing you to collect network statistics and with the troubleshooting of application issues packet capture can be invaluable in the investigation of network intrusions. You can also use this functionality together with Azure Functions to start network captures in response to specific Azure alerts.

For more information on Azure Network Watcher and how to start testing some of the functionality in your labs take a look at the [Azure network watcher monitoring overview](#)

NOTE

Azure Network Watcher is still in public preview so it may not have the same level of availability and reliability as services that are in general availability release. Certain features may not be supported, may have constrained capabilities, and may not be available in all Azure locations. For the most up-to-date notifications on availability and status of this service, check the [Azure updates page](#)

Azure Security Center

Security Center helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a large set of security solutions.

Azure Security Center helps you optimize and monitor network security by:

- Providing network security recommendations
- Monitoring the state of your network security configuration
- Alerting you to network based threats both at the endpoint and network levels

Learn more:

- [Introduction to Azure Security Center](#)

Logging

Logging at a network level is a key function for any network security scenario. In Azure, you can log information obtained for Network Security Groups to get network level logging information. With NSG logging, you get information from:

- [Activity logs](#) – these logs are used to view all operations submitted to your Azure subscriptions. These logs are enabled by default and can be used within the Azure portal. They were previously known as "Audit logs" or "Operational Logs".
- Event logs – these logs provide information about what NSG rules were applied.
- Counter logs – these logs let you know how many times each NSG rule was applied to deny or allow traffic.

You can also use [Microsoft Power BI](#), a powerful data visualization tool, to view and analyze these logs.

Learn more:

- [Log Analytics for Network Security Groups \(NSGs\)](#)

Azure database security overview

8/9/2017 • 12 min to read • [Edit Online](#)

Security is a top concern when managing databases, and it has always been a priority for Azure SQL Database. Azure SQL Database supports connection security with firewall rules and connection encryption. It supports authentication with username and password and Azure Active Directory Authentication, which uses identities managed by Azure Active Directory. Authorization uses role-based access control.

Azure SQL Database supports encryption by performing real-time encryption and decryption of databases, associated backups, and transaction log files at rest without requiring changes to the application.

Microsoft provides additional ways to encrypt enterprise data:

- Cell-level encryption to encrypt specific columns or even cells of data with different encryption keys.
- If you need a Hardware Security Module or central management of your encryption key hierarchy, consider using Azure Key Vault with SQL Server in an Azure VM.
- Always Encrypted (currently in preview) makes encryption transparent to applications and allows clients to encrypt sensitive data inside client applications without sharing the encryption keys with SQL Database.

Azure SQL Database Auditing allows enterprises to record events to an audit log in Azure Storage. SQL Database Auditing also integrates with Microsoft Power BI to facilitate drill-down reports and analyses.

SQL Azure databases can be tightly secured to satisfy most regulatory or security requirements, including HIPAA, ISO 27001/27002, and PCI DSS Level 1, among others. A current list of security compliance certifications is available at the [Microsoft Azure Trust Center site](#).

This article walks through the basics of securing Microsoft Azure SQL Databases for Structured, Tabular and Relational Data. In particular, this article will get you started with resources for protecting data, controlling access, and proactive monitoring.

This Azure Database Security Overview article focuses on the following areas:

- Protect data
- Access control
- Proactive monitoring
- Centralized security management
- Azure marketplace

Protect data

SQL Database secures your data by providing encryption for data in motion using [Transport Layer Security](#), for data at rest using [Transparent Data Encryption](#), and for data in use using [Always Encrypted](#).

In this section, we talk about:

- Encryption in motion
- Encryption at rest
- Encryption in use (Client)

For other ways to encrypt your data, consider:

- [Cell-level encryption](#) to encrypt specific columns or even cells of data with different encryption keys.
- If you need a Hardware Security Module or central management of your encryption key hierarchy, consider

using [Azure Key Vault with SQL Server in an Azure VM](#).

Encryption in motion

A common problem for all client/server applications is the need for privacy as data moves over public and private networks. If data moving over a network is not encrypted, there's the chance that it can be captured and stolen by unauthorized users. When dealing with database services, you need to make sure that data is encrypted between the database client and server, as well as between database servers that communicate with each other and with middle-tier applications.

One problem when you administer a network is securing data that is being sent between applications across an untrusted network. You can use [TLS/SSL](#) to authenticate servers and clients and then use it to encrypt messages between the authenticated parties.

In the authentication process, a TLS/SSL client sends a message to a TLS/SSL server, and the server responds with the information that the server needs to authenticate itself. The client and server perform an additional exchange of session keys, and the authentication dialog ends. When authentication is completed, SSL-secured communication can begin between the server and the client using the symmetric encryption keys that are established during the authentication process.

All connections to Azure SQL Database require encryption (SSL/TLS) at all times while data is "in transit" to and from the database. SQL Azure uses TLS/SSL to authenticate servers and clients and then use it to encrypt messages between the authenticated parties. In your application's connection string, you must specify parameters to encrypt the connection and not to trust the server certificate (this is done for you if you copy your connection string out of the Azure Classic Portal), otherwise the connection will not verify the identity of the server and will be susceptible to "man-in-the-middle" attacks. For the ADO.NET driver, for instance, these connection string parameters are Encrypt=True and TrustServerCertificate=False.

Encryption at rest

You can take several precautions to help secure the database such as designing a secure system, encrypting confidential assets, and building a firewall around the database servers. However, in a scenario where the physical media (such as drives or backup tapes) are stolen, a malicious party can just restore or attach the database and browse the data.

One solution is to encrypt the sensitive data in the database and protect the keys that are used to encrypt the data with a certificate. This prevents anyone without the keys from using the data, but this kind of protection must be planned.

To solve this problem, SQL Server and Azure SQL support [Transparent Data Encryption \(TDE\)](#). TDE encrypts SQL Server and Azure SQL Database data files, known as encryption data at rest.

Azure SQL Database transparent data encryption helps protect against the threat of malicious activity by performing real-time encryption and decryption of the database, associated backups, and transaction log files at rest without requiring changes to the application.

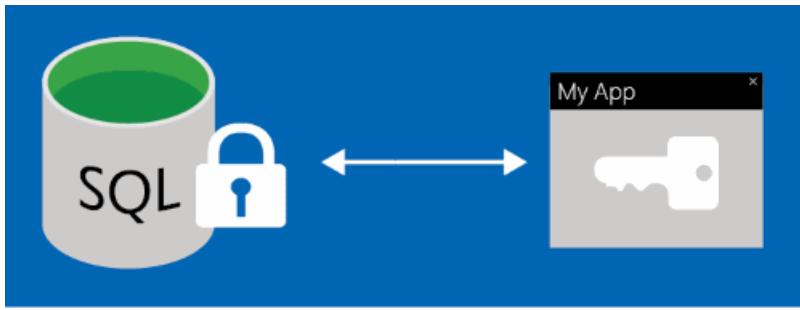
TDE encrypts the storage of an entire database by using a symmetric key called the database encryption key. In SQL Database, the database encryption key is protected by a built-in server certificate. The built-in server certificate is unique for each SQL Database server.

If a database is in a GeoDR relationship, it is protected by a different key on each server. If two databases are connected to the same server, they share the same built-in certificate. Microsoft automatically rotates these certificates at least every 90 days. For a general description of TDE, see [Transparent Data Encryption \(TDE\)](#).

Encryption in use (client)

Most data breaches involve the theft of critical data such as credit card numbers or personally identifiable information. Databases can be treasure troves of sensitive information. They can contain customers' personal data, confidential competitive information, and intellectual property. Lost or stolen data, especially customer data, can

result in brand damage, competitive disadvantage, and serious fines—even lawsuits.



[Always Encrypted](#) is a feature designed to protect sensitive data, such as credit card numbers or national identification numbers (for example, U.S. social security numbers), stored in Azure SQL Database or SQL Server databases. Always Encrypted allows clients to encrypt sensitive data inside client applications and never reveal the encryption keys to the Database Engine (SQL Database or SQL Server).

Always Encrypted provides a separation between those who own the data (and can view it) and those who manage the data (but should have no access). By ensuring on-premises database administrators, cloud database operators, or other high-privileged, but unauthorized users, cannot access the encrypted data,

In addition, Always Encrypted makes encryption transparent to applications. An Always Encrypted-enabled driver installed on the client computer so that it can automatically encrypt and decrypt sensitive data in the client application. The driver encrypts the data in sensitive columns before passing the data to the Database Engine, and automatically rewrites queries so that the semantics to the application are preserved. Similarly, the driver transparently decrypts data, stored in encrypted database columns, contained in query results.

Access control

To provide security, SQL Database controls access with firewall rules limiting connectivity by IP address, authentication mechanisms requiring users to prove their identity, and authorization mechanisms limiting users to specific actions and data.

Database access

Data protection begins with controlling access to your data. The datacenter hosting your data manages physical access, while you can configure a firewall to manage security at the network layer. You also control access by configuring logins for authentication and defining permissions for server and database roles.

In this section, we talk about:

- Firewall and firewall rules
- Authentication
- Authorization

Firewall and firewall rules

Microsoft Azure SQL Database provides a relational database service for Azure and other Internet-based applications. To help protect your data, firewalls prevent all access to your database server until you specify which computers have permission. The firewall grants access to databases based on the originating IP address of each request. For more information, see [Overview of Azure SQL Database firewall rules](#).

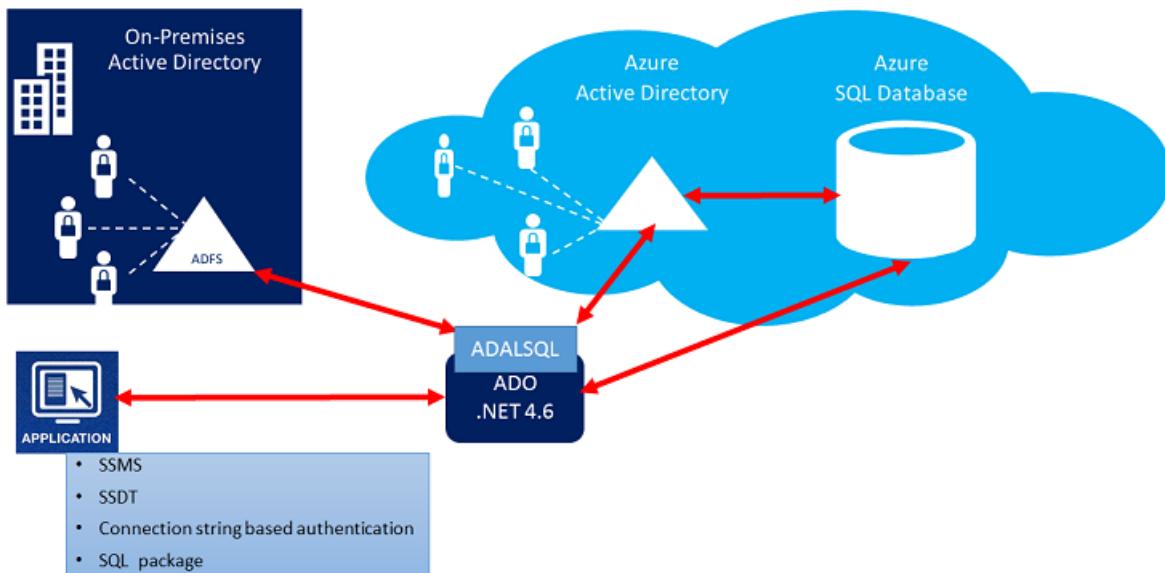
The [Azure SQL Database](#) service is only available through TCP port 1433. To access a SQL Database from your computer, ensure that your client computer firewall allows outgoing TCP communication on TCP port 1433. If not needed for other applications, block inbound connections on TCP port 1433.

Authentication

SQL database authentication refers to how you prove your identity when connecting to the database. SQL Database supports two types of authentication:

- **SQL Authentication:** A single login account is created when a logical SQL instance is created, called the SQL Database Subscriber Account. This account connects using [SQL Server authentication](#) (user name and password). This account is an administrator on the logical server instance and on all user databases attached to that instance. The permissions of the Subscriber Account cannot be restricted. Only one of these accounts can exist.
- **Azure Active Directory Authentication:** [Azure Active Directory authentication](#) is a mechanism of connecting to Microsoft Azure SQL Database and SQL Data Warehouse by using identities in Azure Active Directory (Azure AD). This enables you to centrally manage identities of database users.

Azure AD Authentication with SQL DB



Advantages of Azure Active Directory authentication include:

- It provides an alternative to SQL Server authentication.
- It also Helps stop the proliferation of user identities across database servers & allows password rotation in a single place.
- You can manage database permissions using external (Azure Active Directory) groups.
- It can eliminate storing passwords by enabling integrated Windows authentication and other forms of authentication supported by Azure Active Directory.

Authorization

[Authorization](#) refers to what a user can do within an Azure SQL Database, and this is controlled by your user account's database [role memberships](#) and [object-level permissions](#). Authorization is the process of determining which securable resources a principal can access, and which operations are allowed for those resources.

Application access

In this section, we talk about:

- Dynamic data masking
- Row-level security

Dynamic data masking

A service representative at a call center may identify callers by several digits of their social security number or credit card number, but those data items should not be fully exposed to the service representative.

A masking rule can be defined that masks all but the last four digits of any social security number or credit card number in the result set of any query.



As another example, an appropriate data mask can be defined to protect personally identifiable information (PII) data, so that a developer can query production environments for troubleshooting purposes without violating compliance regulations.

[SQL Database Dynamic Data Masking](#) limits sensitive data exposure by masking it to non-privileged users.

Dynamic data masking is supported for the V12 version of Azure SQL Database.

[Dynamic data masking](#) helps prevent unauthorized access to sensitive data by enabling you to designate how much of the sensitive data to reveal with minimal impact on the application layer. It's a policy-based security feature that hides the sensitive data in the result set of a query over designated database fields, while the data in the database is not changed.

NOTE

Dynamic data masking can be configured by the Azure Database admin, server admin, or security officer roles.

Row level security

Another common security requirement for multitenant databases is [Row-Level Security](#). This feature enables you to control access to rows in a database table based on the characteristics of the user executing a query (e.g., group membership or execution context).



The access restriction logic is located in the database tier rather than away from the data in another application tier. The database system applies the access restrictions every time that data access is attempted from any tier. This makes your security system more reliable and robust by reducing the surface area of your security system.

Row level security introduces predicate based access control. It features a flexible, centralized, predicate-based evaluation that can take into consideration metadata or any other criteria the administrator determines as

appropriate. The predicate is used as a criterion to determine whether or not the user has the appropriate access to the data based on user attributes. Label-based access control can be implemented by using predicate-based access control.

Proactive monitoring

SQL Database secures your data by providing **auditing** and **threat detection** capabilities.

Auditing

SQL Database Auditing increases your ability to gain insight into events and changes that occur within the database, including updates and queries against the data.

[Azure SQL Database Auditing](#) tracks database events and writes them to an audit log in your Azure Storage account. Auditing can help you maintain regulatory compliance, understand database activity, and gain insight into discrepancies and anomalies that could indicate business concerns or suspected security violations. Auditing enables and facilitates adherence to compliance standards but doesn't guarantee compliance.

SQL Database Auditing allows you to:

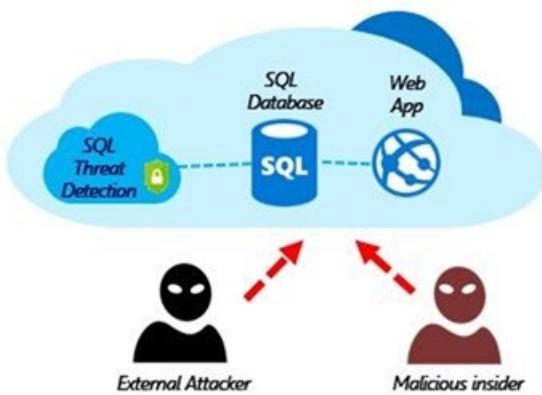
- **Retain** an audit trail of selected events. You can define categories of database actions to be audited.
- **Report** on database activity. You can use preconfigured reports and a dashboard to get started quickly with activity and event reporting.
- **Analyze** reports. You can find suspicious events, unusual activity, and trends.

There are two Auditing methods:

- **Blob auditing** - logs are written to Azure Blob Storage. This is a newer auditing method, which provides higher performance, supports higher granularity object-level auditing, and is more cost effective.
- **Table auditing** - logs are written to Azure Table Storage.

Threat detection

[Azure SQL Database threat detection](#) detects suspicious activities that indicate potential security threats. Threat detection enables you to respond to suspicious events in the database, such as SQL Injections, as they occur. It provides alerts and allows the use of Azure SQL Database Auditing to explore the suspicious events.



For example, SQL injection is one of the common Web application security issues on the Internet, used to attack data-driven applications. Attackers take advantage of application vulnerabilities to inject malicious SQL statements into application entry fields, breaching or modifying data in the database.

Security officers or other designated administrators can get an immediate notification about suspicious database activities as they occur. Each notification provides details of the suspicious activity and recommends how to further investigate and mitigate the threat.

Centralized security management

Azure Security Center helps you prevent, detect, and respond to threats. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Security Center helps you safeguard data in SQL Database by providing visibility into the security of all your servers and databases. With Security Center, you can:

- Define policies for SQL Database encryption and auditing.
- Monitor the security of SQL Database resources across all your subscriptions.
- Quickly identify and remediate security issues.
- Integrate alerts from [Azure SQL Database threat detection](#).
- Security Center supports role-based access.

Azure Marketplace

The Azure Marketplace is an online applications and services marketplace that enables start-ups and independent software vendors (ISVs) to offer their solutions to Azure customers around the world. The Azure Marketplace combines Microsoft Azure partner ecosystems into a single, unified platform to better serve our customers and partners. Click [here](#) to glance database security products available on Azure market place.

Next steps

- Learn more about [Secure your Azure SQL Database](#).
- Learn more about [Azure Security Center and Azure SQL Database service](#).
- To learn more about threat detection, see [SQL Database Threat Detection](#).
- To learn more, see [Improve SQL database performance](#).

Azure storage security overview

8/21/2017 • 4 min to read • [Edit Online](#)

Azure Storage is the cloud storage solution for modern applications that rely on durability, availability, and scalability to meet the needs of their customers. Azure Storage provides a comprehensive set of security capabilities:

- The storage account can be secured using Role-Based Access Control and Azure Active Directory.
- Data can be secured in transit between an application and Azure by using Client-Side Encryption, HTTPS, or SMB 3.0.
- Data can be set to be automatically encrypted when written to Azure Storage using Storage Service Encryption.
- OS and Data disks used by virtual machines can be set to be encrypted using Azure Disk Encryption.
- Delegated access to the data objects in Azure Storage can be granted using Shared Access Signatures.
- The authentication method used by someone when they access storage can be tracked using Storage analytics.

For a more detailed look at security in Azure Storage, see the [Azure Storage security guide](#). This guide provides a deep dive into the security features of Azure Storage such as storage account keys, data encryption in transit and at rest, and storage analytics.

This article provides an overview of Azure security features that can be used with Azure Storage. Links are provided to articles that give details of each feature so you can learn more.

Here are the core features to be covered in this article:

- Role-Based Access Control
- Delegated access to storage objects
- Encryption in transit
- Encryption at rest/Storage Service Encryption
- Azure Disk Encryption
- Azure Key Vault

Role-Based Access Control (RBAC)

You can secure your storage account with Role-Based Access Control (RBAC). Restricting access based on the [need to know](#) and [least privilege](#) security principles is imperative for organizations that want to enforce security policies for data access. These access rights are granted by assigning the appropriate RBAC role to groups and applications at a certain scope. You can use [built-in RBAC roles](#), such as Storage Account Contributor, to assign privileges to users.

Learn more:

- [Azure Active Directory Role-based Access Control](#)

Delegated access to storage objects

A shared access signature (SAS) provides delegated access to resources in your storage account. The SAS means that you can grant a client limited permissions to objects in your storage account for a specified period of time and with a specified set of permissions. You can grant these limited permissions without having to share your account access keys. The SAS is a URI that encompasses in its query parameters all the information necessary for authenticated access to a storage resource. To access storage resources with the SAS, the client only needs to provide the SAS to the appropriate constructor or method.

Learn more:

- [Understanding the SAS model](#)
- [Create and use a SAS with Blob storage](#)

Encryption in transit

Encryption in transit is a mechanism of protecting data when it is transmitted across networks. With Azure Storage you can secure data using:

- [Transport-level encryption](#), such as HTTPS when you transfer data into or out of Azure Storage.
- [Wire encryption](#), such as SMB 3.0 encryption for Azure File shares.
- [Client-side encryption](#), to encrypt the data before it is transferred into storage and to decrypt the data after it is transferred out of storage.

Learn more about client-side encryption:

- [Client-Side Encryption for Microsoft Azure Storage](#)
- [Cloud security controls series: Encrypting Data in Transit](#)

Encryption at rest

For many organizations, [data encryption at rest](#) is a mandatory step towards data privacy, compliance, and data sovereignty. There are three Azure features that provide encryption of data that is "at rest":

- [Storage Service Encryption](#) allows you to request that the storage service automatically encrypt data when writing it to Azure Storage.
- [Client-side Encryption](#) also provides the feature of encryption at rest.
- [Azure Disk Encryption](#) allows you to encrypt the OS disks and data disks used by an IaaS virtual machine.

Learn more about Storage Service Encryption:

- [Azure Storage Service Encryption](#) is available for [Azure Blob Storage](#). For details on other Azure storage types, see [File](#), [Disk \(Premium Storage\)](#), [Table](#), and [Queue](#).
- [Azure Storage Service Encryption for Data at Rest](#)

Azure Disk Encryption

Azure Disk Encryption for virtual machines (VMs) helps you address organizational security and compliance requirements by encrypting your VM disks (including boot and data disks) with keys and policies you control in [Azure Key Vault](#).

Disk Encryption for VMs works for Linux and Windows operating systems. It also uses Key Vault to help you safeguard, manage, and audit use of your disk encryption keys. All the data in your VM disks is encrypted at rest by using industry-standard encryption technology in your Azure Storage accounts. The Disk Encryption solution for Windows is based on [Microsoft BitLocker Drive Encryption](#), and the Linux solution is based on [dm-crypt](#).

Learn more:

- [Azure Disk Encryption for Windows and Linux IaaS Virtual Machines](#)

Azure Key Vault

Azure Disk Encryption uses [Azure Key Vault](#) to help you control and manage disk encryption keys and secrets in your key vault subscription, while ensuring that all data in the virtual machine disks are encrypted at rest in your Azure Storage. You should use Key Vault to audit keys and policy usage.

Learn more:

- [What is Azure Key Vault?](#)
- [Get started with Azure Key Vault](#)

Azure Virtual Machines security overview

8/9/2017 • 7 min to read • [Edit Online](#)

Azure Virtual Machines lets you deploy a wide range of computing solutions in an agile way. With support for Microsoft Windows, Linux, Microsoft SQL Server, Oracle, IBM, SAP, and Azure BizTalk Services, you can deploy any workload and any language on nearly any operating system.

An Azure virtual machine gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs the virtual machine. You can build and deploy your applications with the assurance that your data is protected and safe in our highly secure datacenters.

With Azure, you can build security-enhanced, compliant solutions that:

- Protect your virtual machines from viruses and malware
- Encrypt your sensitive data
- Secure network traffic
- Identify and detect threats
- Meet compliance requirements

The goal of this article is to provide an overview of the core Azure security features that can be used with virtual machines. We also provide links to articles that give details of each feature so you can learn more.

The core Azure Virtual Machine security capabilities to be covered in this article:

- Antimalware
- Hardware Security Module
- Virtual machine disk encryption
- Virtual machine backup
- Azure Site Recovery
- Virtual networking
- Security policy management and reporting
- Compliance

Antimalware

With Azure, you can use antimalware software from security vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky to protect your virtual machines from malicious files, adware, and other threats. See the Learn More section below to find articles on partner solutions.

Microsoft Antimalware for Azure Cloud Services and Virtual Machines is a real-time protection capability that helps identify and remove viruses, spyware, and other malicious software. Microsoft Antimalware provides configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems.

Microsoft Antimalware is a single-agent solution for applications and tenant environments, designed to run in the background without human intervention. You can deploy protection based on the needs of your application workloads, with either basic secure-by-default or advanced custom configuration, including antimalware monitoring.

When you deploy and enable Microsoft Antimalware, the following core features are available:

- Real-time protection - monitors activity in Cloud Services and on Virtual Machines to detect and block malware execution.

- Scheduled scanning - periodically performs targeted scanning to detect malware, including actively running programs.
- Malware remediation - automatically takes action on detected malware, such as deleting or quarantining malicious files and cleaning up malicious registry entries.
- Signature updates - automatically installs the latest protection signatures (virus definitions) to ensure protection is up-to-date on a pre-determined frequency.
- Antimalware Engine updates – automatically updates the Microsoft Antimalware engine.
- Antimalware Platform updates – automatically updates the Microsoft Antimalware platform.
- Active protection - reports to Azure telemetry metadata about detected threats and suspicious resources to ensure rapid response and enables real-time synchronous signature delivery through the Microsoft Active Protection System (MAPS).
- Samples reporting - provides and reports samples to the Microsoft Antimalware service to help refine the service and enable troubleshooting.
- Exclusions – allows application and service administrators to configure certain files, processes, and drives to exclude them from protection and scanning for performance and other reasons.
- Antimalware event collection - records the antimalware service health, suspicious activities, and remediation actions taken in the operating system event log and collects them into the customer's Azure Storage account.

Learn more: To learn more about antimalware software to protect your virtual machines, see:

- [Microsoft Antimalware for Azure Cloud Services and Virtual Machines](#)
- [Deploying Antimalware Solutions on Azure Virtual Machines](#)
- [How to install and configure Trend Micro Deep Security as a Service on a Windows VM](#)
- [How to install and configure Symantec Endpoint Protection on a Windows VM](#)
- [Security solutions in the Azure Marketplace](#)

Hardware security Module

Encryption and authentication protections can be enhanced by improving key security. You can simplify the management and security of your critical secrets and keys by storing them in Azure Key Vault. Key Vault provides the option to store your keys in hardware security modules (HSMs) certified to FIPS 140-2 Level 2 standards. Your SQL Server encryption keys for backup or [transparent data encryption](#) can all be stored in Key Vault with any keys or secrets from your applications. Permissions and access to these protected items are managed through [Azure Active Directory](#).

Learn more:

- [What is Azure Key Vault?](#)
- [Get started with Azure Key Vault](#)
- [Azure Key Vault blog](#)

Virtual machine disk encryption

Azure Disk Encryption is a new capability that lets you encrypt your Windows and Linux Azure Virtual Machine disks. Azure Disk Encryption uses the industry standard [BitLocker](#) feature of Windows and the [dm-crypt](#) feature of Linux to provide volume encryption for the OS and the data disks.

The solution is integrated with Azure Key Vault to help you control and manage the disk encryption keys and secrets in your key vault subscription, while ensuring that all data in the virtual machine disks are encrypted at rest in your Azure storage.

Learn more:

- [Azure Disk Encryption for Windows and Linux IaaS VMs](#)

- [Azure Disk Encryption for Linux and Windows Virtual Machines](#)
- [Encrypt a virtual machine](#)

Virtual machine backup

Azure Backup is a scalable solution that protects your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications. With Azure Backup, your virtual machines running Windows and Linux are protected.

Learn more:

- [What is Azure Backup?](#)
- [Azure Backup Learning Path](#)
- [Azure Backup Service - FAQ](#)

Azure Site Recovery

An important part of your organization's BCDR strategy is figuring out how to keep corporate workloads and apps up and running when planned and unplanned outages occur. Azure Site Recovery helps orchestrate replication, failover, and recovery of workloads and apps so that they are available from a secondary location if your primary location goes down.

Site Recovery:

- **Simplifies your BCDR strategy** — Site Recovery makes it easy to handle replication, failover, and recovery of multiple business workloads and apps from a single location. Site recovery orchestrates replication and failover but doesn't intercept your application data or have any information about it.
- **Provides flexible replication** — Using Site Recovery you can replicate workloads running on Hyper-V virtual machines, VMware virtual machines, and Windows/Linux physical servers.
- **Supports failover and recovery** — Site Recovery provides test failovers to support disaster recovery drills without affecting production environments. You can also run planned failovers with a zero-data loss for expected outages, or unplanned failovers with minimal data loss (depending on replication frequency) for unexpected disasters. After failover, you can failback to your primary sites. Site Recovery provides recovery plans that can include scripts and Azure automation workbooks so that you can customize failover and recovery of multi-tier applications.
- **Eliminates secondary datacenter** — You can replicate to a secondary on-premises site, or to Azure. Using Azure as a destination for disaster recovery eliminates the cost and complexity of maintaining a secondary site. Replicated data is stored in Azure Storage.
- **Integrates with existing BCDR technologies** — Site Recovery partners with other application BCDR features. For example, you can use Site Recovery to protect the SQL Server back end of corporate workloads. This includes native support for SQL Server AlwaysOn to manage the failover of availability groups.

Learn more:

- [What is Azure Site Recovery?](#)
- [How Does Azure Site Recovery Work?](#)
- [What Workloads are Protected by Azure Site Recovery?](#)

Virtual networking

Virtual machines need network connectivity. To support that requirement, Azure requires virtual machines to be connected to an Azure Virtual Network. An Azure Virtual Network is a logical construct built on top of the physical Azure network fabric. Each logical Azure Virtual Network is isolated from all other Azure Virtual Networks. This isolation helps insure that network traffic in your deployments is not accessible to other Microsoft Azure

customers.

Learn more:

- [Azure Network Security Overview](#)
- [Virtual Network Overview](#)
- [Networking features and partnerships for Enterprise scenarios](#)

Security policy management and reporting

Azure Security Center helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Azure Security Center helps you optimize and monitor virtual machine security by:

- Providing virtual machine [security recommendations](#) such as apply system updates, configure ACLs endpoints, enable antimalware, enable network security groups, and apply disk encryption.
- Monitoring the state of your virtual machines

Learn more:

- [Introduction to Azure Security Center](#)
- [Azure Security Center Frequently Asked Questions](#)
- [Azure Security Center Planning and Operations](#)

Compliance

Azure Virtual Machines is certified for FISMA, FedRAMP, HIPAA, PCI DSS Level 1, and other key compliance programs. This certification makes it easier for your own Azure applications to meet compliance requirements and for your business to address a wide range of domestic and international regulatory requirements.

Learn more:

- [Microsoft Trust Center: Compliance](#)
- [Trusted Cloud: Microsoft Azure Security, Privacy, and Compliance](#)

Azure operational security overview

8/9/2017 • 10 min to read • [Edit Online](#)

Azure Operational Security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure. [Azure Operational Security](#) is a framework that incorporates the knowledge gained through a variety of capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Center program, and deep awareness of the cyber security threat landscape.

This Azure Operational Security Overview article focuses on the following areas:

- Azure Operations Management Suite
- Azure Security Center
- Azure Monitor
- Azure Network Watcher
- Azure Storage Analytics
- Azure Active Directory

Azure Operations Management Suite

IT Operations is responsible for managing datacenter infrastructure, applications, and data, including the stability and security of these systems. However, gaining security insights across increasing complex IT environments often requires organizations to cobble together data from multiple security and management systems.

[Microsoft Operations Management Suite \(OMS\)](#) is Microsoft's cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure.

OMS is a cloud-based IT management solution with many offerings, such as IT Automation, Security & Compliance, Log Analytics, and Backup & Recovery. As such, it's a perfect aid to manage and protect your IT infrastructure—on premises and in the cloud.

The core functionality of OMS is provided by a set of services that run in Azure. Each service provides a specific management function, and you can combine services to achieve different management scenarios. Which includes:

- Log Analytics
- Automation
- Backup
- Site Recovery

Log Analytics

[Log Analytics](#) provides monitoring services for OMS by collecting data from managed resources into a central repository. This data could include events, performance data, or custom data provided through the API. Once collected, the data is available for alerting, analysis, and export. This method allows you to consolidate data from a variety of sources so you can combine data from your Azure services with your existing on-premises environment. It also clearly separates the collection of the data from the action taken on that data so that all actions are available to all kinds of data.

Automation

Microsoft [Azure Automation](#) provides a way for users to automate the manual, long-running, error-prone, and frequently repeated tasks that are commonly performed in a cloud and enterprise environment. It saves time and increases the reliability of regular administrative tasks and even schedules them to be automatically performed at

regular intervals. You can automate processes using runbooks or automate configuration management using Desired State Configuration.

Backup

[Azure Backup](#) is the Azure-based service you can use to back up (or protect) and restore your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or off-site backup solution with a cloud-based solution that is reliable, secure, and cost-competitive. Azure Backup offers multiple components that you download and deploy on the appropriate computer, server, or in the cloud. The component, or agent, that you deploy depends on what you want to protect. All Azure Backup components (no matter whether you're protecting data on-premises or in the cloud) can be used to back up data to a Recovery Services vault in Azure. See the [Azure Backup components table](#).

Site recovery

[Azure Site Recovery](#) provides business continuity by orchestrating replication of on-premises virtual and physical machines to Azure, or to a secondary site. If your primary site is unavailable, you fail over to the secondary location so that users can keep working, and fail back when systems return to working order. intelligent and effective threat detection.

Azure Active Directory

[Azure Active Directory](#) is Microsoft's comprehensive Identity as a Service (IDaaS) solution that:

- Enables IAM as a cloud service
- Provides central access management, single-sign on (SSO), and reporting
- Supports integrated access management for [thousands of applications](#) in the application gallery, including Salesforce, Google Apps, Box, Concur, and more.

Azure AD also includes a full suite of [identity management capabilities](#) including [multi-factor authentication](#), [device registration](#), [self-service password management](#), [self-service group management](#), [privileged account management](#), [role-based access control](#), [application usage monitoring](#), [rich auditing](#), and [security monitoring and alerting](#).

With Azure Active Directory, all applications you publish for your partners and customers (business or consumer) have the same identity and access management capabilities. This enables you to significantly reduce your operational costs.

Azure Security Center

[Azure Security Center](#) helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. It provides integrated security monitoring and policy management across your subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

[Security Center](#) helps you safeguard virtual machine data in Azure by providing visibility into your virtual machine's security settings and monitoring for threats. Security Center can monitor your virtual machines for:

- Operating System (OS) security settings with the recommended configuration rules
- System security and critical updates that are missing
- Endpoint protection recommendations
- Disk encryption validation
- Network-based attacks

Azure Security Center uses [Role-Based Access Control \(RBAC\)](#), which provides [built-in roles](#) that can be assigned to users, groups, and services in Azure.

Security Center assesses the configuration of your resources to identify security issues and vulnerabilities. In Security Center, you only see information related to a resource when you are assigned the role of Owner,

Contributor, or Reader for the subscription or resource group that a resource belongs to.

NOTE

See [Permissions in Azure Security Center](#) to learn more about roles and allowed actions in Security Center.

Security Center uses the Microsoft Monitoring Agent – this is the same agent used by the Operations Management Suite and Log Analytics service. Data collected from this agent is stored in either an existing Log Analytics [workspace](#) associated with your Azure subscription or a new workspace(s), taking into account the geolocation of the VM.

Azure Monitor

Performance issues in your cloud app can impact your business. With multiple interconnected components and frequent releases, degradations can happen at any time. And if you're developing an app, your users usually discover issues that you didn't find in testing. You should know about these issues immediately, and have tools for diagnosing and fixing the problems.

[Azure Monitor](#) is basic tool for monitoring services running on Azure. It gives you infrastructure-level data about the throughput of a service and the surrounding environment. If you are managing your apps all in Azure, deciding whether to scale up or down resources, then Azure Monitor gives you what you use to start.

In addition, you can use monitoring data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention. It includes:

- Azure Activity Log
- Azure Diagnostic Logs
- Metrics
- Azure Diagnostics

Azure Activity Log

It is a log that provides insight into the operations that were performed on resources in your subscription. The [Activity Log](#) was previously known as "Audit Logs" or "Operational Logs," since it reports control-plane events for your subscriptions.

Azure Diagnostic Logs

[Azure Diagnostic Logs](#) are emitted by a resource and provide rich, frequent data about the operation of that resource. The content of these logs varies by resource type.

For example, Windows event system logs are one category of Diagnostic Log for VMs and blob, table, and queue logs are categories of Diagnostic Logs for storage accounts.

Diagnostics Logs differ from the [Activity Log \(formerly known as Audit Log or Operational Log\)](#). The Activity log provides insight into the operations that were performed on resources in your subscription. Diagnostics logs provide insight into operations that your resource performed itself.

Metrics

Azure Monitor enables you to consume telemetry to gain visibility into the performance and health of your workloads on Azure. The most important type of Azure telemetry data is the metrics (also called performance counters) emitted by most Azure resources. Azure Monitor provides several ways to configure and consume these [metrics](#) for monitoring and troubleshooting.

Azure Diagnostics

It is the capability within Azure that enables the collection of diagnostic data on a deployed application. You can use

the diagnostics extension from various different sources. Currently supported are [Azure Cloud Service Web and Worker Roles](#), [Azure Virtual Machines](#) running Microsoft Windows, and [Service Fabric](#).

Network Watcher

Customers build an end-to-end network in Azure by orchestrating and composing various individual network resources such as VNet, ExpressRoute, Application Gateway, Load balancers, and more. Monitoring is available on each of the network resources.

The end to end network can have complex configurations and interactions between resources, creating complex scenarios that need scenario-based monitoring through Network Watcher.

[Network Watcher](#) will simplifies monitoring and diagnosing of your Azure network. Diagnostic and visualization tools available with Network Watcher enable you to take remote packet captures on an Azure Virtual Machine, gain insights into your network traffic using flow logs, and diagnose VPN Gateway and Connections.

Network Watcher currently has the following capabilities:

- [Topology](#) - Provides a network level view showing the various interconnections and associations between network resources in a resource group.
- [Variable Packet capture](#) - Captures packet data in and out of a virtual machine. Advanced filtering options and fine-tuned controls such as being able to set time and size limitations provide versatility. The packet data can be stored in a blob store or on the local disk in .cap format.
- [IP flows verify](#) - Checks if a packet is allowed or denied based on flow information 5-tuple packet parameters (Destination IP, Source IP, Destination Port, Source Port, and Protocol). If the packet is denied by a security group, the rule and group that denied the packet is returned.
- [Next hop](#) - Determines the next hop for packets being routed in the Azure Network Fabric, enabling you to diagnose any misconfigured user-defined routes.
- [Security group view](#) - Gets the effective and applied security rules that are applied on a VM.
- [NSG Flow logging](#) - Flow logs for Network Security Groups enable you to capture logs related to traffic that are allowed or denied by the security rules in the group. The flow is defined by a 5-tuple information – Source IP, Destination IP, Source Port, Destination Port, and Protocol.
- [Virtual Network Gateway and Connection troubleshooting](#) - Provides the ability to troubleshoot Virtual Network Gateways and Connections.
- [Network subscription limits](#) - Enables you to view network resource usage against limits.
- [Configuring Diagnostics Log](#) – Provides a single pane to enable or disable Diagnostics logs for network resources in a resource group.

To learn more how to configure network watcher see, [configure network watcher](#).

Developer Operations (DevOps)

Prior to DevOps application development, teams were in charge of gathering business requirements for a software program and writing code. Then a separate QA team tests the program in an isolated development environment, if requirements were met, and releases the code for operations to deploy. The deployment teams are further fragmented into siloed groups like networking and database. Each time a software program is “thrown over the wall” to an independent team it adds bottlenecks.

[DevOps](#) enables teams to deliver more secure, higher-quality solutions faster, and cheaper. Customers expect a dynamic and reliable experience when consuming software and services. Teams must rapidly iterate on software updates, measure the impact of the updates, and respond quickly with new development iterations to address issues or provide more value. Cloud platforms such as Microsoft Azure have removed traditional bottlenecks and helped commoditize infrastructure. Software reigns in every business as the key differentiator and factor in business outcomes. No organization, developer, or IT worker can or should avoid the DevOps movement.

Mature DevOps practitioners adopt several of the following practices. These practices [involve people](#) to form strategies based on the business scenarios. Tooling can help automate the various practices:

- [Agile planning and project management](#) techniques are used to plan and isolate work into sprints, manage team capacity, and help teams quickly adapt to changing business needs.
- [Version control, usually with Git](#), enables teams located anywhere in the world to share source and integrate with software development tools to automate the release pipeline.
- [Continuous Integration](#) drives the ongoing merging and testing of code, which leads to finding defects early. Other benefits include less time wasted on fighting merge issues and rapid feedback for development teams.
- [Continuous Delivery](#) of software solutions to production and testing environments help organizations quickly fix bugs and respond to ever-changing business requirements.
- [Monitoring](#) of running applications including production environments for application health as well as customer usage help organizations form a hypothesis and quickly validate or disprove strategies. Rich data is captured and stored in various logging formats.
- [Infrastructure as Code \(IaC\)](#) is a practice, which enables the automation and validation of creation and teardown of networks and virtual machines to help with delivering secure, stable application hosting platforms.
- [Microservices](#) architecture is leveraged to isolate business use cases into small reusable services. This architecture enables scalability and efficiency.

Next steps

To learn more about OMS Security and Audit solution, see the following articles:

- [Operations Management Suite | Security & Compliance](#).
- [Monitoring and Responding to Security Alerts in Operations Management Suite Security and Audit Solution](#).
- [Monitoring Resources in Operations Management Suite Security and Audit Solution](#).

Azure Security Management and Monitoring Overview

8/11/2017 • 6 min to read • [Edit Online](#)

Azure provides security mechanisms to aid in the management and monitoring of Azure cloud services and virtual machines. This article provides an overview of these core security features and services. Links are provided to articles that give details of each so you can learn more.

The security of your Microsoft cloud services is a partnership and shared responsibility between you and Microsoft. Shared responsibility means Microsoft is responsible for the Microsoft Azure and physical security of its data centers (by using security protections such as locked badge entry doors, fences, and guards). In addition, Azure provides strong levels of cloud security at the software layer that meets the security, privacy, and compliance needs of its demanding customers.

You own your data and identities, the responsibility for protecting them, the security of your on-premises resources, and the security of cloud components over which you have control. Microsoft provides you with security controls and capabilities to help you protect your data and applications. Your degree of responsibility for security is based on the type of cloud service.

The following chart summarizes the balance of responsibility for both Microsoft and the customer.

Responsibility	SaaS	PaaS	IaaS	On-prem
Data governance & rights management	Customer	Customer	Customer	Customer
Client endpoints	Customer	Customer	Customer	Customer
Account & access management	Customer	Customer	Customer	Customer
Identity & directory infrastructure	Microsoft	Microsoft	Customer	Customer
Application	Microsoft	Microsoft	Customer	Customer
Network controls	Microsoft	Microsoft	Customer	Customer
Operating system	Microsoft	Microsoft	Customer	Customer
Physical hosts	Microsoft	Microsoft	Microsoft	Customer
Physical network	Microsoft	Microsoft	Microsoft	Customer
Physical datacenter	Microsoft	Microsoft	Microsoft	Customer
		Microsoft	Customer	

For a deeper dive into security management, see [Security management in Azure](#).

Here are the core features to be covered in this article:

- Role-Based Access Control
- Antimalware
- Multi-Factor Authentication
- ExpressRoute
- Virtual network gateways
- Privileged identity management
- Identity protection
- Security Center

Role-Based Access Control

Role-Based Access Control (RBAC) provides fine-grained access management for Azure resources. Using RBAC, you can grant people only the amount of access that they need to perform their jobs. RBAC can also help you ensure that when people leave the organization they lose access to resources in the cloud.

Learn more:

- [Active Directory team blog on RBAC](#)
- [Azure Role-Based Access Control](#)

Antimalware

With Azure, you can use antimalware software from major security vendors such as Microsoft, Symantec, Trend Micro, McAfee, and Kaspersky to help protect your virtual machines from malicious files, adware, and other threats.

Microsoft Antimalware offers you the ability to install an antimalware agent for both PaaS roles and virtual machines. Based on System Center Endpoint Protection, this feature brings proven on-premises security technology to the cloud.

We also offer deep integration for Trend's [Deep Security™](#) and [SecureCloud™](#) products in the Azure platform. DeepSecurity is an Antivirus solution and SecureCloud is an encryption solution. DeepSecurity is deployed inside VMs using an extension model. Using the portal UI and PowerShell, you can choose to use DeepSecurity inside new VMs that are being spun up, or existing VMs that are already deployed.

Symantec End Point Protection (SEP) is also supported on Azure. Through portal integration, customers can specify that they intend to use SEP within a VM. SEP can be installed on a brand new VM via the Azure portal or can be installed on an existing VM using PowerShell.

Learn more:

- [Deploying Antimalware Solutions on Azure Virtual Machines](#)
- [Microsoft Antimalware for Azure Cloud Services and Virtual Machines](#)
- [How to install and configure Trend Micro Deep Security as a Service on a Windows VM](#)
- [How to install and configure Symantec Endpoint Protection on a Windows VM](#)
- [New Antimalware Options for Protecting Azure Virtual Machines – McAfee Endpoint Protection](#)

Multi-Factor Authentication

Azure Multi-factor authentication (MFA) is a method of authentication that requires the use of more than one verification method and adds a critical second layer of security to user sign-ins and transactions. MFA helps safeguard access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication via a range of verification options—phone call, text message, or mobile app notification or verification code and third party OATH tokens.

Learn more:

- [Multi-factor authentication](#)
- [What is Azure Multi-Factor Authentication?](#)
- [How Azure Multi-Factor Authentication works](#)

ExpressRoute

Microsoft Azure ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a dedicated private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure, Office 365, and CRM Online. Connectivity can be from an any-to-any (IP VPN) network, a point-to-point Ethernet network, or a virtual cross-connection through a connectivity provider at a co-location facility. ExpressRoute connections do not go over the public Internet. This allows ExpressRoute connections to offer more reliability, faster speeds, lower latencies, and higher security than typical connections over the Internet.

Learn more:

- [ExpressRoute technical overview](#)

Virtual network gateways

VPN Gateways, also called Azure Virtual Network Gateways, are used to send network traffic between virtual networks and on-premises locations. They are also used to send traffic between multiple virtual networks within Azure (VNet-to-VNet). VPN gateways provide secure cross-premises connectivity between Azure and your infrastructure.

Learn more:

- [About VPN gateways](#)
- [Azure Network Security Overview](#)

Privileged Identity Management

Sometimes users need to carry out privileged operations in Azure resources or other SaaS applications. This often means organizations have to give them permanent privileged access in Azure Active Directory (Azure AD). This is a growing security risk for cloud-hosted resources because organizations can't sufficiently monitor what those users are doing with their privileged access. Additionally, if a user account with privileged access is compromised, that one breach could impact your overall cloud security. Azure AD Privileged Identity Management helps to resolve this risk by lowering the exposure time of privileges and increasing visibility into usage.

Privileged Identity Management introduces the concept of a temporary admin for a role or "just in time" administrator access, which is a user who needs to complete an activation process for that assigned role. The activation process changes the assignment of the user to a role in Azure AD from inactive to active, for a specified time period such as eight hours.

Learn more:

- [Azure AD Privileged Identity Management](#)
- [Get started with Azure AD Privileged Identity Management](#)

Identity Protection

Azure Active Directory (AD) Identity Protection provides a consolidated view of suspicious sign-in activities and potential vulnerabilities to help protect your business. Identity Protection detects suspicious activities for users and privileged (admin) identities, based on signals like brute-force attacks, leaked credentials, and sign-ins from unfamiliar locations and infected devices.

By providing notifications and recommended remediation, Identity Protection helps to mitigate risks in real time. It calculates user risk severity, and you can configure risk-based policies to automatically help safeguard application access from future threats.

Learn more:

- [Azure Active Directory Identity Protection](#)
- [Channel 9: Azure AD and Identity Show: Identity Protection Preview](#)

Security Center

Azure Security Center helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Security Center helps you optimize and monitor the security of your Azure resources by:

- Enabling you to define policies for your Azure subscription resources according to your company's security

needs and the type of applications or sensitivity of the data in each subscription.

- Monitoring the state of your Azure virtual machines, networking, and applications.
- Providing a list of prioritized security alerts, including alerts from integrated partner solutions, along with the information you need to quickly investigate and recommendations on how to remediate an attack.

Learn more:

- [Introduction to Azure Security Center](#)

Azure Service Fabric security overview

8/29/2017 • 10 min to read • [Edit Online](#)

[Azure Service Fabric](#) is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices. Service Fabric addresses the significant challenges of developing and managing cloud applications. Developers and administrators can avoid complex infrastructure problems and focus on implementing mission-critical, demanding workloads that are scalable, reliable, and manageable.

This Azure Service Fabric Security overview article focuses on the following areas:

- Securing your cluster
- Understanding monitoring and diagnostics
- Creating more secure environments by using certificates
- Using Role-Based Access Control (RBAC)
- Securing clusters by using Windows security
- Configuring application security in Service Fabric
- Securing communication for services in Azure Service Fabric

Secure your cluster

Azure Service Fabric orchestrates services across a cluster of machines. Clusters must be secured to prevent unauthorized users from connecting to them, especially when they are running production workloads. Although it's possible to create an unsecured cluster, this might allow anonymous users to connect to it (if it exposes management endpoints to the public internet).

This section provides an overview of the security scenarios for clusters that are running either standalone or on Azure. It also describes the various technologies that are used to implement those scenarios. The cluster security scenarios are:

- Node-to-node security
- Client-to-node security

Node-to-node security

Node-to-node security secures communication between the VMs or machines in a cluster. With node-to-node security, only computers that are authorized to join the cluster can participate in hosting applications and services in the cluster.

Clusters that are running on Azure or standalone clusters that are running on Windows can use either [certificate security](#) or [Windows security](#) for Windows Server machines.

Understand node-to-node certificate security

Service Fabric uses X.509 server certificates that you specify when you create a cluster. For a quick overview of what these certificates are and how you can acquire or create them, see [Working with certificates](#).

You configure certificate security when you create the cluster, either through the Azure portal, Azure Resource Manager templates, or a standalone JSON template. You can specify a primary certificate and an optional secondary certificate that is used for certificate rollovers. The primary and secondary certificates you specify should be different than the admin client and read-only client certificates that you specify for [client-to-node security](#).

Client-to-node security

You configure client-to-node security by using client identities. To establish trust between a client and a cluster, you

must configure the cluster to know which client identities it can trust. This can be done in two different ways:

- Specify the domain group users that can connect.
- Specify the domain node users that can connect.

Service Fabric supports two different access control types for clients that are connected to a Service Fabric cluster:

- Administrator
- User

By using access control, cluster administrators can limit access to certain types of cluster operations. This makes the cluster more secure.

Administrators have full access to management capabilities (including read/write capabilities). Users, by default, have only read access to management capabilities (for example, query capabilities), and the ability to resolve applications and services.

Understand client-to-node certificate security

You configure client-to-node certificate security when you create a cluster either through the Azure portal, Resource Manager templates, or a standalone JSON template. You need to specify an admin client certificate and/or a user client certificate.

The admin client and user client certificates that you specify should be different than the primary and secondary certificates that you specify for node-to-node security.

Clients that connect to the cluster by using the admin certificate have full access to management capabilities. Clients that connect to the cluster by using the read-only user client certificate have only read access to management capabilities. In other words, these certificates are used for RBAC.

To learn how to configure certificate security in a cluster, see [Set up a cluster by using an Azure Resource Manager template](#).

Understand client-to-node Azure Active Directory security on Azure

Clusters that are running on Azure can also secure access to the management endpoints by using Azure Active Directory (Azure AD). For information about how to create the necessary Azure Active Directory artifacts, how to populate them during cluster creation, and how to connect to those clusters, see [Set up a cluster by using an Azure Resource Manager template](#).

Azure AD enables organizations (known as tenants) to manage user access to applications. There are applications with a web-based sign-in UI, and applications with a native client experience.

A Service Fabric cluster offers several entry points to its management functionality, including the web-based Service Fabric Explorer and Visual Studio. As a result, you create two Azure AD applications to control access to the cluster: one web application, and one native application.

For Azure clusters, we recommend that you use Azure AD security to authenticate clients and certificates for node-to-node security.

For standalone Windows Server clusters with Windows Server 2012 R2 and Active Directory, we recommend that you use Windows security with group managed accounts. Otherwise, use Windows security with Windows accounts.

Understand monitoring and diagnostics in Azure Service Fabric

[Monitoring and diagnostics](#) are critical to developing, testing, and deploying applications and services in any environment. Service Fabric solutions work best when you implement monitoring and diagnostics to ensure that applications and services work as expected in a local development environment or in production.

From a security perspective, the main goals of monitoring and diagnostics are:

- Detect and diagnose hardware and infrastructure issues that might be caused by a security event.
- Detect software and app issues that could be an indicator of compromise (IoC).
- Understand resource consumption to help prevent inadvertent denial of service.

The overall workflow of monitoring and diagnostics consists of three steps:

- **Event generation:** Event generation includes events (logs, traces, custom events) at both the infrastructure (cluster) and application/service level. Read more about [infrastructure-level events](#) and [application-level events](#) to understand what is provided and how to add further instrumentation.
- **Event aggregation:** Generated events need to be collected and aggregated before they can be displayed. We typically recommend using [Azure Diagnostics](#) (similar to agent-based log collection) or [EventFlow](#) (in-process log collection).
- **Analysis:** Events need to be visualized and accessible in some format, to allow for analysis and display. There are several platforms for the analysis and visualization of monitoring and diagnostics data. The two that we recommend are [Operations Management Suite](#) and [Azure Application Insights](#) due to their good integration with Service Fabric.

You can also use [Azure Monitor](#) to monitor many of the Azure resources on which a Service Fabric cluster is built.

A watchdog is a separate service that can watch health, load across services, and report health for anything in the health model hierarchy. Using a watchdog can help prevent errors that would not be detected based on the view of a single service.

Watchdogs are also a good place to host code that performs remedial actions without user interaction (for example, cleaning up log files in storage at certain time intervals). You can find a sample watchdog service implementation at [Azure Service Fabric watchdog sample](#).

Understand how to secure communication by using certificates

Certificates help you secure the communication between the various nodes of your standalone Windows cluster. By using X.509 certificates, you can also authenticate clients that are connecting to this cluster. This ensures that only authorized users can access the cluster. We recommend that you enable a certificate on the cluster when you create it.

X.509 certificates and Service Fabric

X.509 digital certificates are commonly used to authenticate clients and servers. They are also used to encrypt and digitally sign messages.

The following table lists the certificates that you need on your cluster setup:

CERTIFICATE INFORMATION SETTING	DESCRIPTION
ClusterCertificate	This certificate is required to secure the communication between the nodes on a cluster. You can use two different certificates: a primary certificate, and a secondary for upgrade.
ServerCertificate	This certificate is presented to the client when it tries to connect to this cluster. You can use two different server certificates: a primary certificate, and a secondary for upgrade.
ClientCertificateThumbprints	This is a set of certificates to install on the authenticated clients.

CERTIFICATE INFORMATION SETTING	DESCRIPTION
ClientCertificateCommonNames	This is the common name of the first client certificate for CertificateCommonName. CertificateIssuerThumbprint is the thumbprint for the issuer of this certificate.
ReverseProxyCertificate	This is an optional certificate that can be specified to secure your reverse proxy .

For more information about securing certificates, see [Secure a standalone cluster on Windows using X.509 certificates](#).

Understand Role-Based Access Control

Access control allows the cluster administrator to limit access to certain cluster operations for different groups of users, thus making the cluster more secure. Two different access control types are supported for clients that are connecting to a cluster:

- Administrator role
- User role

Administrators have full access to management capabilities (including read/write capabilities). Users, by default, have only read access to management capabilities (for example, query capabilities), and the ability to resolve applications and services.

You specify the administrator and user client roles at the time of cluster creation by providing separate identities (including certificates) for each. For more information about the default access control settings and how to change the default settings, see [Role-Based Access Control for Service Fabric clients](#).

Secure standalone cluster by using Windows security

To prevent unauthorized access to a Service Fabric cluster, you must secure the cluster. Security is especially important when the cluster runs production workloads. It describes how to configure node-to-node and client-to-node security by using Windows security in the ClusterConfig.JSON file.

Configure Windows security by using gMSA

When Service Fabric needs to run under gMSA, you configure node-to-node security by setting [ClustergMSAIdentity](#). To build trust relationships between nodes, they must be made aware of each other.

You configure client-to-node security by using ClientIdentities. To establish trust between a client and the cluster, you must configure the cluster to recognize which client identities it can trust.

Configure Windows security by using a machine group

If you want to use a machine group within an Active Directory domain, you configure node-to-node security by setting ClusterIdentity. For more information, see [Create a machine group in Active Directory](#).

You configure client-to-node security by using ClientIdentities. To establish trust between a client and the cluster, you must configure the cluster to recognize the client identities that the cluster can trust. You can establish trust in two different ways:

- Specify the domain group users that can connect.
- Specify the domain node users that can connect.

Configure application security in Service Fabric

Manage secrets in Service Fabric applications

This method helps manage secrets in a Service Fabric application. Secrets can be any sensitive information, such as storage connection strings, passwords, or other values that should not be handled in plain text.

This approach uses [Azure Key Vault](#) to manage keys and secrets. However, using secrets in an application is cloud platform-agnostic. This means that applications can be deployed to a cluster that's hosted anywhere. There are four main steps in this flow:

- Obtain a data encipherment certificate.
- Install the certificate on your cluster.
- Encrypt secret values when deploying an application with the certificate and inject them into a service's Settings.xml configuration file.
- Read encrypted values out of Settings.xml by decrypting them with the same encipherment certificate.

NOTE

Learn more about [managing secrets in Service Fabric applications](#).

Configure security policies for your application

By using Azure Service Fabric security, you can help secure applications that are running in the cluster under different user accounts. Service Fabric Security also helps secure the resources that are used by applications at the time of deployment under the user accounts--for example, files, directories, and certificates. This makes running applications, even in a shared hosted environment, more secure.

The steps include:

- Configuring the policy for a service setup entry point.
- Starting PowerShell commands from a setup entry point.
- Using console redirection for local debugging.
- Configuring a policy for service code packages.
- Assigning a security access policy for HTTP and HTTPS endpoints.

Secure communication for services in Azure Service Fabric security

Security is one of the most important aspects of communication. The Reliable Services application framework provides a few prebuilt communication stacks and tools that can be used to improve security.

- [Help secure a service when you're using service remoting](#)
- [Help secure a service when you're using a WCF-based communication stack](#)

Next steps

- For conceptual information about cluster security, see [Create a Service Fabric cluster by using Azure Resource Manager](#) and [Azure portal](#).
- To learn more about cluster security in Service Fabric, see [Service Fabric cluster security](#).

Azure identity management security overview

8/11/2017 • 7 min to read • [Edit Online](#)

Microsoft identity and access management solutions help IT protect access to applications and resources across the corporate datacenter and into the cloud, enabling additional levels of validation such as multi-factor authentication and conditional access policies. Monitoring suspicious activity through advanced security reporting, auditing and alerting helps mitigate potential security issues. [Azure Active Directory Premium](#) provides single sign-on to thousands of cloud (SaaS) apps and access to web apps you run on-premises.

Security benefits of Azure Active Directory (AD) include the ability to:

- Create and manage a single identity for each user across your hybrid enterprise, keeping users, groups, and devices in sync
- Provide single sign-on access to your applications including thousands of pre-integrated SaaS apps
- Enable application access security by enforcing rules-based Multi-Factor Authentication for both on-premises and cloud applications
- Provision secure remote access to on-premises web applications through Azure AD Application Proxy

The goal of this article is to provide an overview of the core Azure security features that help with identity management. We also provide links to articles that give details of each feature so you can learn more.

The article focuses on the following core Azure Identity management capabilities:

- Single sign-on
- Reverse proxy
- Multi-factor authentication
- Security monitoring, alerts, and machine learning-based reports
- Consumer identity and access management
- Device registration
- Privileged identity management
- Identity protection
- Hybrid identity management

Single sign-on

Single sign-on (SSO) means being able to access all the applications and resources that you need to do business, by signing in only once using a single user account. Once signed in, you can access all of the applications you need without being required to authenticate (for example, type a password) a second time.

Many organizations rely upon software as a service (SaaS) applications such as Office 365, Box and Salesforce for end user productivity. Historically, IT staff needed to individually create and update user accounts in each SaaS application, and users had to remember a password for each SaaS application.

Azure AD extends on-premises Active Directory environments into the cloud, enabling users to use their primary organizational account to not only sign in to their domain-joined devices and company resources, but also all the web and SaaS applications needed for their job.

Not only do users not have to manage multiple sets of usernames and passwords, application access can be automatically provisioned or de-provisioned based on organizational groups and their status as an employee. Azure AD introduces security and access governance controls that enable you to centrally manage users' access across SaaS applications.

Learn more:

- [Overview of Single Sign-On](#)
- [What is application access and single sign-on with Azure Active Directory?](#)
- [Integrate Azure Active Directory single sign-on with SaaS apps](#)

Reverse proxy

Azure AD Application Proxy lets you publish on-premises applications, such as [SharePoint](#) sites, [Outlook Web App](#), and [IIS](#)-based apps inside your private network and provides secure access to users outside your network.

Application Proxy provides remote access and single sign-on (SSO) for many types of on-premises web applications with the thousands of SaaS applications that Azure AD supports. Employees can log in to your apps from home on their own devices and authenticate through this cloud-based proxy.

Learn more:

- [Enabling Azure AD Application Proxy](#)
- [Publish applications using Azure AD Application Proxy](#)
- [Single-sign-on with Application Proxy](#)
- [Working with conditional access](#)

Multi-factor authentication

Azure Multi-factor authentication (MFA) is a method of authentication that requires the use of more than one verification method and adds a critical second layer of security to user sign-ins and transactions. MFA helps safeguard access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication via a range of verification options—phone call, text message, or mobile app notification or verification code and third party OAuth tokens.

Learn more:

- [Multi-factor authentication](#)
- [What is Azure Multi-Factor Authentication?](#)
- [How Azure Multi-Factor Authentication works](#)

Security monitoring, alerts, and machine learning-based reports

Security monitoring and alerts and machine learning-based reports that identify inconsistent access patterns can help you protect your business. You can use Azure Active Directory's access and usage reports to gain visibility into the integrity and security of your organization's directory. With this information, a directory admin can better determine where possible security risks may lie so that they can adequately plan to mitigate those risks.

In the Azure classic portal, reports are categorized in the following ways:

- Anomaly reports – contain sign in events that we found to be anomalous. Our goal is to make you aware of such activity and enable you to be able to make a determination about whether an event is suspicious.
- Integrated Application reports – provide insights into how cloud applications are being used in your organization. Azure Active Directory offers integration with thousands of cloud applications.
- Error reports – indicate errors that may occur when provisioning accounts to external applications.
- User-specific reports – display device/sign in activity data for a specific user.
- Activity logs – contain a record of all audited events within the last 24 hours, last 7 days, or last 30 days, and group activity changes, and password reset and registration activity.

Learn more:

- [View your access and usage reports](#)
- [Getting started with Azure Active Directory Reporting](#)
- [Azure Active Directory Reporting Guide](#)

Consumer identity and access management

Azure Active Directory B2C is a highly available, global, identity management service for consumer-facing applications that scales to hundreds of millions of identities. It can be integrated across mobile and web platforms. Your consumers can log on to all your applications through customizable experiences by using their existing social accounts or by creating new credentials.

In the past, application developers who wanted to sign up and sign in consumers into their applications would have written their own code. And they would have used on-premises databases or systems to store usernames and passwords. Azure Active Directory B2C offers your organization a better way to integrate consumer identity management into applications with the help of a secure, standards-based platform and a large set of extensible policies.

When you use Azure Active Directory B2C, your consumers can sign up for your applications by using their existing social accounts (Facebook, Google, Amazon, LinkedIn) or by creating new credentials (email address and password, or username and password).

Learn more:

- [What is Azure Active Directory B2C?](#)
- [Azure Active Directory B2C preview: Sign up and sign in consumers in your applications](#)
- [Azure Active Directory B2C Preview: Types of Applications](#)

Device registration

Azure AD Device Registration is the foundation for device-based [conditional access](#) scenarios. When a device is registered, Azure Active Directory Device Registration provides the device with an identity that is used to authenticate the device when the user signs in. The authenticated device, and the attributes of the device, can then be used to enforce conditional access policies for applications that are hosted in the cloud and on-premises.

When combined with a mobile device management (MDM) solution such as Intune, the device attributes in Azure Active Directory are updated with additional information about the device. This allows you to create conditional access rules that enforce access from devices to meet your standards for security and compliance.

Learn more:

- [Get started with Azure Active Directory Device Registration](#)
- [Automatic device registration with Azure Active Directory for Windows domain-joined devices](#)
- [Set up automatic registration of Windows domain-joined devices with Azure Active Directory](#)

Privileged identity management

Azure Active Directory (AD) Privileged Identity Management lets you manage, control, and monitor your privileged identities and access to resources in Azure AD as well as other Microsoft online services like Office 365 or Microsoft Intune.

Sometimes users need to carry out privileged operations in Azure or Office 365 resources, or other SaaS apps. This often means organizations have to give them permanent privileged access in Azure AD. This is a growing security risk for cloud-hosted resources because organizations can't sufficiently monitor what those users are doing with their admin privileges. Additionally, if a user account with privileged access is compromised, that one breach could impact their overall cloud security. Azure AD Privileged Identity Management helps to resolve this risk.

Azure AD Privileged Identity Management lets you:

- See which users are Azure AD admins
- Enable on-demand, "just in time" administrative access to Microsoft Online Services like Office 365 and Intune
- Get reports about administrator access history and changes in administrator assignments
- Get alerts about access to a privileged role

Learn more:

- [Azure AD Privileged Identity Management](#)
- [Roles in Azure AD Privileged Identity Management](#)
- [Azure AD Privileged Identity Management: How to add or remove a user role](#)

Identity protection

Azure AD Identity Protection is a security service that provides a consolidated view into risk events and potential vulnerabilities affecting your organization's identities. Identity Protection leverages existing Azure Active Directory's anomaly detection capabilities (available through Azure AD's Anomalous Activity Reports), and introduces new risk event types that can detect anomalies in real-time.

Learn more:

- [Azure Active Directory Identity Protection](#)
- [Channel 9: Azure AD and Identity Show: Identity Protection Preview](#)

Hybrid identity management

Microsoft's approach to identity spans on-premises and the cloud, creating a single user identity for authentication and authorization to all resources, regardless of location.

Learn more:

- [Hybrid identity white paper](#)
- [Azure Active Directory](#)
- [Active Directory Team Blog](#)

Internet of Things security architecture

7/3/2017 • 24 min to read • [Edit Online](#)

When designing a system, it is important to understand the potential threats to that system, and add appropriate defenses accordingly, as the system is designed and architected. It is particularly important to design the product from the start with security in mind because understanding how an attacker might be able to compromise a system helps make sure appropriate mitigations are in place from the beginning.

Security starts with a threat model

Microsoft has long used threat models for its products and has made the company's threat modeling process publicly available. The company experience demonstrates that the modeling has unexpected benefits beyond the immediate understanding of what threats are the most concerning. For example, it also creates an avenue for an open discussion with others outside the development team, which can lead to new ideas and improvements in the product.

The objective of threat modeling is to understand how an attacker might be able to compromise a system and then make sure appropriate mitigations are in place. Threat modeling forces the design team to consider mitigations as the system is designed rather than after a system is deployed. This fact is critically important, because retrofitting security defenses to a myriad of devices in the field is infeasible, error prone and will leave customers at risk.

Many development teams do an excellent job capturing the functional requirements for the system that benefit customers. However, identifying non-obvious ways that someone might misuse the system is more challenging. Threat modeling can help development teams understand what an attacker might do and why. Threat modeling is a structured process that creates a discussion about the security design decisions in the system, as well as changes to the design that are made along the way that impact security. While a threat model is simply a document, this documentation also represents an ideal way to ensure continuity of knowledge, retention of lessons learned, and help new team onboard rapidly. Finally, an outcome of threat modeling is to enable you to consider other aspects of security, such as what security commitments you wish to provide to your customers. These commitments in conjunction with threat modeling will inform and drive testing of your Internet of Things (IoT) solution.

When to threat model

[Threat modeling](#) offers the greatest value if it is incorporated into the design phase. When you are designing, you have the greatest flexibility to make changes to eliminate threats. Eliminating threats by design is the desired outcome. It is much easier than adding mitigations, testing them, and ensuring they remain current and moreover, such elimination is not always possible. It becomes harder to eliminate threats as a product becomes more mature, and in turn will ultimately require more work and a lot harder tradeoffs than threat modeling early on in the development.

What to threat model

You should threat model the solution as a whole and also focus in the following areas:

- The security and privacy features
- The features whose failures are security relevant
- The features that touch a trust boundary

Who threat models

Threat modeling is a process like any other. It is a good idea to treat the threat model document like any other component of the solution and validate it. Many development teams do an excellent job capturing the functional requirements for the system that benefit customers. However, identifying non-obvious ways that someone might

misuse the system is more challenging. Threat modeling can help development teams understand what an attacker might do and why.

How to threat model

The threat modeling process is composed of four steps; the steps are:

- Model the application
- Enumerate Threats
- Mitigate threats
- Validate the mitigations

The process steps

Three rules of thumb to keep in mind when building a threat model:

1. Create a diagram out of reference architecture.
2. Start breadth-first. Get an overview, and understand the system as a whole, before deep-diving. This helps ensure that you deep-dive in the right places.
3. Drive the process, don't let the process drive you. If you find an issue in the modeling phase and want to explore it, go for it! Don't feel you need to follow these steps slavishly.

Threats

The four core elements of a threat model are:

- Processes (web services, Win32 services, *nix daemons, etc. Note that some complex entities (for example field gateways and sensors) can be abstracted as a process when a technical drill-down in these areas is not possible.)
- Data stores (anywhere data is stored, such as a configuration file or database)
- Data flow (where data moves between other elements in the application)
- External Entities (anything that interacts with the system, but is not under the control of the application, examples include users and satellite feeds)

All elements in the architectural diagram are subject to various threats; we will use the STRIDE mnemonic. Read [Threat Modeling Again, STRIDE](#) to know more about the STRIDE elements.

Different elements of the application diagram are subject to certain STRIDE threats:

- Processes are subject to STRIDE
- Data flows are subject to TID
- Data stores are subject to TID, and sometimes R, if the data stores are log files.
- External entities are subject to SRD

Security in IoT

Connected special-purpose devices have a significant number of potential interaction surface areas and interaction patterns, all of which must be considered to provide a framework for securing digital access to those devices. The term "digital access" is used here to distinguish from any operations that are carried out through direct device interaction where access security is provided through physical access control. For example, putting the device into a room with a lock on the door. While physical access cannot be denied using software and hardware, measures can be taken to prevent physical access from leading to system interference.

As we explore the interaction patterns, we will look at "device control" and "device data" with the same level of attention. "Device control" can be classified as any information that is provided to a device by any party with the goal of changing or influencing its behavior towards its state or the state of its environment. "Device data" can be classified as any information that a device emits to any other party about its state and the observed state of its environment.

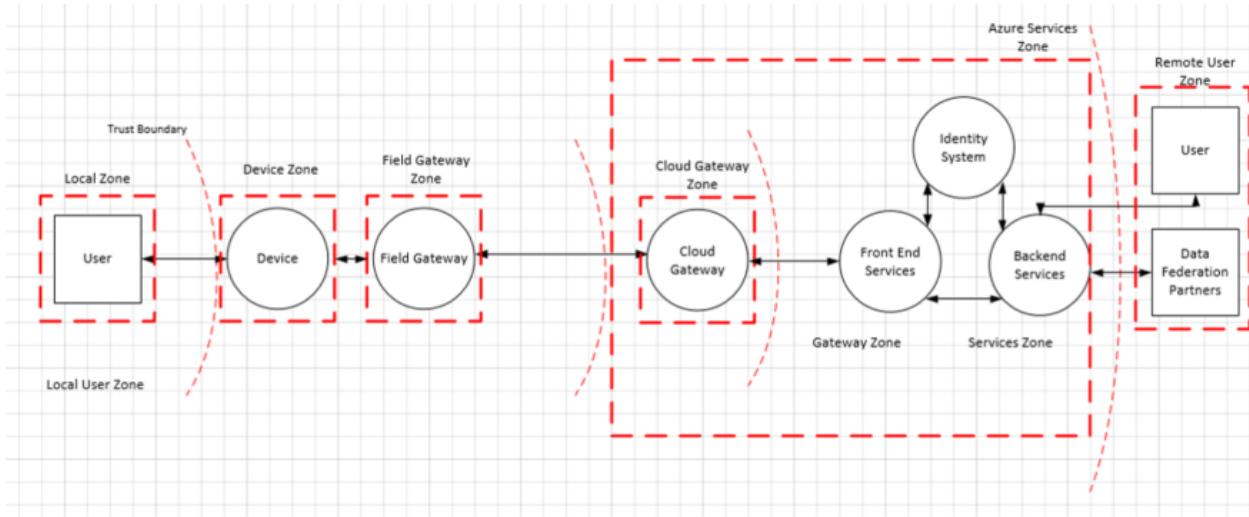
In order to optimize security best practices, it is recommended that a typical IoT architecture be divided into several

component/zones as part of the threat modeling exercise. These zones are described fully throughout this section and include:

- Device,
- Field Gateway,
- Cloud gateways, and
- Services.

Zones are broad way to segment a solution; each zone often has its own data and authentication and authorization requirements. Zones can also be used to isolation damage and restrict the impact of low trust zones on higher trust zones.

Each zone is separated by a Trust Boundary, which is noted as the dotted red line in the diagram below. It represents a transition of data/information from one source to another. During this transition, the data/information could be subject to Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service and Elevation of Privilege (STRIDE).



The components depicted within each boundary are also subjected to STRIDE, enabling a full 360 threat modeling view of the solution. The sections below elaborate on each of the components and specific security concerns and solutions that should be put into place.

The sections that follow will discuss standard components typically found in these zones.

The Device Zone

The device environment is the immediate physical space around the device where physical access and/or "local network" peer-to-peer digital access to the device is feasible. A "local network" is assumed to be a network that is distinct and insulated from – but potentially bridged to – the public Internet, and includes any short-range wireless radio technology that permits peer-to-peer communication of devices. It does *not* include any network virtualization technology creating the illusion of such a local network and it does also not include public operator networks that require any two devices to communicate across public network space if they were to enter a peer-to-peer communication relationship.

The Field Gateway Zone

Field gateway is a device/appliance or some general-purpose server computer software that acts as communication enabler and, potentially, as a device control system and device data processing hub. The field gateway zone includes the field gateway itself and all devices that are attached to it. As the name implies, field gateways act outside dedicated data processing facilities, are usually location bound, are potentially subject to physical intrusion, and will have limited operational redundancy. All to say that a field gateway is commonly a thing one can touch and sabotage while knowing what its function is.

A field gateway is different from a mere traffic router in that it has had an active role in managing access and information flow, meaning it is an application addressed entity and network connection or session terminal. An NAT device or firewall, in contrast, does not qualify as field gateways since they are not explicit connection or session terminals, but rather a route (or block) connections or sessions made through them. The field gateway has two distinct surface areas. One faces the devices that are attached to it and represents the inside of the zone, and the other faces all external parties and is the edge of the zone.

The cloud gateway zone

Cloud gateway is a system that enables remote communication from and to devices or field gateways from several different sites across public network space, typically towards a cloud-based control and data analysis system, a federation of such systems. In some cases, a cloud gateway may immediately facilitate access to special-purpose devices from terminals such as tablets or phones. In the context discussed here, "cloud" is meant to refer to a dedicated data processing system that is not bound to the same site as the attached devices or field gateways. Also in a Cloud Zone, operational measures prevent targeted physical access and are not necessarily exposed to a "public cloud" infrastructure.

A cloud gateway may potentially be mapped into a network virtualization overlay to insulate the cloud gateway and all of its attached devices or field gateways from any other network traffic. The cloud gateway itself is neither a device control system nor a processing or storage facility for device data; those facilities interface with the cloud gateway. The cloud gateway zone includes the cloud gateway itself along with all field gateways and devices directly or indirectly attached to it. The edge of the zone is a distinct surface area where all external parties communicate through.

The services zone

A "service" is defined for this context as any software component or module that is interfacing with devices through a field- or cloud gateway for data collection and analysis, as well as for command and control. Services are mediators. They act under their identity towards gateways and other subsystems, store and analyze data, autonomously issue commands to devices based on data insights or schedules and expose information and control capabilities to authorized end users.

Information-devices vs. special-purpose devices

PCs, phones, and tablets are primarily interactive information devices. Phones and tablets are explicitly optimized around maximizing battery lifetime. They preferably turn off partially when not immediately interacting with a person, or when not providing services like playing music or guiding their owner to a particular location. From a systems perspective, these information technology devices are mainly acting as proxies towards people. They are "people actuators" suggesting actions and "people sensors" collecting input.

Special-purpose devices, from simple temperature sensors to complex factory production lines with thousands of components inside them, are different. These devices are much more scoped in purpose and even if they provide some user interface, they are largely scoped to interfacing with or be integrated into assets in the physical world. They measure and report environmental circumstances, turn valves, control servos, sound alarms, switch lights, and do many other tasks. They help to do work for which an information device is either too generic, too expensive, too big, or too brittle. The concrete purpose immediately dictates their technical design as well the available monetary budget for their production and scheduled lifetime operation. The combination of these two key factors constrains the available operational energy budget, physical footprint, and thus available storage, compute, and security capabilities.

If something "goes wrong" with automated or remote controllable devices, for example, physical defects or control logic defects to willful unauthorized intrusion and manipulation. The production lots may be destroyed, buildings may be looted or burned down, and people may be injured or even die. This is, of course, a whole different class of damage than someone maxing out a stolen credit card's limit. The security bar for devices that make things move, and also for sensor data that eventually results in commands that cause things to move, must be higher than in any e-commerce or banking scenario.

Device control and device data interactions

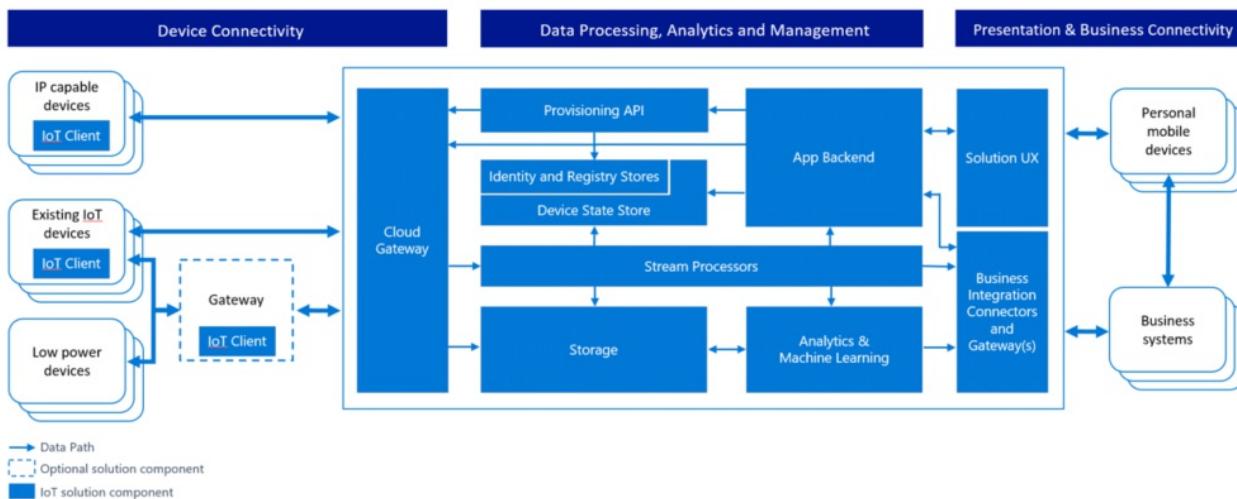
Connected special-purpose devices have a significant number of potential interaction surface areas and interaction patterns, all of which must be considered to provide a framework for securing digital access to those devices. The term "digital access" is used here to distinguish from any operations that are carried out through direct device interaction where access security is provided through physical access control. For example, putting the device into a room with a lock on the door. While physical access cannot be denied using software and hardware, measures can be taken to prevent physical access from leading to system interference.

As we explore the interaction patterns, we will look at "device control" and "device data" with the same level of attention while threat modeling. "Device control" can be classified as any information that is provided to a device by any party with the goal of changing or influencing its behavior towards its state or the state of its environment. "Device data" can be classified as any information that a device emits to any other party about its state and the observed state of its environment.

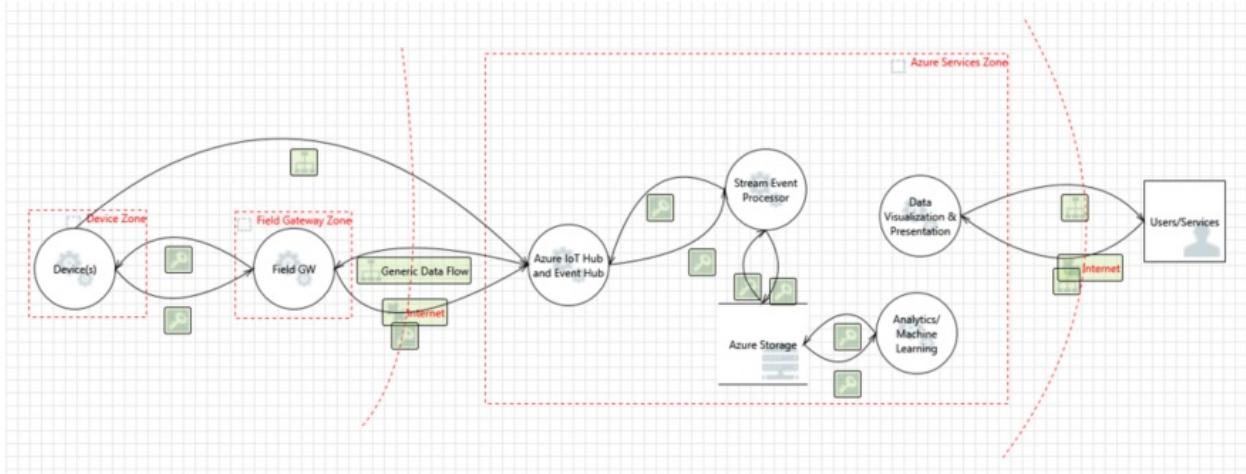
Threat modeling the Azure IoT reference architecture

Microsoft uses the framework outlined above to do threat modeling for Azure IoT. In the section below we therefore use the concrete example of Azure IoT Reference Architecture to demonstrate how to think about threat modeling for IoT and how to address the threats identified. In our case we identified four main areas of focus:

- Devices and Data Sources,
- Data Transport,
- Device and Event Processing, and
- Presentation



The diagram below provides a simplified view of Microsoft's IoT Architecture using a Data Flow Diagram model that is used by the Microsoft Threat Modeling Tool:



It is important to note that the architecture separates the device and gateway capabilities. This allows the user to leverage gateway devices that are more secure: they are capable of communicating with the cloud gateway using secure protocols, which typically requires greater processing overhead than a native device - such as a thermostat - could provide on its own. In the Azure services zone, we assume that the Cloud Gateway is represented by the Azure IoT Hub service.

Device and data sources/data transport

This section explores the architecture outlined above through the lens of threat modeling and gives an overview of how we are addressing some of the inherent concerns. We will focus on the core elements of a threat model:

- Processes (those under our control and external items)
- Communication (also called data flows)
- Storage (also called data stores)

Processes

In each of the categories outlined in the Azure IoT architecture, we try to mitigate a number of different threats across the different stages data/information exists in: process, communication, and storage. Below we give an overview of the most common ones for the "process" category, followed by an overview of how these could be best mitigated:

Spoofing (S): An attacker may extract cryptographic key material from a device, either at the software or hardware level, and subsequently access the system with a different physical or virtual device under the identity of the device the key material has been taken from. A good illustration is remote controls that can turn any TV and that are popular prankster tools.

Denial of Service (D): A device can be rendered incapable of functioning or communicating by interfering with radio frequencies or cutting wires. For example, a surveillance camera that had its power or network connection intentionally knocked out will not report data, at all.

Tampering (T): An attacker may partially or wholly replace the software running on the device, potentially allowing the replaced software to leverage the genuine identity of the device if the key material or the cryptographic facilities holding key materials were available to the illicit program. For example, an attacker may leverage extracted key material to intercept and suppress data from the device on the communication path and replace it with false data that is authenticated with the stolen key material.

Information Disclosure (I): If the device is running manipulated software, such manipulated software could potentially leak data to unauthorized parties. For example, an attacker may leverage extracted key material to inject itself into the communication path between the device and a controller or field gateway or cloud gateway to siphon off information.

Elevation of Privilege (E): A device that does specific function can be forced to do something else. For example, a valve that is programmed to open half way can be tricked to open all the way.

COMPONENT	THREAT	MITIGATION	RISK	IMPLEMENTATION
Device	S	Assigning identity to the device and authenticating the device	Replacing device or part of the device with some other device. How do we know we are talking to the right device?	Authenticating the device, using Transport Layer Security (TLS) or IPSec. Infrastructure should support using pre-shared key (PSK) on those devices that cannot handle full asymmetric cryptography. Leverage Azure AD, OAuth
TRID	Apply tamperproof mechanisms to the device for example by making it very hard to impossible to extract keys and other cryptographic material from the device.	The risk is if someone is tampering the device (physical interference). How are we sure, that device has not tampered with.	The most effective mitigation is a trusted platform module (TPM) capability that allows storing keys in special on-chip circuitry from which the keys cannot be read, but can only be used for cryptographic operations that use the key but never disclose the key. Memory encryption of the device. Key management for the device. Signing the code.	
E	Having access control of the device. Authorization scheme.	If the device allows for individual actions to be performed based on commands from an outside source, or even compromised sensors, it will allow the attack to perform operations not otherwise accessible.	Having authorization scheme for the device	
Field Gateway	S	Authenticating the Field gateway to Cloud Gateway (cert based, PSK, Claim based,..)	If someone can spoof Field Gateway, then it can present itself as any device.	TLS RSA/PSK, IPSec, RFC 4279 . All the same key storage and attestation concerns of devices in general – best case is use TPM. 6LowPAN extension for IPSec to support Wireless Sensor Networks (WSN).

COMPONENT	THREAT	MITIGATION	RISK	IMPLEMENTATION
TRID	Protect the Field Gateway against tampering (TPM?)	Spoofing attacks that trick the cloud gateway thinking it is talking to field gateway could result in information disclosure and data tampering	Memory encryption, TPM's, authentication.	
E	Access control mechanism for Field Gateway			

Here are some examples of threats in this category:

Spoofing: An attacker may extract cryptographic key material from a device, either at the software or hardware level, and subsequently access the system with a different physical or virtual device under the identity of the device the key material has been taken from.

Denial of Service: A device can be rendered incapable of functioning or communicating by interfering with radio frequencies or cutting wires. For example, a surveillance camera that had its power or network connection intentionally knocked out will not report data, at all.

Tampering: An attacker may partially or wholly replace the software running on the device, potentially allowing the replaced software to leverage the genuine identity of the device if the key material or the cryptographic facilities holding key materials were available to the illicit program.

Tampering: A surveillance camera that's showing a visible-spectrum picture of an empty hallway could be aimed at a photograph of such a hallway. A smoke or fire sensor could be reporting someone holding a lighter under it. In either case, the device may be technically fully trustworthy towards the system, but it will report manipulated information.

Tampering: An attacker may leverage extracted key material to intercept and suppress data from the device on the communication path and replace it with false data that is authenticated with the stolen key material.

Tampering: An attacker may partially or completely replace the software running on the device, potentially allowing the replaced software to leverage the genuine identity of the device if the key material or the cryptographic facilities holding key materials were available to the illicit program.

Information Disclosure: If the device is running manipulated software, such manipulated software could potentially leak data to unauthorized parties.

Information Disclosure: An attacker may leverage extracted key material to inject itself into the communication path between the device and a controller or field gateway or cloud gateway to siphon off information.

Denial of Service: The device can be turned off or turned into a mode where communication is not possible (which is intentional in many industrial machines).

Tampering: The device can be reconfigured to operate in a state unknown to the control system (outside of known calibration parameters) and thus provide data that can be misinterpreted

Elevation of Privilege: A device that does specific function can be forced to do something else. For example, a valve that is programmed to open half way can be tricked to open all the way.

Denial of Service: The device can be turned into a state where communication is not possible.

Tampering: The device can be reconfigured to operate in a state unknown to the control system (outside of known

calibration parameters) and thus provide data that can be misinterpreted.

Spoofing/Tampering/Repudiation: If not secured (which is rarely the case with consumer remote controls) an attacker can manipulate the state of a device anonymously. A good illustration is remote controls that can turn any TV and that are popular prankster tools.

Communication

Threats around communication path between devices, devices and field gateways and device and cloud gateway.

The table below has some guidance around open sockets on the device/VPN:

COMPONENT	THREAT	MITIGATION	RISK	IMPLEMENTATION
Device IoT Hub	TID	(D)TLS (PSK/RSA) to encrypt the traffic	Eavesdropping or interfering the communication between the device and the gateway	Security on the protocol level. With custom protocols, we need to figure out how to protect them. In most cases, the communication takes place from the device to the IoT Hub (device initiates the connection).
Device Device	TID	(D)TLS (PSK/RSA) to encrypt the traffic.	Reading data in transit between devices. Tampering with the data. Overloading the device with new connections	Security on the protocol level (MQTT/AMQP/HTTP/CoAP). With custom protocols, we need to figure out how to protect them. The mitigation for the DoS threat is to peer devices through a cloud or field gateway and have them only act as clients towards the network. The peering may result in a direct connection between the peers after having been brokered by the gateway
External Entity Device	TID	Strong pairing of the external entity to the device	Eavesdropping the connection to the device. Interfering the communication with the device	Securely pairing the external entity to the device NFC/Bluetooth LE. Controlling the operational panel of the device (Physical)
Field Gateway Cloud Gateway	TID	TLS (PSK/RSA) to encrypt the traffic.	Eavesdropping or interfering the communication between the device and the gateway	Security on the protocol level (MQTT/AMQP/HTTP/CoAP). With custom protocols, we need to figure out how to protect them.

COMPONENT	THREAT	MITIGATION	RISK	IMPLEMENTATION
Device Cloud Gateway	TID	TLS (PSK/RSA) to encrypt the traffic.	Eavesdropping or interfering the communication between the device and the gateway	Security on the protocol level (MQTT/AMQP/HTTP/CoAP). With custom protocols, we need to figure out how to protect them.

Here are some examples of threats in this category:

Denial of Service: Constrained devices are generally under DoS threat when they actively listen for inbound connections or unsolicited datagrams on a network, because an attacker can open many connections in parallel and not service them or service them very slowly, or the device can be flooded with unsolicited traffic. In both cases, the device can effectively be rendered inoperable on the network.

Spoofing, Information Disclosure: Constrained devices and special-purpose devices often have one-for-all security facilities like password or PIN protection, or they wholly rely on trusting the network, meaning they will grant access to information when a device is on the same network, and that network is often only protected by a shared key. That means that when the shared secret to device or network is disclosed, it is possible to control the device or observe data emitted from the device.

Spoofing: an attacker may intercept or partially override the broadcast and spoof the originator (man in the middle)

Tampering: an attacker may intercept or partially override the broadcast and send false information

Information Disclosure: an attacker may eavesdrop on a broadcast and obtain information without authorization

Denial of Service: an attacker may jam the broadcast signal and deny information distribution

Storage

Every device and field gateway has some form of storage (temporary for queuing the data, operating system (OS) image storage).

COMPONENT	THREAT	MITIGATION	RISK	IMPLEMENTATION
Device storage	TRID	Storage encryption, signing the logs	Reading data from the storage (PII data), tampering with telemetry data. Tampering with queued or cached command control data. Tampering with configuration or firmware update packages while cached or queued locally can lead to OS and/or system components being compromised	Encryption, message authentication code (MAC) or digital signature. Where possible, strong access control through resource access control lists (ACLs) or permissions.
Device OS image	TRID		Tampering with OS /replacing the OS components	Read-only OS partition, signed OS image, Encryption

COMPONENT	THREAT	MITIGATION	RISK	IMPLEMENTATION
Field Gateway storage (queuing the data)	TRID	Storage encryption, signing the logs	Reading data from the storage (PII data), tampering with telemetry data, tampering with queued or cached command control data. Tampering with configuration or firmware update packages (destined for devices or field gateway) while cached or queued locally can lead to OS and/or system components being compromised	BitLocker
Field Gateway OS image	TRID		Tampering with OS /replacing the OS components	Read-only OS partition, signed OS image, Encryption

Device and event processing/cloud gateway zone

A cloud gateway is system that enables remote communication from and to devices or field gateways from several different sites across public network space, typically towards a cloud-based control and data analysis system, a federation of such systems. In some cases, a cloud gateway may immediately facilitate access to special-purpose devices from terminals such as tablets or phones. In the context discussed here, "cloud" is meant to refer to a dedicated data processing system that is not bound to the same site as the attached devices or field gateways, and where operational measures prevent targeted physical access but is not necessarily to a "public cloud" infrastructure. A cloud gateway may potentially be mapped into a network virtualization overlay to insulate the cloud gateway and all of its attached devices or field gateways from any other network traffic. The cloud gateway itself is neither a device control system nor a processing or storage facility for device data; those facilities interface with the cloud gateway. The cloud gateway zone includes the cloud gateway itself along with all field gateways and devices directly or indirectly attached to it.

Cloud gateway is mostly custom built piece of software running as a service with exposed endpoints to which field gateway and devices connect. As such it must be designed with security in mind. Please follow [SDL](#) process for designing and building this service.

Services zone

A control system (or controller) is a software solution that interfaces with a device, or a field gateway, or cloud gateway for the purpose of controlling one or multiple devices and/or to collect and/or store and/or analyze device data for presentation, or subsequent control purposes. Control systems are the only entities in the scope of this discussion that may immediately facilitate interaction with people. The exception is intermediate physical control surfaces on devices, like a switch that allows a person to turn the device off or change other properties, and for which there is no functional equivalent that can be accessed digitally.

Intermediate physical control surfaces are those where any sort of governing logic constrains the function of the physical control surface such that an equivalent function can be initiated remotely or input conflicts with remote input can be avoided – such intermediated control surfaces are conceptually attached to a local control system that leverages the same underlying functionality as any other remote control system that the device may be attached to in parallel. Top threats to the cloud computing can be read at [Cloud Security Alliance \(CSA\)](#) page.

Additional resources

Refer to the following articles for additional information:

- [SDL Threat Modeling Tool](#)
- [Microsoft Azure IoT reference architecture](#)

See also

To learn more about securing your IoT solution, see [Secure your IoT deployment](#).

You can also explore some of the other features and capabilities of the IoT Suite preconfigured solutions:

- [Predictive maintenance preconfigured solution overview](#)
- [Frequently asked questions for IoT Suite](#)

You can read about IoT Hub security in [Control access to IoT Hub](#) in the IoT Hub developer guide.

Azure encryption overview

8/22/2017 • 11 min to read • [Edit Online](#)

This article provides an overview of how encryption is used in Microsoft Azure. It covers the major areas of encryption, including encryption at rest, encryption in flight, and key management with Key Vault. Each section includes links for more detailed information.

Encryption of data at rest

Data at rest includes information that resides in persistent storage on physical media, in any digital format. This includes files on magnetic or optical media, archived data, and data backups. Microsoft Azure offers a variety of data storage solutions to meet different needs, including file, disk, blob, and table storage. Microsoft also provides encryption to protect [Azure SQL Database](#), [CosmosDB](#), and Azure Data Lake.

Data encryption at rest is available for services across the Azure Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS) cloud models. This document summarizes and provides resources to help you use Azure's encryption options.

For more detailed discussion of how data at rest is encrypted in Azure, see the document titled [Azure Data Encryption-at-Rest](#)

Azure Encryption models

Azure supports various encryption models, including server-side encryption using service-managed keys, using customer-managed keys in Azure Key Vault, or using customer-managed keys on customer-controlled hardware. Client-side encryption allows you to manage and store keys on-premises or in another secure location.

Client-side encryption

Client-side encryption is performed outside of Azure. Client-side encryption includes:

- Data encrypted by an application that's running in the customer's data center or by a service application
- Data that is already encrypted when it is received by Azure.

With client-side encryption the cloud service provider doesn't have access to the encryption keys and cannot decrypt this data. You maintain complete control of the keys.

Server-side encryption

The three server-side encryption models offer different key management characteristics, which can be chosen per your requirements.

- **Service-managed keys** provide a combination of control and convenience with low overhead
- **Customer-managed keys** give you control over the keys, including the ability to bring your own keys (BYOK) or to generate new ones.
- **Service-managed keys in customer-controlled hardware** enables you to manage keys in your proprietary repository that is outside of Microsoft's control. This is called Host Your Own Key (HYOK). However, configuration is complex, and most Azure services don't support this model.

Azure Disk Encryption

Windows and Linux virtual machines can be protected using [Azure Disk Encryption](#), which uses the [Windows BitLocker](#) technology and Linux [DM-Crypt](#) to protect both operating system disks and data disks with full volume

encryption.

Encryption keys and secrets are safeguarded in your [Azure Key Vault](#) subscription. You can back up and restore encrypted VMs that are encrypted with the KEK configuration using the Azure Backup service.

Azure Storage service encryption

Data at rest in Azure storage (both Blob and File) can be encrypted in both server-side and client-side scenarios.

[Azure Storage Service Encryption](#) (SSE) can automatically encrypt data before it is stored and automatically decrypts it when you retrieve it, making the process completely transparent users. Storage Service Encryption uses 256-bit [AES encryption](#), which is one of the strongest block ciphers available, and handles encryption, decryption, and key management in a transparent fashion.

Client-side encryption of Azure blobs

Client-side encryption of Azure blobs can be performed in different ways.

You can use the Azure Storage Client Library for .NET NuGet package to encrypt data within your client applications prior to uploading it to Azure Storage.

To learn more about and download the Azure Storage Client Library for .NET NuGet package, see the document titled [Windows Azure Storage 8.3.0](#)

When you use client-side encryption with Azure Key Vault, your data is encrypted using a one-time symmetric Content Encryption Key (CEK) that is generated by the Azure Storage client SDK. The CEK is encrypted using a Key Encryption Key (KEK), which can be either a symmetric key or an asymmetric key pair. You can manage it locally or store it in Azure Key Vault. The encrypted data is then uploaded to Azure Storage service.

To learn more about client-side encryption with Azure Key Vault and get started with how-to instructions, see the document titled [Tutorial: Encrypt and decrypt blobs in Microsoft Azure Storage using Azure Key Vault](#)

Finally, you can also use the Azure Storage Client Library for Java to perform client-side encryption before uploading data to Azure Storage, and to decrypt the data when downloading it to the client. This library also supports integration with [Azure Key Vault](#) for storage account key management.

Encryption of data at rest with Azure SQL database

[Azure SQL Database](#) is a general-purpose relational database service in Microsoft Azure that supports structures such as relational data, JSON, spatial, and XML. Azure SQL supports both server-side encryption via the Transparent Data Encryption (TDE) feature and client-side encryption via the Always Encrypted feature.

Transparent data encryption

TDE [Transparent data encryption](#) is used to encrypt [SQL Server](#), [Azure SQL Database](#), and [Azure SQL Data Warehouse](#) data files in real time, using a database encryption key (DEK), which is stored in the database boot record for availability during recovery.

TDE protects data and log files, using AES and 3DES encryption algorithms. Encryption of the database file is performed at the page level; the pages in an encrypted database are encrypted before they are written to disk and are decrypted when they're read into memory. TDE is now enabled by default on newly created Azure SQL databases.

Always encrypted

The [Always Encrypted](#) feature in Azure SQL enables you to encrypt data within client applications prior to storing in Azure SQL Database and allows you to enable delegation of on-premises database administration to third parties and maintain separation between those who own and can view the data and those who manage it but should not have access to it.

Cell/Column Level Encryption

Azure SQL Database enables you to apply symmetric encryption to a column of data using Transact-SQL. This is called [cell level encryption or column level encryption](#) (CLE), because you can use it to encrypt specific columns or

even specific cells of data with different encryption keys. This gives you more granular encryption capability than TDE, which encrypts data in pages.

CLE has built-in functions that you can use to encrypt data using either symmetric or asymmetric keys, with the public key of a certificate, or with a passphrase using 3DES.

Cosmos DB database encryption

[Azure Cosmos DB](#) is Microsoft's globally distributed, multi-model database. User data stored in Cosmos DB in non-volatile storage (solid-state drives) is encrypted by default; there are no controls to turn it on or off. Encryption at rest is implemented by using a number of security technologies, including secure key storage systems, encrypted networks, and cryptographic APIs. Encryption keys are managed by Microsoft and are rotated per Microsoft's internal guidelines.

At-rest Encryption in Azure Data Lake

[Azure Data Lake](#) is an enterprise-wide repository of every type of data collected in a single place prior to any formal definition of requirements or schema. Azure Data Lake Store supports "on by default," transparent encryption of data at rest, which is set up during the creation of your account. By default, Data Lake Store manages the keys for you, but you have the option to manage them yourself.

Three types of keys are used in encrypting and decrypting data: the Master Encryption Key (MEK), Data Encryption Key (DEK), and Block Encryption Key (BEK). The MEK is used to encrypt the DEK, which is stored on persistent media, and the BEK is derived from the DEK and the data block. If you are managing your own keys, you can rotate the MEK.

Encryption of data in transit

Azure offers many mechanisms for keeping data private as it moves from one location to another.

TLS/SSL encryption in Azure

Microsoft uses the [Transport Layer Security](#) (TLS) protocol to protect data when it's traveling between the cloud services and customers. Microsoft's data centers negotiate a TLS connection with client systems that connect to Azure services. TLS provides strong authentication, message privacy, and integrity (enabling detection of message tampering, interception, and forgery), interoperability, algorithm flexibility, ease of deployment and use.

[Perfect Forward Secrecy](#) (PFS) protects connections between customers' client systems and Microsoft's cloud services by unique keys. Connections also use RSA-based 2,048-bit encryption key lengths. This combination makes it difficult for someone to intercept and access data that is in-transit.

Azure Storage transactions

When you interact with Azure Storage through the Azure portal, all transactions take place over HTTPS. You can also use the Storage REST API over HTTPS to interact with Azure Storage. You can enforce the use of HTTPS when calling the REST APIs to access objects in storage accounts by enabling Secure transfer required for the storage account.

Shared Access Signatures ([SAS](#)), which can be used to delegate access to Azure Storage objects, include an option to specify that only the HTTPS protocol can be used when using Shared Access Signatures. This ensures that anybody sending out links with SAS tokens uses the proper protocol.

[SMB 3.0](#) used to access Azure File Shares supports encryption, and it's available in Windows Server 2012 R2, Windows 8, Windows 8.1, and Windows 10, allowing cross-region access, and even access on the desktop.

Client-side encryption encrypts the data before it's sent to Azure Storage, so that it's encrypted as it travels across the network.

SMB Encryption over Azure Virtual Networks

[SMB 3.0](#) in Azure VMs running Windows Server 2012 and above gives you the ability to make data transfers secure

by encrypting data in transit over Azure Virtual Networks, to protect against tampering and eavesdropping attacks. Administrators can enable SMB Encryption for the entire server, or just specific shares.

By default, once SMB Encryption is turned on for a share or server, only SMB 3 clients are allowed to access the encrypted shares.

In-transit Encryption in Azure Virtual Machines

Data in transit to, from, and between Azure VMs running Windows is encrypted in a number of ways, depending on the nature of the connection.

RDP sessions

You can connect and log on to an Azure VM using the [Remote Desktop Protocol](#) (RDP) from a Windows client computer, or from a Mac with an RDP client installed. Data in transit over the network in RDP sessions can be protected by TLS.

You can also use Remote Desktop to connect to a Linux VM in Azure.

Secure access to Linux VMs with SSH

You can use [Secure Shell](#) (SSH) to connect to Linux VMs running in Azure for remote management. SSH is an encrypted connection protocol that allows secure logins over unsecured connections. It is the default connection protocol for Linux VMs hosted in Azure. By using SSH keys for authentication, you eliminate the need for passwords to log in. SSH uses a public/private key pair (asymmetric encryption) for authentication.

Azure VPN encryption

You can connect to Azure through a virtual private network that creates a secure tunnel to protect the privacy of the data being sent across the network.

Azure VPN Gateway

[Azure VPN gateway](#) can be used to send encrypted traffic between your virtual network and your on-premises location across a public connection, or to send traffic between virtual networks.

Site-to-site VPN uses [IPsec](#) for transport encryption. Azure VPN gateways use a set of default proposals. You can configure Azure VPN gateways to use a custom IPsec/IKE policy with specific cryptographic algorithms and key strengths, rather than the Azure default policy sets.

Point-to-site VPN

Point-to-Site VPNs allow individual client computers access to an Azure Virtual Network. [The Secure Socket Tunneling Protocol](#) (SSTP) is used to create the VPN tunnel and can traverse firewalls (the tunnel appears as an HTTPS connection). You can use your own internal PKI root CA for point-to-site connectivity.

You can configure a point-to-site VPN connection to a virtual network using the Azure portal with certificate authentication or PowerShell.

To learn more about point-to-site VPN connections to Azure VNets, see: [Configure a Point-to-Site connection to a VNet using certification authentication: Azure portal](#) and

[Configure a Point-to-Site connection to a VNet using certificate authentication: PowerShell](#)

Site-to-site VPN

A Site-to-Site VPN gateway connection is used to connect your on-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. This type of connection requires a VPN device located on-premises that has an externally facing public IP address assigned to it.

You can configure a site-to-site VPN connection to a virtual network using the Azure portal, PowerShell, or the Azure Command Line Interface (CLI).

Read these for more info:

[Create a Site-to-Site connection in the Azure portal](#)

[Create a Site-to-Site connection](#)

[Create a virtual network with a site-to-site VPN connection using CLI](#)

In-transit Encryption in Azure Data Lake

Data in transit (also known as data in motion) is also always encrypted in Data Lake Store. In addition to encrypting data prior to storing to persistent media, the data is also always secured in transit by using HTTPS. HTTPS is the only protocol that is supported for the Data Lake Store REST interfaces.

To learn more about encryption of data in transit in Azure Data Lake, see the document titled [Encryption of data in Azure Data Lake Store](#).

Key management with Azure Key Vault

Without proper protection and management of the keys, encryption is rendered useless. Azure Key Vault is Microsoft's recommended solution for managing and controlling access to encryption keys used by cloud services. Permissions to access keys can be assigned to services or to users through Azure Active Directory accounts.

Azure Key Vault relieves organizations of the need to configure, patch, and maintain Hardware Security Modules (HSMs) and key management software. With Azure Key Vault, Microsoft never sees your keys and applications don't have direct access to them; you maintain control. You can also import or generate keys in HSMs.

Next steps

- [Azure security overview](#)
- [Azure network security overview](#)
- [Azure database security overview](#)
- [Azure virtual machines security overview](#)
- [Data encryption at rest](#)
- [Data security and encryption best practices](#)

Security architecture overview

6/27/2017 • 1 min to read • [Edit Online](#)

Having a strong architectural foundation is one of the keys to success when it comes to secure solution deployments in Azure. With this knowledge you're able to better understand your requirements by knowing the right questions to ask and more equipped to find the right answers to your questions. Getting right answers to the right questions goes a long way toward optimizing the security of your deployments.

In this section you'll see articles on Azure Security Architecture that will help you build secure solutions. A popular collection of Azure security best practices and patterns is also included. At this time, we have the following articles – make sure to visit our site and the Azure Security Team blog for updates on a regular basis:

- [Data Classification for Cloud Readiness](#)
- [Application Architecture on Microsoft Azure](#)
- [Azure Security Best Practices and Patterns](#)

Azure Operational Security

7/19/2017 • 19 min to read • [Edit Online](#)

Introduction

Overview

We know that security is job one in the cloud and how important it is that you find accurate and timely information about Azure security. One of the best reasons to use Azure for your applications and services is to take advantage of the wide array of security tools and capabilities available. These tools and capabilities help make it possible to create secure solutions on the secure Azure platform. Windows Azure must provide confidentiality, integrity, and availability of customer data, while also enabling transparent accountability.

To help customers better understand the array of security controls implemented within Microsoft Azure from both the customer's and Microsoft operational perspectives, this white paper, "Azure Operational Security", is written that provides a comprehensive look at the operational security available with Windows Azure.

Azure Platform

Azure is a public cloud service platform that supports a broad selection of operating systems, programming languages, frameworks, tools, databases, and devices. It can run Linux containers with Docker integration; build apps with JavaScript, Python, .NET, PHP, Java, and Node.js; build back-ends for iOS, Android, and Windows devices. Azure Cloud service supports the same technologies millions of developers and IT professionals already rely on and trust.

When you build on, or migrate IT assets to, a public cloud service provider you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security requirements. In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your organization's deployments. This document will help you understand how Azure security capabilities can help you fulfill these requirements.

Abstract

Azure Operational Security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure. Azure Operational Security is built on a framework that incorporates the knowledge gained through various capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Center program, and deep awareness of the cybersecurity threat landscape.

This white paper outlines Microsoft's approach to Azure Operational Security within the Microsoft Azure cloud platform and covers following services:

1. [Azure Operations Management Suite](#)
2. [Azure Security Center](#)
3. [Azure Monitor](#)
4. [Azure Network Watcher](#)
5. [Azure Storage Analytics](#)
6. [Azure Active Directory](#)

Microsoft Operations Management Suite

Microsoft Operations Management Suite (OMS) is the IT management solution for the hybrid cloud. Used alone or to extend your existing System Center deployment, OMS gives you the maximum flexibility and control for cloud-based management of your infrastructure.

Insight & Analytics

- Quickly diagnose root cause across the full stack of modern applications and underlying infrastructure
- Monitor and alert on key metrics and KPIs in real time to rapidly identify problems
- Collect, process and analyze petabytes of data
- Create and share data insights across your company in minutes
- Integrate with and extend the value of existing monitoring tools

Protection & Recovery

- Protection of Cloud Assets (DR/Backup for IAAS, Backup of SQL PaaS)
- Enhanced Capacity Planning and Monitoring with Log Analytics
- Enterprise coverage with Linux distros, SQL AG, Encryption at rest
- Faster, Cheaper, Compact Backup Storage ([Xcool](#), De-dup, [ReFS](#))
- Centralized hybrid backup monitoring and reporting in Azure
- Workload protection for public, hybrid, and private cloud
- Enterprise grade VMware VM Backup

Automation & Control

- Trigger immediate action in response to issues automatically or on-demand
- Maintain the state of IT resources and resolve configuration drifts
- Keep IT systems updated with minimal downtime
- Track and manage changes with ease

Security & Compliance

- Collection of security data from virtually any source
- Insight into security status (antimalware, system updates)
- Correlations to detect malicious activities and search for rapid investigation
- Integrates operational and security management
- Threat detection using advanced analytics

With OMS, you can manage any instance in any cloud, including on-premises, Azure, AWS, Windows Server, Linux, VMware, and OpenStack, at a lower cost than competitive solutions. Built for the cloud-first world, OMS offers a new approach to managing your enterprise that is the fastest, most cost-effective way to meet new business challenges and accommodate new workloads, applications and cloud environments.

OMS services

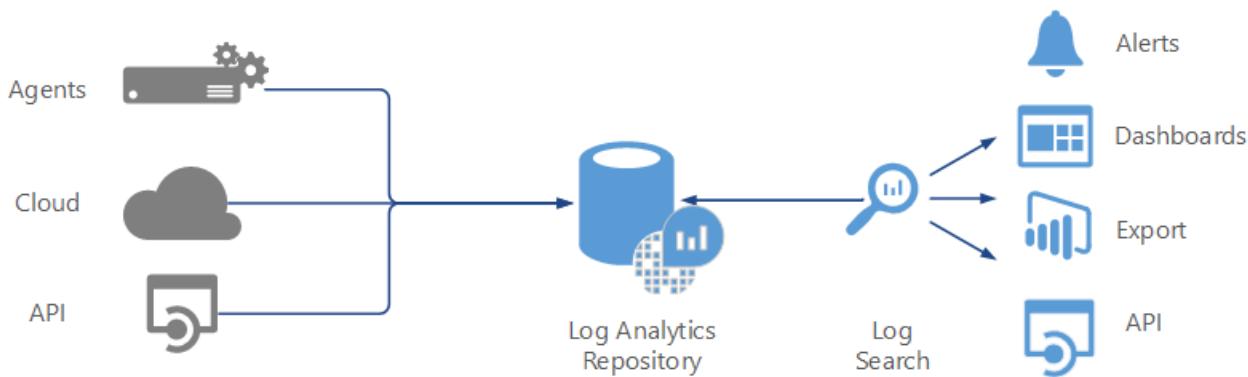
The core functionality of OMS is provided by a set of services that run in Azure. Each service provides a specific management function, and you can combine services to achieve different management scenarios.

SERVICE	DESCRIPTION
Log Analytics	Monitor and analyze the availability and performance of different resources including physical and virtual machines.
Automation	Automate manual processes and enforce configurations for physical and virtual machines.
Backup	Back up and restore critical data.
Site Recovery	Provide high availability for critical applications.

Log Analytics

[Log Analytics](#) provides monitoring services for OMS by collecting data from managed resources into a central repository. This data could include events, performance data, or custom data provided through the API. Once collected, the data is available for alerting, analysis, and export.

This method allows you to consolidate data from various sources, so you can combine data from your Azure services with your existing on-premises environment. It also clearly separates the collection of the data from the action taken on that data so that all actions are available to all kinds of data.



The Log Analytics service manages your cloud-based data securely by using the following methods:

- data segregation
- data retention
- physical security
- incident management
- compliance
- security standards certifications

Azure Backup

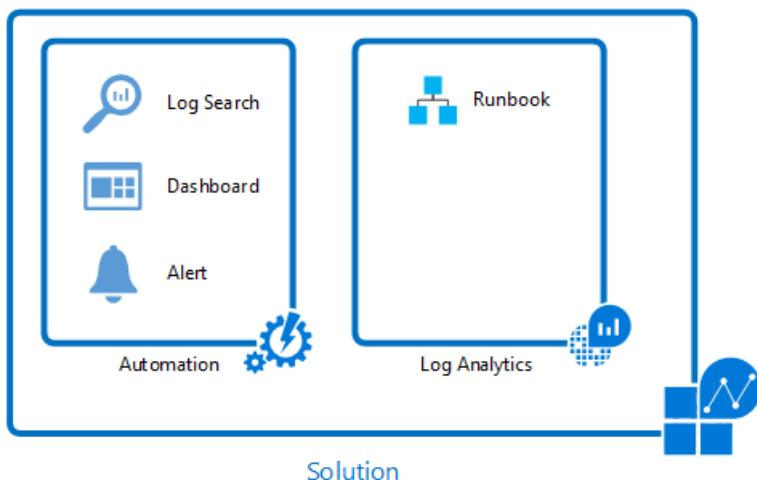
[Azure Backup](#) provides data backup and restore services and is part of the OMS suite of products and services. It protects your application data and retains it for years without any capital investment and with minimal operating costs. It can back up data from physical and virtual Windows servers in addition to application workloads such as SQL Server and SharePoint. It can also be used by [System Center Data Protection Manager \(DPM\)](#) to replicate protected data to Azure for redundancy and long-term storage.

Protected data in Azure Backup is stored in a backup vault located in a particular geographic region. The data is replicated within the same region and, depending on the type of vault, may also be replicated to another region for further resiliency.

Management Solutions

[Microsoft Operations Management Suite \(OMS\)](#) is Microsoft's cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure.

[Management Solutions](#) are prepackaged sets of logics that implement a particular management scenario using one or more OMS services. Different solutions are available from Microsoft and from partners that you can easily add to your Azure subscription to increase the value of your investment in OMS. As a partner, you can create your own solutions to support your applications and services and provide them to users through the Azure Marketplace or Quick Start Templates.



A good example of a solution that uses multiple services to provide additional functionality is the [Update Management solution](#). This solution uses the [Log Analytics agent](#) for Windows and Linux to collect information about required updates on each agent. It writes this data to the Log Analytics repository where you can analyze it with an included dashboard.

When you create a deployment, runbooks in [Azure Automation](#) are used to install required updates. You manage this entire process in the portal and don't need to worry about the underlying details.

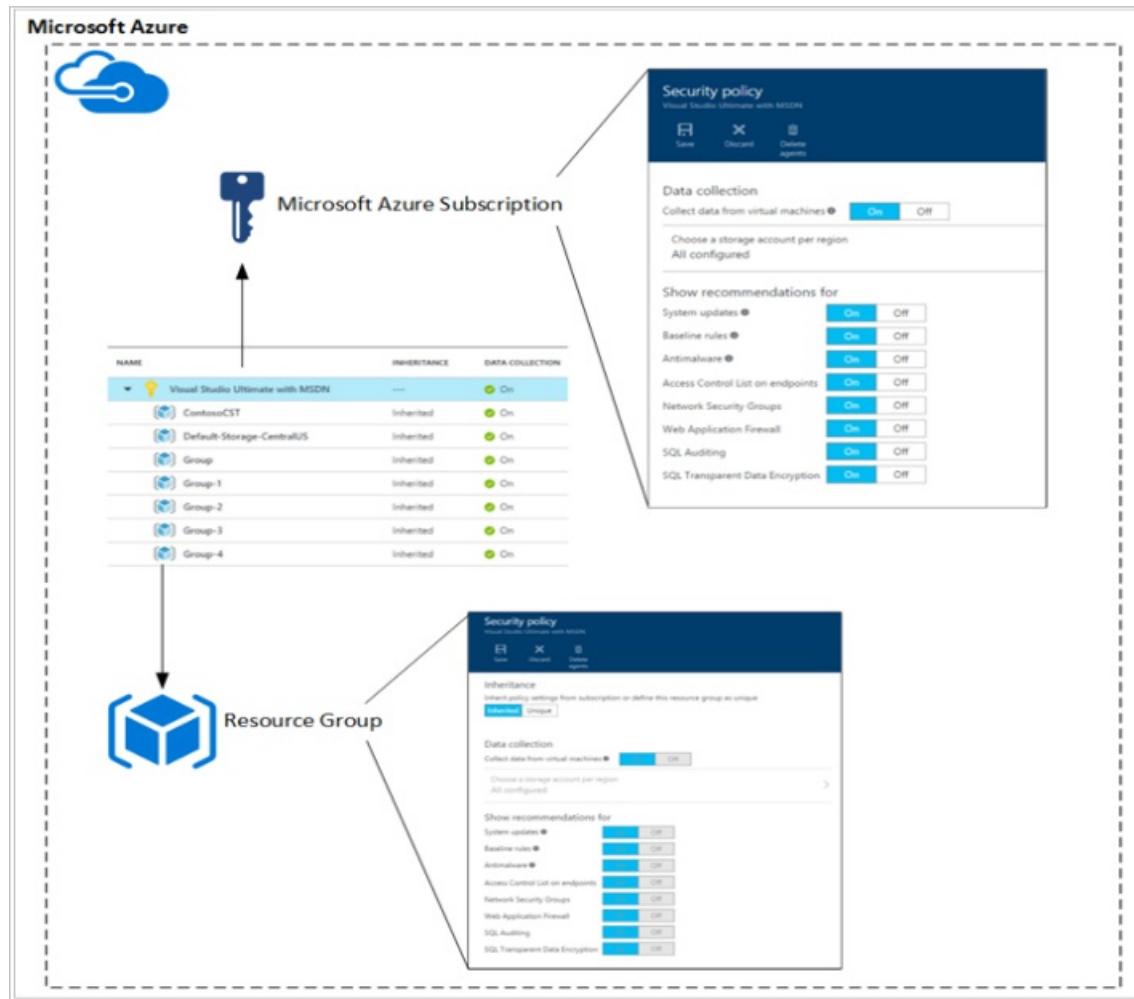
Azure Security Center

Azure Security Center helps protect your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions. Within the service, you are able to define policies not only against your Azure subscriptions, but also against [Resource Groups](#), so you can be more granular.

Security policies and recommendations

A security policy defines the set of controls, which are recommended for resources within the specified subscription or resource group.

In Security Center, you define policies according to your company's security requirements and the type of applications or sensitivity of the data.



Policies that are enabled in the subscription level automatically propagate to all resources groups within the subscription as shown in the diagram at the right side:

Data collection

Security Center collects data from your virtual machines (VMs) to assess their security state, provide security recommendations, and alert you to threats. When you first access Security Center, data collection is enabled on all VMs in your subscription. Data collection is recommended, but you can opt out by turning off data collection in the

Security Center policy.

Data sources

- Azure Security Center analyzes data from the following sources to provide visibility into your security state, identify vulnerabilities and recommend mitigations, and detect active threats:
- Azure Services: Uses information about the configuration of Azure services you have deployed by communicating with that service's resource provider.
- Network Traffic: Uses sampled network traffic metadata from Microsoft's infrastructure, such as source/destination IP/port, packet size, and network protocol.
- Partner Solutions: Uses security alerts from integrated partner solutions, such as firewalls and antimalware solutions.
- Your Virtual Machines: Uses configuration information and information about security events, such as Windows event and audit logs, IIS logs, syslog messages, and crash dump files from your virtual machines.

Data protection

To help customers prevent, detect, and respond to threats, Azure Security Center collects and processes security-related data, including configuration information, metadata, event logs, crash dump files, and more. Microsoft adheres to strict compliance and security guidelines—from coding to operating a service.

- **Data segregation:** Data is kept logically separate on each component throughout the service. All data is tagged per organization. This tagging persists throughout the data lifecycle, and it is enforced at each layer of the service.
- **Data access:** To provide security recommendations and investigate potential security threats, Microsoft personnel may access information collected or analyzed by Azure services, including crash dump files, process creation events, VM disk snapshots and artifacts, which may unintentionally include Customer Data or personal data from your virtual machines. We adhere to the [Microsoft Online Services Terms and Privacy Statement](#), which state that Microsoft is not uses Customer Data or derive information from it for any advertising or similar commercial purposes.
- **Data use:** Microsoft uses patterns and threat intelligence seen across multiple tenants to enhance our prevention and detection capabilities; we do so in accordance with the privacy commitments described in our [Privacy Statement](#).

Data location

Azure Security Center collects ephemeral copies of your crash dump files and analyzes them for evidence of exploit attempts and successful compromises. Azure Security Center performs this analysis within the same Geo as the workspace, and deletes the ephemeral copies when analysis is complete. Machine artifacts are stored centrally in the same region as the VM.

- **Your Storage Accounts:** A storage account is specified for each region where virtual machines are running. This enables you to store data in the same region as the virtual machine from which the data is collected.
- **Azure Security Center Storage:** Information about security alerts, including partner alerts, recommendations, and security health status is stored centrally, currently in the United States. This information may include related configuration information and security events collected from your virtual machines as needed to provide you with the security alert, recommendation, or security health status.

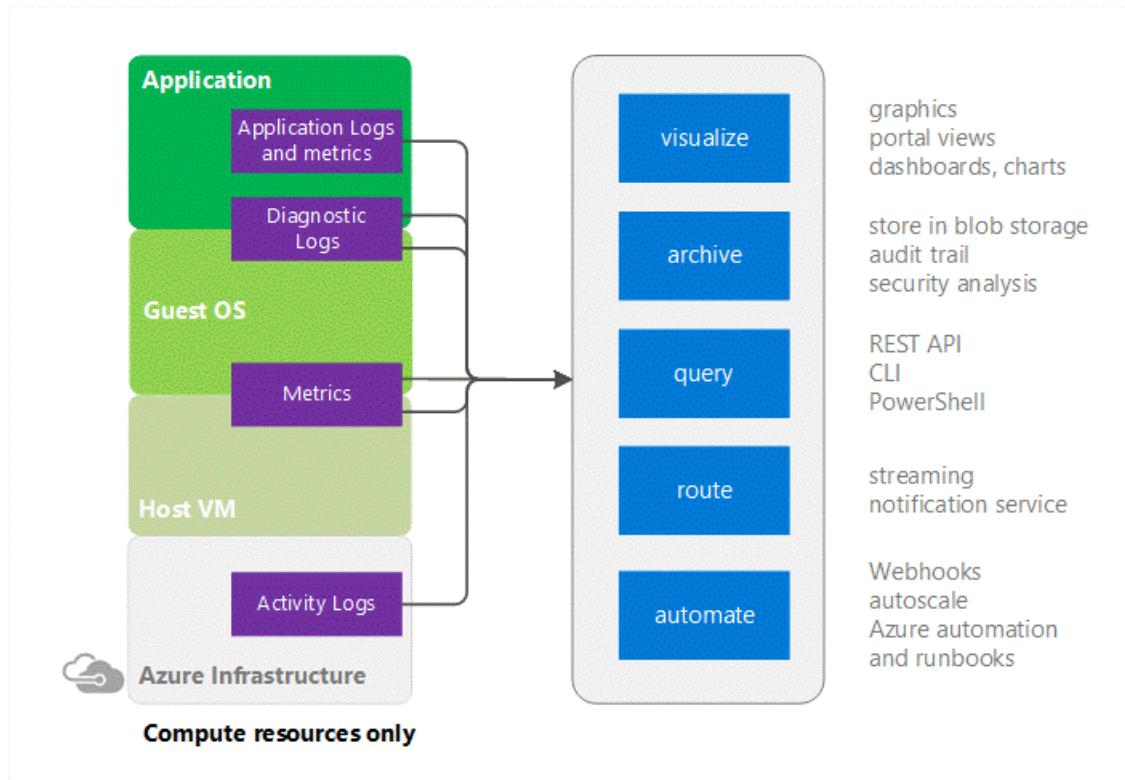
Azure Monitor

The [OMS Security](#) and Audit solution enables IT to actively monitor all resources, which can help minimize the impact of security incidents. OMS Security and Audit have security domains that can be used for monitoring resources. The security domain provides quick access to options, for security monitoring the following domains are

covered in more details:

- Malware assessment
- Update assessment
- Identity and Access.

Azure Monitor provides pointers to information on specific types of resources. It offers visualization, query, routing, alerting, auto scale, and automation on data both from the Azure infrastructure (Activity Log) and each individual Azure resource (Diagnostic Logs).

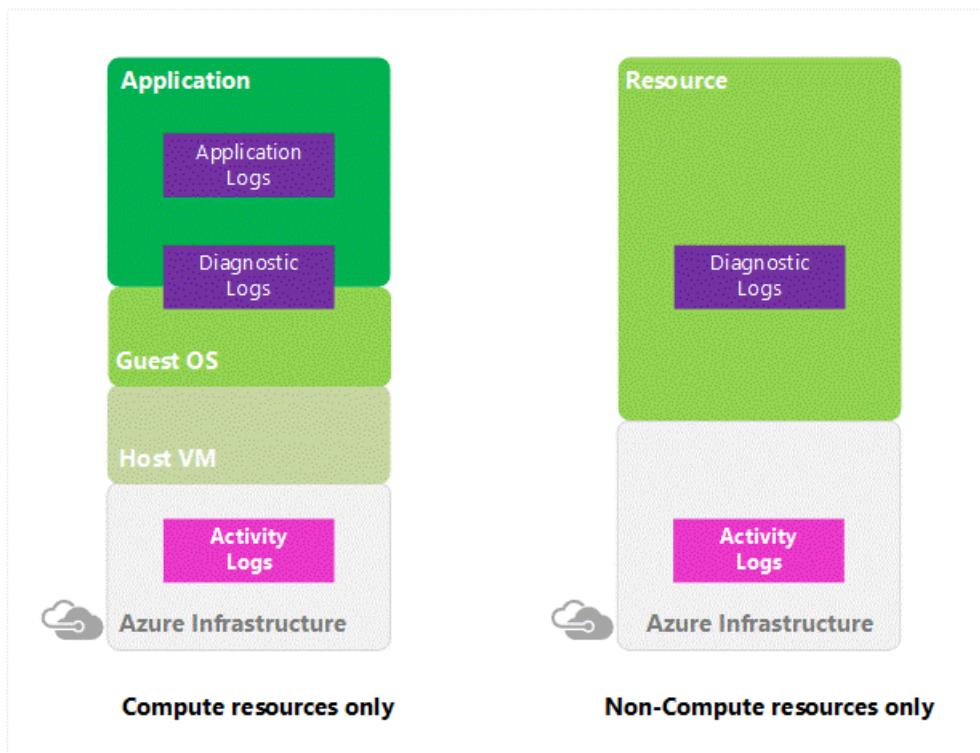


Cloud applications are complex with many moving parts. Monitoring provides data to ensure that your application stays up and running in a healthy state. It also helps you to stave off potential problems or troubleshoot past ones.

In addition, you can use monitoring data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention.

Azure Activity Log

It is a log that provides insight into the operations that were performed on resources in your subscription. The Activity Log was previously known as "Audit Logs" or "Operational Logs," since it reports control-plane events for your subscriptions.



Using the Activity Log, you can determine the 'what, who, and when' for any write operations (PUT, POST, DELETE) taken on the resources in your subscription. You can also understand the status of the operation and other relevant properties. The Activity Log does not include read (GET) operations or operations for resources that use the Classic model.

Azure Diagnostic Logs

These logs are emitted by a resource and provide rich, frequent data about the operation of that resource. The content of these logs varies by resource type.

For example, Windows event system logs are one category of Diagnostic Log for VMs and blob, table, and queue logs are categories of Diagnostic Logs for storage accounts.

Diagnostics Logs differ from the [Activity Log \(formerly known as Audit Log or Operational Log\)](#). The Activity log provides insight into the operations that were performed on resources in your subscription. Diagnostics logs provide insight into operations that your resource performed itself.

Metrics

Azure Monitor enables you to consume telemetry to gain visibility into the performance and health of your workloads on Azure. The most important type of Azure telemetry data is the metrics (also called performance counters) emitted by most Azure resources. Azure Monitor provides several ways to configure and consume these [metrics](#) for monitoring and troubleshooting. Metrics are a valuable source of telemetry and enable you to do the following tasks:

- **Track the performance** of your resource (such as a VM, website, or logic app) by plotting its metrics on a portal chart and pinning that chart to a dashboard.
- **Get notified of an issue** that impacts the performance of your resource when a metric crosses a certain threshold.
- **Configure automated actions**, such as auto scaling a resource or firing a runbook when a metric crosses a certain threshold.
- **Perform advanced analytics** or reporting on performance or usage trends of your resource.
- **Archive** the performance or health history of your resource for compliance or auditing purposes.

Azure Diagnostics

It is the capability within Azure that enables the collection of diagnostic data on a deployed application. You can use the diagnostics extension from various different sources. Currently supported are [Azure Cloud Service Web and Worker Roles](#), [Azure Virtual Machines](#) running Microsoft Windows, and [Service Fabric](#). Other Azure services have their own separate diagnostics.

Azure Network Watcher

Auditing your network security is vital for detecting network vulnerabilities and ensuring compliance with your IT security and regulatory governance model. With Security Group view, you can retrieve the configured Network Security Group and security rules, and the effective security rules. With the list of rules applied, you can determine the ports that are open and assess network vulnerability.

[Network Watcher](#) is a regional service that enables you to monitor and diagnose conditions at a network level in, to, and from Azure. Network diagnostic and visualization tools available with Network Watcher help you understand, diagnose, and gain insights to your network in Azure. This service includes packet capture, next hop, IP flow verify, security group view, NSG flow logs. Scenario level monitoring provides an end to end view of network resources in contrast to individual network resource monitoring.

Topology	Network Diagnostics	Metric	Logs
Visualize your network topology	Diagnostic tools for networking related issues	Measure and view your network performance and health	Configure and view your logs
<ul style="list-style-type: none">Topology	<ul style="list-style-type: none">Variable Packet CaptureIP Flow VerifySecurity Group ViewNext HopVPN Diagnostics	<ul style="list-style-type: none">Network Subscription Limits	<ul style="list-style-type: none">Network Security Flow logsSingle place to configure all logs and Alerts

Network Watcher currently has the following capabilities:

- **Audit Logs** - Operations performed as part of the configuration of networks are logged. These logs can be viewed in the Azure portal or retrieved using Microsoft tools such as Power BI or third-party tools. Audit logs are available through the portal, PowerShell, CLI, and Rest API. For more information on Audit logs, see Audit operations with Resource Manager. Audit logs are available for operations done on all network resources.
- **IP flow verifies** - Checks if a packet is allowed or denied based on flow information 5-tuple packet parameters (Destination IP, Source IP, Destination Port, Source Port, and Protocol). If the packet is denied by a Network Security Group, the rule and Network Security Group that denied the packet is returned.
- **Next hop** - Determines the next hop for packets being routed in the Azure Network Fabric, enabling you to diagnose any misconfigured user-defined routes.
- **Security group view** - Gets the effective and applied security rules that are applied on a VM.
- **NSG Flow logging** - Flow logs for Network Security Groups enable you to capture logs related to traffic that are allowed or denied by the security rules in the group. The flow is defined by a 5-tuple information – Source IP, Destination IP, Source Port, Destination Port, and Protocol.

Azure Storage Analytics

Storage Analytics can store metrics that include aggregated transaction statistics and capacity data about Requests to a storage service. Transactions are reported at both the API operation level and at the storage service level, and capacity is reported at the storage service level. Metrics data can be used to analyze storage service usage, diagnose issues with requests made against the storage service, and to improve the performance of applications that use a service.

[Azure Storage Analytics](#) performs logging and provides metrics data for a storage account. You can use this data to trace requests, analyze usage trends, and diagnose issues with your storage account. Storage Analytics logging is available for the [Blob, Queue, and Table services](#). Storage Analytics logs detailed information about successful and failed requests to a storage service.

This information can be used to monitor individual requests and to diagnose issues with a storage service. Requests are logged on a best-effort basis. Log entries are created only if there are requests made against the service endpoint. For example if a storage account has activity in its Blob endpoint but not in its Table or Queue endpoints, only logs pertaining to the Blob service is created.

To use Storage Analytics, you must enable it individually for each service you want to monitor. You can enable it in the [Azure portal](#); for details, see [Monitor a storage account in the Azure portal](#). You can also enable Storage Analytics programmatically via the REST API or the client library. Use the Set Service Properties operation to enable Storage Analytics individually for each service.

The aggregated data is stored in a well-known blob (for logging) and in well-known tables (for metrics), which may be accessed using the Blob service and Table service APIs.

Storage Analytics has a 20-TB limit on the amount of stored data that is independent of the total limit for your storage account. All logs are stored in [block blobs](#) in a container named \$logs, which are automatically created when Storage Analytics is enabled for a storage account.

The following actions performed by Storage Analytics are billable:

- Requests to create blobs for logging
- Requests to create table entities for metrics.

NOTE

For more information on billing and data retention policies, see [Storage Analytics and Billing](#). For optimal performance, you want to limit the number of highly utilized disks attached to the virtual machine to avoid possible throttling. If all disks are not being highly utilized at the same time, the storage account can support a larger number disk.

NOTE

For more information on storage account limits, see [Azure Storage Scalability and Performance Targets](#).

The following types of authenticated and anonymous requests are logged.

AUTHENTICATED	ANONYMOUS
Successful requests	Successful requests
Failed requests, including timeout, throttling, network, authorization, and other errors	Requests using a Shared Access Signature (SAS), including failed and successful requests
Requests using a Shared Access Signature (SAS), including failed and successful requests	Time out errors for both client and server

AUTHENTICATED	ANONYMOUS
Requests to analytics data	Failed GET requests with error code 304 (Not Modified)
Requests made by Storage Analytics itself, such as log creation or deletion, are not logged. A full list of the logged data is documented in the Storage Analytics Logged Operations and Status Messages and Storage Analytics Log Format topics.	All other failed anonymous requests are not logged. A full list of the logged data is documented in the Storage Analytics Logged Operations and Status Messages and Storage Analytics Log Format .

Azure Active Directory

Azure AD also includes a full suite of identity management capabilities including multi-factor authentication, device registration, self-service password management, self-service group management, privileged account management, role-based access control, application usage monitoring, rich auditing, and security monitoring and alerting.

- Improve application security with Azure AD multifactor authentication and conditional access.
- Monitor application usage and protect your business from advanced threats with security reporting and monitoring.

Azure Active Directory (Azure AD) includes security, activity, and audit reports for your directory. [The Azure Active Directory Audit Report](#) helps customers to identify privileged actions that occurred in their Azure Active Directory. Privileged actions include elevation changes (for example, role creation or password resets), changing policy configurations (for example password policies), or changes to directory configuration (for example, changes to domain federation settings).

The reports provide the audit record for the event name, the actor who performed the action, the target resource affected by the change, and the date and time (in UTC). Customers are able to retrieve the list of audit events for their Azure Active Directory via the [Azure portal](#), as described in [View your Audit Logs](#). Here's a list of the reports included:

SECURITY REPORTS	ACTIVITY REPORTS	AUDIT REPORTS
Sign-ins from unknown sources	Application usage: summary	Directory audit report
Sign-ins after multiple failures	Application usage: detailed	
Sign-ins from multiple geographies	Application dashboard	
Sign-ins from IP addresses with suspicious activity	Account provisioning errors	
Irregular sign-in activity	Individual user devices	
Sign-ins from possibly infected devices	Individual user Activity	
Users with anomalous sign-in activity	Groups activity report	
	Password Reset Registration Activity Report	
	Password reset activity	

The data of these reports can be useful to your applications, such as SIEM systems, audit, and business intelligence tools. The Azure AD reporting [APIs](#) provide programmatic access to the data through a set of REST-based APIs. You

can call these APIs from various programming languages and tools.

Events in the Azure AD Audit report are retained for 180 days.

NOTE

For more information about retention on reports, see [Azure Active Directory Report Retention Policies](#).

For customers interested in storing their [audit events](#) for longer retention periods, the Reporting API can be used to regularly pull audit events into a separate data store.

Summary

This article summarizes protecting your privacy and securing your data, while delivering software and services that help you manage the IT infrastructure of your organization. Microsoft recognizes that when they entrust their data to others, that trust requires rigorous security. Microsoft adheres to strict compliance and security guidelines—from coding to operating a service. Securing and protecting data is a top priority at Microsoft.

This article explains

- How data is collected, processed, and secured in the Operations Management Suite (OMS).
- Quickly analyze events across multiple data sources. Identify security risks and understand the scope and impact of threats and attacks to mitigate the damage of a security breach.
- Identify attack patterns by visualizing outbound malicious IP traffic and malicious threat types. Understand the security posture of your entire environment regardless of platform.
- Capture all the log and event data required for a security or compliance audit. Slash the time and resources needed to supply a security audit with a complete, searchable, and exportable log and event data set.
- Collect security-related events, audit, and breach analysis to keep a close eye on your assets:
- Security posture
- Notable issue
- Summaries threats

Next Steps

- [Design and operational security](#)

Microsoft designs its services and software with security in mind to help ensure that its cloud infrastructure is resilient and defended from attacks.

- [Operations Management Suite | Security & Compliance](#)

Use Microsoft security data and analysis to perform more intelligent and effective threat detection.

- [Azure Security Center planning and operations](#) A set of steps and tasks that you can follow to optimize your use of Security Center based on your organization's security requirements and cloud management model.

Azure Advanced Threat Detection

8/21/2017 • 23 min to read • [Edit Online](#)

Introduction

Overview

Microsoft has developed a series of White Papers, Security Overviews, Best Practices, and Checklists to assist Azure customers about the various security-related capabilities available in and surrounding the Azure Platform. The topics range in terms of breadth and depth and are updated periodically. This document is part of that series as summarized in the following abstract section.

Azure Platform

Azure is a public cloud service platform that supports the broadest selection of operating systems, programming languages, frameworks, tools, databases, and devices. It supports the following programming languages:

- Run Linux containers with Docker integration.
- Build apps with JavaScript, Python, .NET, PHP, Java, and Node.js
- Build back-ends for iOS, Android, and Windows devices.

Azure public cloud services support the same technologies millions of developers and IT professionals already rely on and trust.

When you are migrating to a public cloud with an organization, that organization is responsible to protect your data and provide security and governance around the system.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security needs. Azure provides a wide array of options to configure and customize security to meet the requirements of your app deployments. This document helps you meet these requirements.

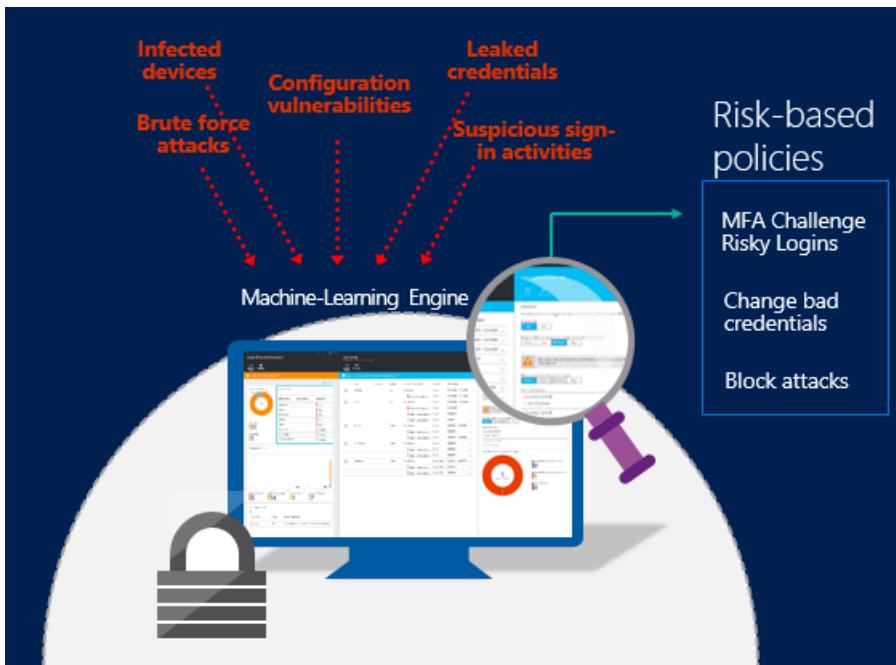
Abstract

Microsoft Azure offers built in advanced threat detection functionality through services like Azure Active Directory, Azure Operations Management Suite (OMS), and Azure Security Center. This collection of security services and capabilities provides a simple and fast way to understand what is happening within your Azure deployments.

This white paper will guide you the "Microsoft Azure approaches" towards threat vulnerability diagnostic and analysing the risk associated with the malicious activities targeted against servers and other Azure resources. This helps you to identify the methods of identification and vulnerability management with optimized solutions by the Azure platform and customer-facing security services and technologies.

This white paper focuses on the technology of Azure platform and customer-facing controls, and does not attempt to address SLAs, pricing models, and DevOps practice considerations.

Azure Active Directory Identity Protection



Azure Active Directory Identity Protection is a feature of the [Azure AD Premium P2](#) edition that provides you an overview of the risk events and potential vulnerabilities affecting your organization's identities. Microsoft has been securing cloud-based identities for over a decade, and with Azure AD Identity Protection, Microsoft is making these same protection systems available to enterprise customers. Identity Protection uses existing Azure AD's anomaly detection capabilities available through [Azure AD's Anomalous Activity Reports](#), and introduces new risk event types that can detect real time anomalies.

Identity Protection uses adaptive machine learning algorithms and heuristics to detect anomalies and risk events that may indicate that an identity has been compromised. Using this data, Identity Protection generates reports and alerts that enable you to investigate these risk events and take appropriate remediation or mitigation action.

But Azure Active Directory Identity Protection is more than a monitoring and reporting tool. Based on risk events, Identity Protection calculates a user risk level for each user, enabling you to configure risk-based policies to automatically protect the identities of your organization.

These risk-based policies, in addition to other [conditional access controls](#) provided by Azure Active Directory and [EMS](#), can automatically block or offer adaptive remediation actions that include password resets and multi-factor authentication enforcement.

Identity Protection's capabilities

Azure Active Directory Identity Protection is more than a monitoring and reporting tool. To protect your organization's identities, you can configure risk-based policies that automatically respond to detected issues when a specified risk level has been reached. These policies, in addition to other conditional access controls provided by Azure Active Directory and EMS, can either automatically block or initiate adaptive remediation actions including password resets and multi-factor authentication enforcement.

Examples of some of the ways that Azure Identity Protection can help secure your accounts and identities include:

Detecting risk events and risky accounts:

- Detecting six risk event types using machine learning and heuristic rules
- Calculating user risk levels
- Providing custom recommendations to improve overall security posture by highlighting vulnerabilities

Investigating risk events:

- Sending notifications for risk events
- Investigating risk events using relevant and contextual information

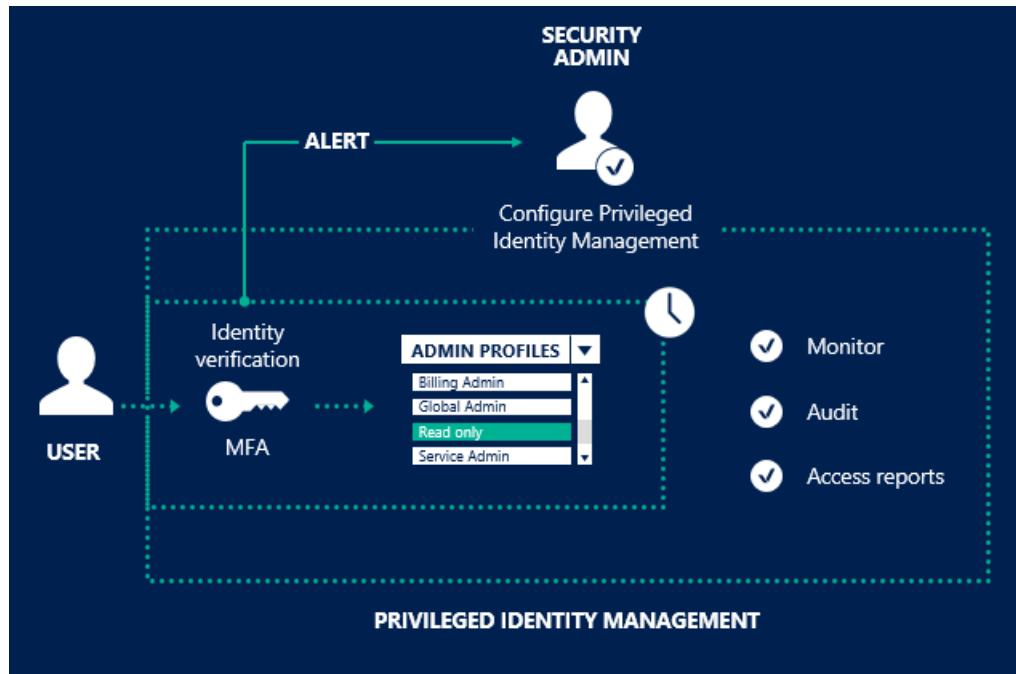
- Providing basic workflows to track investigations
- Providing easy access to remediation actions such as password reset

Risk-based conditional access policies:

- Policy to mitigate risky sign-ins by blocking sign-ins or requiring multi-factor authentication challenges.
- Policy to block or secure risky user accounts
- Policy to require users to register for multi-factor authentication

Azure AD Privileged Identity Management (PIM)

With [Azure Active Directory \(AD\) Privileged Identity Management](#),



you can manage, control, and monitor access within your organization. This includes access to resources in Azure AD and other Microsoft online services like Office 365 or Microsoft Intune.

Azure AD Privileged Identity Management helps you:

- Get an alert and report on Azure AD administrators and "just in time" administrative access to Microsoft Online Services like Office 365 and Intune
- Get reports about administrator access history and changes in administrator assignments
- Get alerts about access to a privileged role

Microsoft Operations Management Suite (OMS)

[Microsoft Operations Management Suite](#) is Microsoft's cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure. Since OMS is implemented as a cloud-based service, you can have it up and running quickly with minimal investment in infrastructure services. New security features are delivered automatically, saving your ongoing maintenance and upgrade costs.

In addition to providing valuable services on its own, OMS can integrate with System Center components such as [System Center Operations Manager](#) to extend your existing security management investments into the cloud. System Center and OMS can work together to provide a full hybrid management experience.

Holistic Security and Compliance Posture

The [OMS Security and Audit dashboard](#) provides a comprehensive view into your organization's IT security posture with built-in search queries for notable issues that require your attention. The Security and Audit dashboard is the

home screen for everything related to security in OMS. It provides high-level insight into the security state of your computers. It also includes the ability to view all events from the past 24 hours, 7 days, or any other custom time frame.

OMS dashboards help you quickly and easily understand the overall security posture of any environment, all within the context of IT Operations, including: software update assessment, antimalware assessment, and configuration baselines. Furthermore, security log data is readily accessible to streamline the security and compliance audit processes.

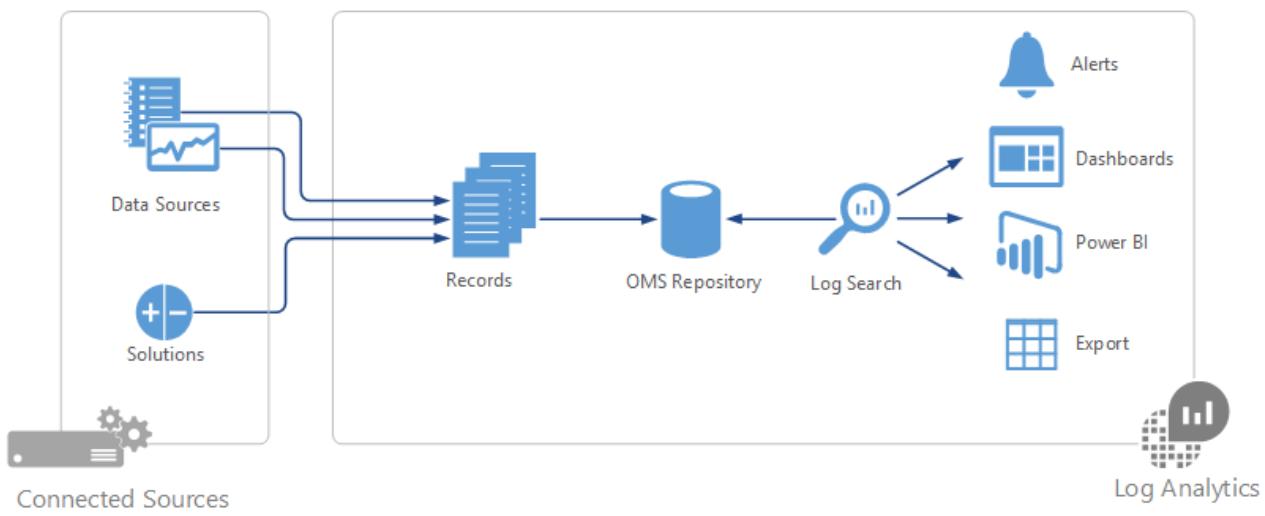
The OMS Security and Audit dashboard is organized in four major categories:



- **Security Domains:** in this area, you will be able to further explore security records over time, access malware assessment, update assessment, network security, identity and access information, computers with security events and quickly have access to Azure Security Center dashboard.
- **Notable Issues:** this option allows you to quickly identify the number of active issues and the severity of these issues.
- **Detections (Preview):** enables you to identify attack patterns by visualizing security alerts as they take place against your resources.
- **Threat Intelligence:** enables you to identify attack patterns by visualizing the total number of servers with outbound malicious IP traffic, the malicious threat type, and a map that shows where these IPs are coming from.
- **Common security queries:** this option provides you a list of the most common security queries that you can use to monitor your environment. When you click in one of those queries, it opens the Search blade with the results for that query.

Insight and Analytics

At the center of [Log Analytics](#) is the OMS repository, which is hosted in the Azure cloud.



Data is collected into the repository from connected sources by configuring data sources and adding solutions to your subscription.

The screenshot shows the Microsoft Operations Management Suite (OMS) portal dashboard. The left sidebar includes links for Log Search, My Dashboard, Solutions Gallery, Usage, and Settings. The main area displays various performance metrics and charts:

- Alert Management**: 0 active critical alerts, 0 active warning alerts.
- Malware Assessment**: 7 computers need attention.
- Automation**: Runbooks, Jobs in the last 7 days.
- Change Tracking**: 32 software changes, 19 Windows service and Linux daemon changes.
- Security and Audit**: 13 active computers, 776 accounts authenticated.
- SQL Assessment**: 2 servers assessed, 7 low priority recommendations, 83 passed checks.
- System Update Assessment**: 23.1% of computers need attention.
- Latest News**: MS Ops Mgmt Suite (@msopsmgmt) tweet about security events.
- Settings**: 100% of items completed, 32 data sources connected.

Data sources and solutions will each create different record types that have their own set of properties but may still be analyzed together in queries to the repository. This allows you to use the same tools and methods to work with different kinds of data collected by different sources.

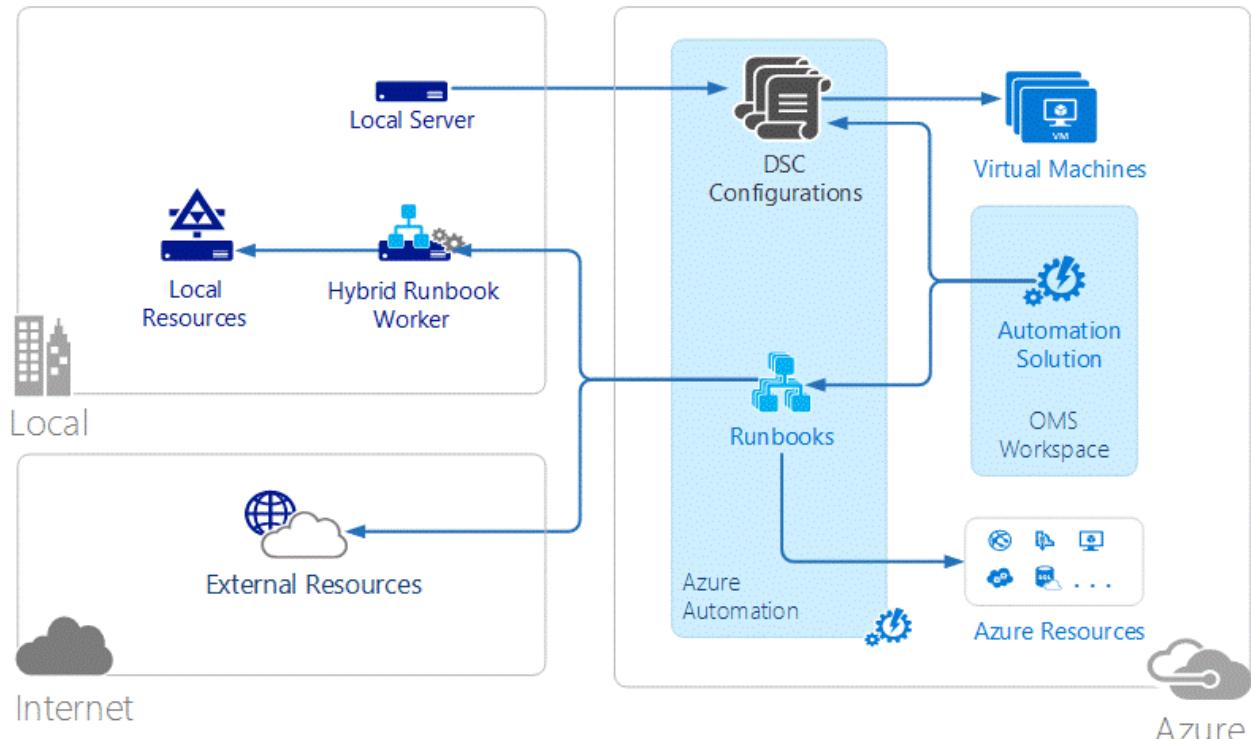
Most of your interaction with Log Analytics is through the OMS portal, which runs in any browser and provides you with access to configuration settings and multiple tools to analyze and act on collected data. From the portal, you can use [log searches](#) where you construct queries to analyze collected data, [dashboards](#), which you can customize with graphical views of your most valuable searches, and [solutions](#), which provide additional functionality and analysis tools.

Solutions Gallery							
							
App Dependency Monitor Coming Soon Automatically discover and map servers and their dependencies in real-time.	Malware Assessment Owned View status of antivirus and antimalware scans across your servers.	Containers Coming Soon See Docker container performance metrics and logs from containers across your public or private cloud environments.	Network Performance Monitor Coming Soon Offers near real time monitoring of network performance parameters like loss and latency.	Security and Audit Owned Provides the ability to explore security related data and helps identify security breaches.	System Update Assessment Owned Identify missing system updates across your servers.	AD Replication Status Owned Identify Active Directory replication issues in your environment.	Malware Assessment Owned View status of antivirus and antimalware scans across your servers.
							
Azure Networking Analytics Coming Soon Gain insight into your Azure Network data	Security and Audit Owned Provides the ability to explore security related data and helps identify security breaches.	Wire Data Coming Soon Provides the ability to explore wire data and helps identify network related issues.	Office 365 Coming Soon Get full visibility into your Office 365 user activities, perform forensics as well as audit and compliance.	SQL Assessment Free Assess the risk and health of SQL Server environments.	AD Assessment Owned Assess the risk and health of Active Directory environments.	Alert Management Owned View your Operations Manager and OMS alerts to easily triage alerts and identify the root causes of problems in your environment.	Automation Owned Automate time consuming and frequently repeated tasks in the cloud and on-premises.

Solutions add functionality to Log Analytics. They primarily run in the cloud and provide analysis of data collected in the OMS repository. They may also define new record types to be collected that can be analyzed with Log Searches or by additional user interface provided by the solution in the OMS dashboard. The Security and Audit is an example of these types of solutions.

Automation & Control: Alert on security configuration drifts

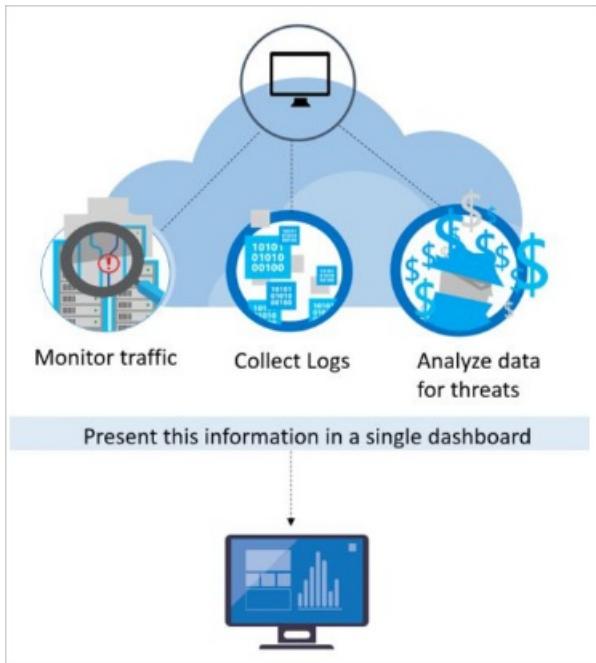
Azure Automation automates administrative processes with runbooks that are based on PowerShell and run in the Azure cloud. Runbooks can also be executed on a server in your local data center to manage local resources. Azure Automation provides configuration management with PowerShell DSC (Desired State Configuration).



You can create and manage DSC resources hosted in Azure and apply them to cloud and on-premises systems to define and automatically enforce their configuration or get reports on drift to help insure that security configurations remain within policy.

Azure Security Center

Azure Security Center helps protect your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions. Within the service, you are able to define policies not only against your Azure subscriptions, but also against [Resource Groups](#), so you can be more granular.



Microsoft security researchers are constantly on the lookout for threats. They have access to an expansive set of telemetry gained from Microsoft's global presence in the cloud and on-premises. This wide-reaching and diverse collection of datasets enables Microsoft to discover new attack patterns and trends across its on-premises consumer and enterprise products, as well as its online services.

Thus, Security Center can rapidly update its detection algorithms as attackers release new and increasingly sophisticated exploits. This approach helps you keep pace with a fast-moving threat environment.

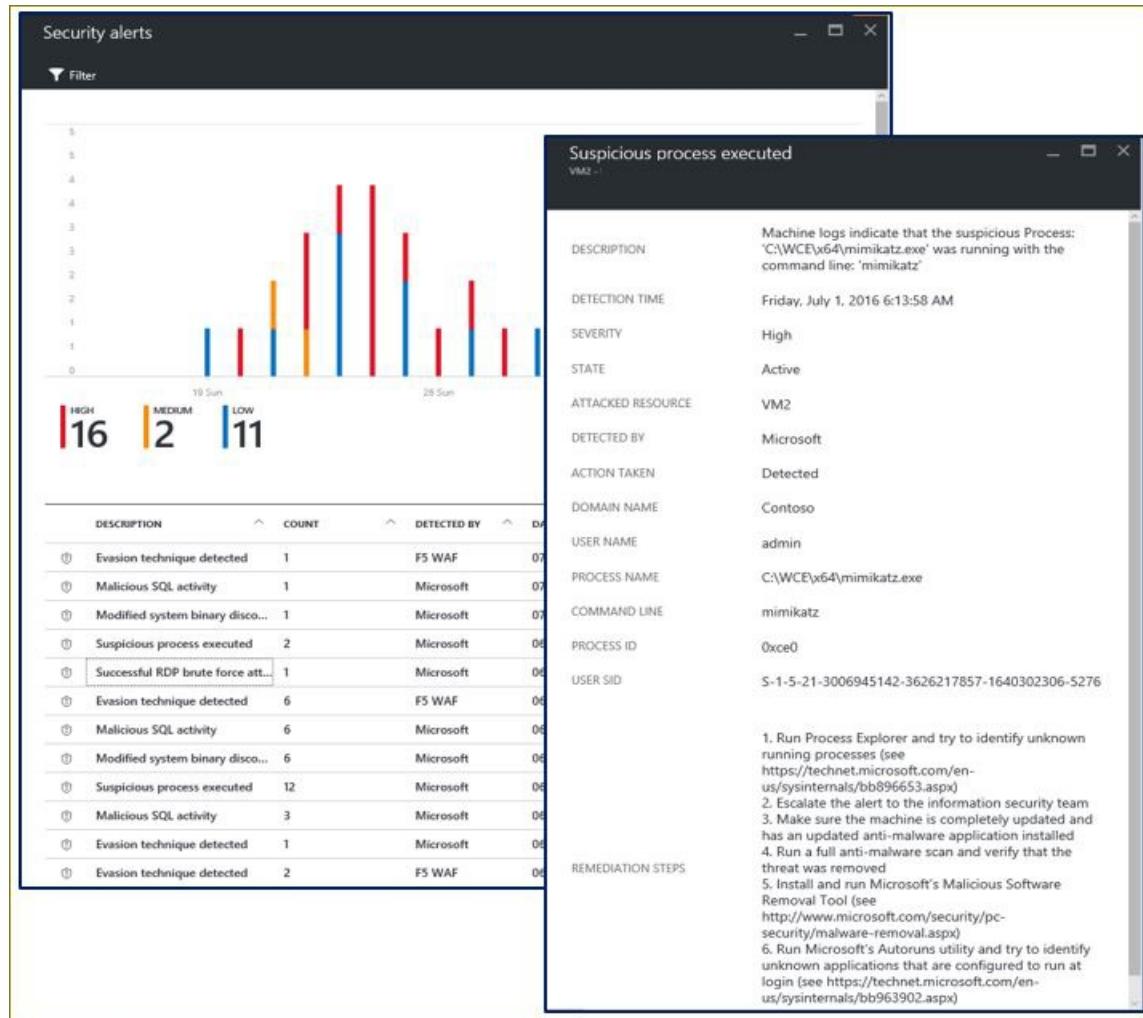


Security Center threat detection works by automatically collecting security information from your Azure resources, the network, and connected partner solutions. It analyzes this information, correlating information from multiple sources, to identify threats. Security alerts are prioritized in Security Center along with recommendations on how to remediate the threat.

Security Center employs advanced security analytics, which go far beyond signature-based approaches. Breakthroughs in big data and [machine learning](#) technologies are used to evaluate events across the entire cloud fabric – detecting threats that would be impossible to identify using manual approaches and predicting the evolution of attacks. These security analytics includes the following.

Threat Intelligence

Microsoft has an immense amount of global threat intelligence. Telemetry flows in from multiple sources, such as Azure, Office 365, Microsoft CRM online, Microsoft Dynamics AX, outlook.com, MSN.com, the Microsoft Digital Crimes Unit (DCU), and Microsoft Security Response Center (MSRC).



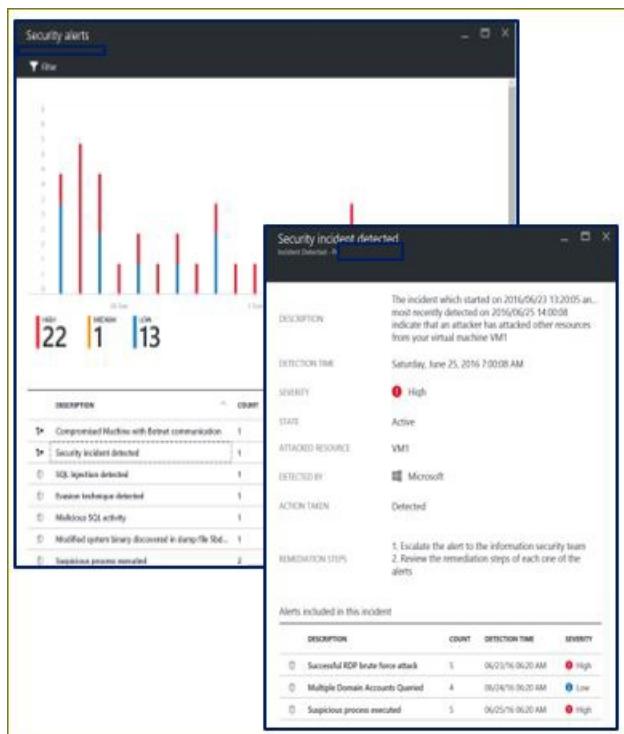
Researchers also receive threat intelligence information that is shared among major cloud service providers and subscribes to threat intelligence feeds from third parties. Azure Security Center can use this information to alert you to threats from known bad actors. Some examples include:

- **Harnessing the Power of Machine Learning** - Azure Security Center has access to a vast amount of data about cloud network activity, which can be used to detect threats targeting your Azure deployments. For example:
- **Brute Force Detections** - Machine learning is used to create a historical pattern of remote access attempts, which allows it to detect brute force attacks against SSH, RDP, and SQL ports.
- **Outbound DDoS and Botnet Detection** - A common objective of attacks targeting cloud resources is to use the compute power of these resources to execute other attacks.

- **New Behavioral Analytics Servers and VMs** - Once a server or virtual machine is compromised, attackers employ a wide variety of techniques to execute malicious code on that system while avoiding detection, ensuring persistence, and obviating security controls.
- **Azure SQL Database Threat Detection** - Threat Detection for Azure SQL Database, which identifies anomalous database activities indicating unusual and potentially harmful attempts to access or exploit databases.

Behavioral analytics

Behavioral analytics is a technique that analyzes and compares data to a collection of known patterns. However, these patterns are not simple signatures. They are determined through complex machine learning algorithms that are applied to massive datasets.



They are also determined through careful analysis of malicious behaviors by expert analysts. Azure Security Center can use behavioral analytics to identify compromised resources based on analysis of virtual machine logs, virtual network device logs, fabric logs, crash dumps, and other sources.

In addition, there is correlation with other signals to check for supporting evidence of a widespread campaign. This correlation helps to identify events that are consistent with established indicators of compromise.

Some examples include:

- **Suspicious process execution:** Attackers employ several techniques to execute malicious software without detection. For example, an attacker might give malware the same names as legitimate system files but place these files in an alternate location, use a name that is very like a benign file, or mask the file's true extension. Security Center models processes behaviors and monitors process executions to detect outliers such as these.
- **Hidden malware and exploitation attempts:** Sophisticated malware can evade traditional antimalware products by either never writing to disk or encrypting software components stored on disk. However, such malware can be detected using memory analysis, as the malware must leave traces in memory to function. When software crashes, a crash dump captures a portion of memory at the time of the crash. By analyzing the memory in the crash dump, Azure Security Center can detect techniques used to exploit vulnerabilities in software, access confidential data, and surreptitiously persist within a compromised machine without impacting the performance of your machine.

- **Lateral movement and internal reconnaissance:** To persist in a compromised network and locate/harvest valuable data, attackers often attempt to move laterally from the compromised machine to others within the same network. Security Center monitors process and login activities to discover attempts to expand an attacker's foothold within the network, such as remote command execution, network probing, and account enumeration.
- **Malicious PowerShell Scripts:** PowerShell can be used by attackers to execute malicious code on target virtual machines for a various purposes. Security Center inspects PowerShell activity for evidence of suspicious activity.
- **Outgoing attacks:** Attackers often target cloud resources with the goal of using those resources to mount additional attacks. Compromised virtual machines, for example, might be used to launch brute force attacks against other virtual machines, send SPAM, or scan open ports and other devices on the Internet. By applying machine learning to network traffic, Security Center can detect when outbound network communications exceed the norm. When SPAM, Security Center also correlates unusual email traffic with intelligence from Office 365 to determine whether the mail is likely nefarious or the result of a legitimate email campaign.

Anomaly Detection

Azure Security Center also uses anomaly detection to identify threats. In contrast to behavioral analytics (which depends on known patterns derived from large data sets), anomaly detection is more "personalized" and focuses on baselines that are specific to your deployments. Machine learning is applied to determine normal activity for your deployments and then rules are generated to define outlier conditions that could represent a security event. Here's an example:

- **Inbound RDP/SSH brute force attacks:** Your deployments may have busy virtual machines with many logins each day and other virtual machines that have few or any logins. Azure Security Center can determine baseline login activity for these virtual machines and use machine learning to define around the normal login activities. If there is any discrepancy with the baseline defined for login related characteristics, then an alert may be generated. Again, machine learning determines what is significant.

Continuous Threat Intelligence Monitoring

Azure Security Center operates with security research and data science teams throughout the world that continuously monitor for changes in the threat landscape. This includes the following initiatives:

- **Threat intelligence monitoring:** Threat intelligence includes mechanisms, indicators, implications, and actionable advice about existing or emerging threats. This information is shared in the security community and Microsoft continuously monitors threat intelligence feeds from internal and external sources.
- **Signal sharing:** Insights from security teams across Microsoft's broad portfolio of cloud and on-premises services, servers, and client endpoint devices are shared and analyzed.
- **Microsoft security specialists:** Ongoing engagement with teams across Microsoft that work in specialized security fields, like forensics and web attack detection.
- **Detection tuning:** Algorithms are run against real customer data sets and security researchers work with customers to validate the results. True and false positives are used to refine machine learning algorithms.

These combined efforts culminate in new and improved detections, which you can benefit from instantly – there's no action for you to take.

Advanced Threat Detection Features - Other Azure Services

Virtual Machine: Microsoft Antimalware

[Microsoft Antimalware](#) for Azure is a single-agent solution for applications and tenant environments, designed to run in the background without human intervention. You can deploy protection based on the needs of your

application workloads, with either basic secure-by-default or advanced custom configuration, including antimalware monitoring. Azure antimalware is a security option for Azure Virtual Machines and is automatically installed on all Azure PaaS virtual machines.

Features of Azure to deploy and enable Microsoft Antimalware for your applications

Microsoft Antimalware Core Features

- **Real-time protection** - monitors activity in Cloud Services and on Virtual Machines to detect and block malware execution.
- **Scheduled scanning** - periodically performs targeted scanning to detect malware, including actively running programs.
- **Malware remediation** - automatically takes action on detected malware, such as deleting or quarantining malicious files and cleaning up malicious registry entries.
- **Signature updates** - automatically installs the latest protection signatures (virus definitions) to ensure protection is up-to-date on a pre-determined frequency.
- **Antimalware Engine updates** - automatically updates the Microsoft Antimalware engine.
- **Antimalware Platform updates** – automatically updates the Microsoft Antimalware platform.
- **Active protection** - reports telemetry metadata about detected threats and suspicious resources to Microsoft Azure to ensure rapid response to the evolving threat landscape, and enabling real-time synchronous signature delivery through the Microsoft Active Protection System (MAPS).
- **Samples reporting** - provides and reports samples to the Microsoft Antimalware service to help refine the service and enable troubleshooting.
- **Exclusions** – allows application and service administrators to configure certain files, processes, and drives to exclude them from protection and scanning for performance and/or other reasons.
- **Antimalware event collection** - records the antimalware service health, suspicious activities, and remediation actions taken in the operating system event log and collects them into the customer's Azure Storage account.

Azure SQL Database Threat Detection

[Azure SQL Database Threat Detection](#) is a new security intelligence feature built into the Azure SQL Database service. Working around the clock to learn, profile and detect anomalous database activities, Azure SQL Database Threat Detection identifies potential threats to the database.

Security officers or other designated administrators can get an immediate notification about suspicious database activities as they occur. Each notification provides details of the suspicious activity and recommends how to further investigate and mitigate the threat.

Currently, Azure SQL Database Threat Detection detects potential vulnerabilities and SQL injection attacks, and anomalous database access patterns.

Upon receiving threat detection email notification, users are able to navigate and view the relevant audit records using the deep link in the mail that opens an audit viewer and/or preconfigured auditing Excel template that shows the relevant audit records around the time of the suspicious event according to the following:

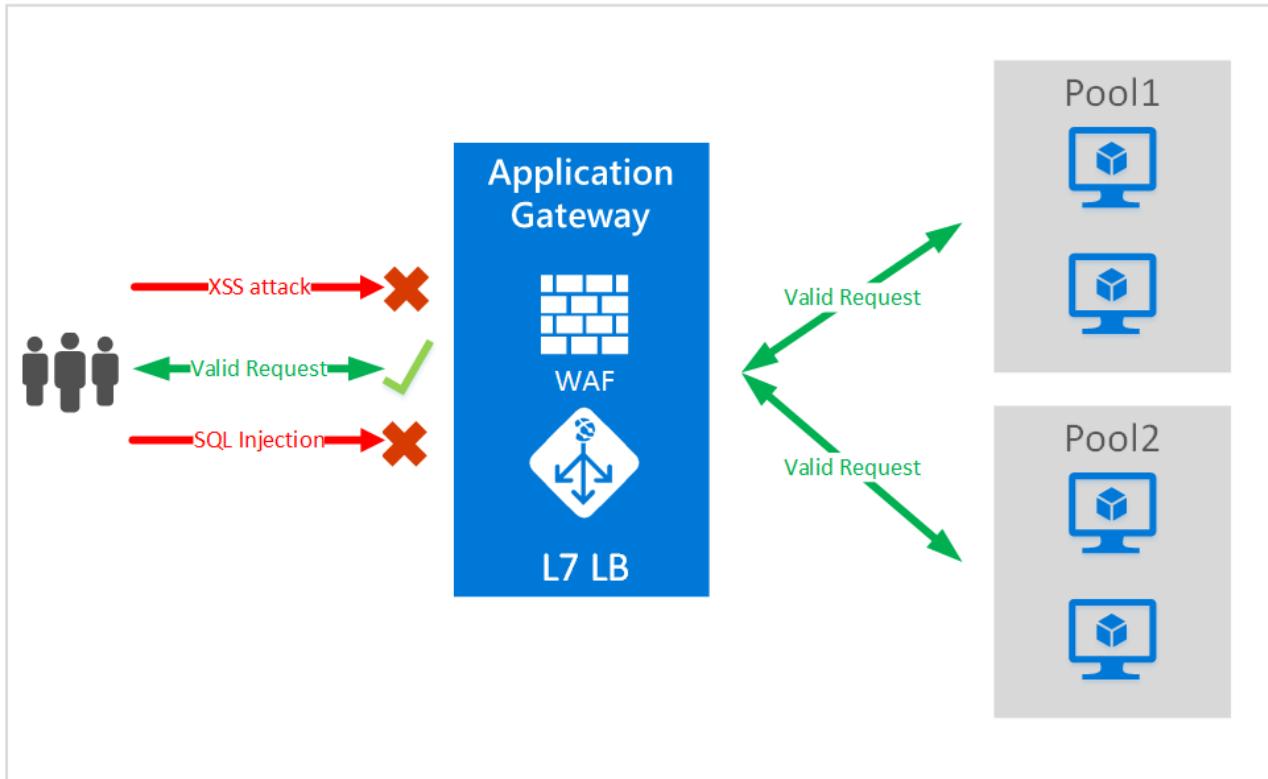
- Audit storage for the database/server with the anomalous database activities
- Relevant audit storage table that was used at the time of the event to write audit log
- Audit records of the following hour since the event occurs.
- Audit records with similar event ID at the time of the event (optional for some detectors)

SQL Database Threat Detectors use one of the following detection methodologies:

- **Deterministic Detection** – detects suspicious patterns (rules based) in the SQL client queries that match known attacks. This methodology has high detection and low false positive, however limited coverage because it falls within the category of "atomic detections".
- **Behavioural Detection** – detects anomalous activity, which is abnormal behavior for the database that was not seen during the last 30 days. An example for SQL client anomalous activity can be a spike of failed logins/queries, high volume of data being extracted, unusual canonical queries, and unfamiliar IP addresses used to access the database

Application Gateway Web Application Firewall

[Web Application Firewall](#) is a feature of [Azure Application Gateway](#) that provides protection to web applications that use application gateway for standard [Application Delivery Control](#) functions. Web application firewall does this by protecting them against most of the [OWASP top 10 common web vulnerabilities](#)



- SQL injection protection
- Cross site scripting protection
- Common Web Attacks Protection such as command injection, HTTP request smuggling, HTTP response splitting, and remote file inclusion attack
- Protection against HTTP protocol violations
- Protection against HTTP protocol anomalies such as missing host user-agent and accept headers
- Prevention against bots, crawlers, and scanners
- Detection of common application misconfigurations (that is, Apache, IIS, etc.)

Configuring WAF at Application Gateway provides the following benefit to you:

- Protect your web application from web vulnerabilities and attacks without modification to backend code.
- Protect multiple web applications at the same time behind an application gateway. Application gateway supports hosting up to 20 websites behind a single gateway that could all be protected against web attacks.

- Monitor your web application against attacks using real-time report generated by application gateway WAF logs.
- Certain compliance controls require all internet facing end points to be protected by a WAF solution. By using application gateway with WAF enabled, you can meet these compliance requirements.

Anomaly Detection – an API built with Azure Machine Learning

Anomaly Detection is an API built with Azure Machine Learning that is useful for detecting different types of anomalous patterns in your time series data. The API assigns an anomaly score to each data point in the time series, which can be used for generating alerts, monitoring through dashboards or connecting with your ticketing systems.

The [Anomaly Detection API](#) can detect the following types of anomalies on time series data:

- **Spikes and Dips:** For example, when monitoring the number of login failures to a service or number of checkouts in an e-commerce site, unusual spikes or dips could indicate security attacks or service disruptions.
- **Positive and negative trends:** When monitoring memory usage in computing, for instance, shrinking free memory size is indicative of a potential memory leak; when monitoring service queue length, a persistent upward trend may indicate an underlying software issue.
- **Level changes and changes in dynamic range of values:** For example, level changes in latencies of a service after a service upgrade or lower levels of exceptions after upgrade can be interesting to monitor.

The machine learning based API enables:

- **Flexible and robust detection:** The anomaly detection models allow users to configure sensitivity settings and detect anomalies among seasonal and non-seasonal data sets. Users can adjust the anomaly detection model to make the detection API less or more sensitive according to their needs. This would mean detecting the less or more visible anomalies in data with and without seasonal patterns.
- **Scalable and timely detection:** The traditional way of monitoring with present thresholds set by experts' domain knowledge are costly and not scalable to millions of dynamically changing data sets. The anomaly detection models in this API are learned and models are tuned automatically from both historical and real-time data.
- **Proactive and actionable detection:** Slow trend and level change detection can be applied for early anomaly detection. The early abnormal signals detected can be used to direct humans to investigate and act on the problem areas. In addition, root cause analysis models and alerting tools can be developed on top of this anomaly detection API service.

The anomaly detection API is an effective and efficient solution for a wide range of scenarios like service health & KPI monitoring, IoT, performance monitoring, and network traffic monitoring. Here are some popular scenarios where this API can be useful:

- IT departments need tools to track events, error code, usage log, and performance (CPU, Memory and so on) in a timely manner.
- Online commerce sites want to track customer activities, page views, clicks, and so on.
- Utility companies want to track consumption of water, gas, electricity, and other resources.
- Facility/Building management services want to monitor temperature, moisture, traffic, and so on.
- IoT/manufacturers want to use sensor data in time series to monitor work flow, quality, and so on.
- Service providers, such as call centers need to monitor service demand trend, incident volume, wait queue length and so on.

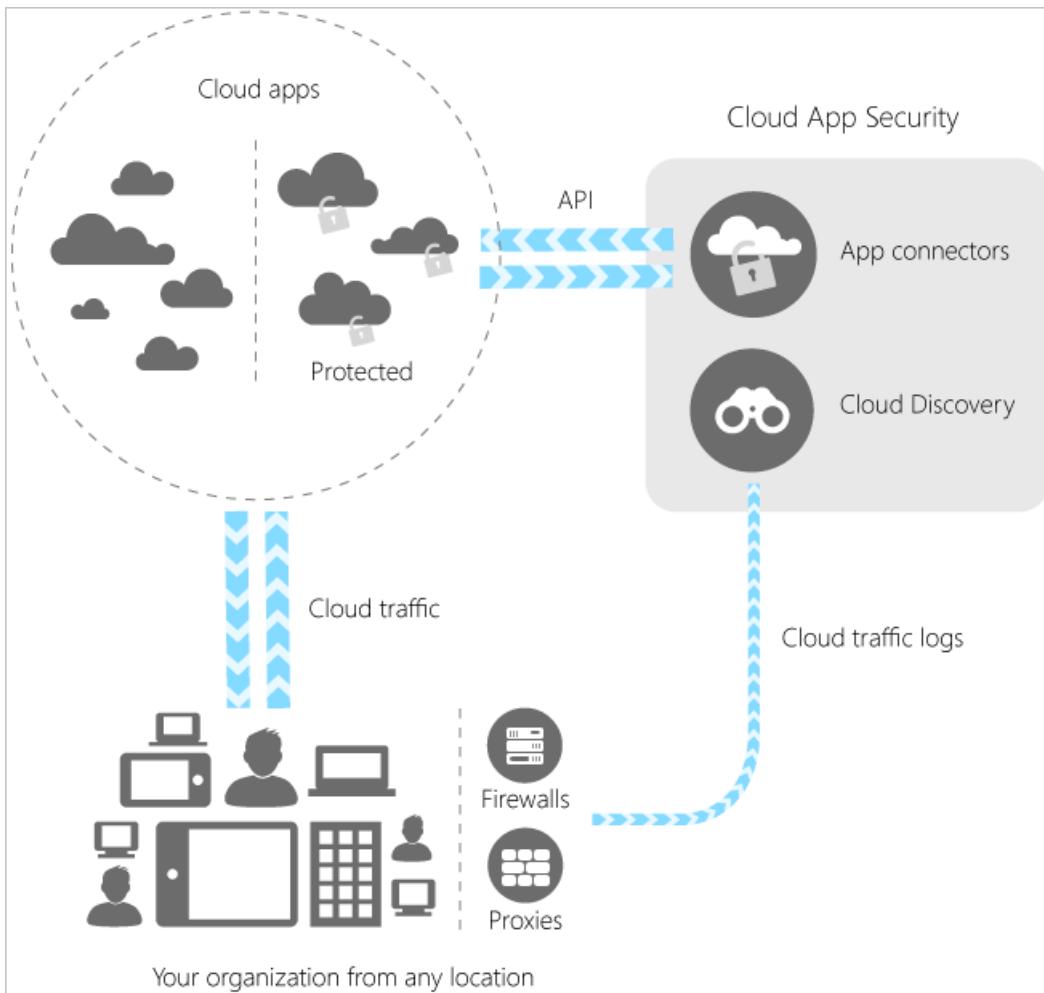
- Business analytics groups want to monitor business KPIs' (such as sales volume, customer sentiments, pricing) abnormal movement in real time.

Cloud App Security

[Cloud App Security](#) is a critical component of the Microsoft Cloud Security stack. It's a comprehensive solution that can help your organization as you move to take full advantage of the promise of cloud applications, but keep you in control, through improved visibility into activity. It also helps increase the protection of critical data across cloud applications.

With tools that help uncover shadow IT, assess risk, enforce policies, investigate activities, and stop threats, your organization can more safely move to the cloud while maintaining control of critical data.

Discover	Uncover shadow IT with Cloud App Security. Gain visibility by discovering apps, activities, users, data, and files in your cloud environment. Discover third-party apps that are connected to your cloud.
Investigate	Investigate your cloud apps by using cloud forensics tools to deep-dive into risky apps, specific users, and files in your network. Find patterns in the data collected from your cloud. Generate reports to monitor your cloud.
Control	Mitigate risk by setting policies and alerts to achieve maximum control over network cloud traffic. Use Cloud App Security to migrate your users to safe, sanctioned cloud app alternatives.
Protect	Use Cloud App Security to sanction or prohibit applications, enforce data loss prevention, control permissions and sharing, and generate custom reports and alerts.
Control	Mitigate risk by setting policies and alerts to achieve maximum control over network cloud traffic. Use Cloud App Security to migrate your users to safe, sanctioned cloud app alternatives.



Cloud App Security integrates visibility with your cloud by

- Using Cloud Discovery to map and identify your cloud environment and the cloud apps your organization is using.
- Sanctioning and prohibiting apps in your cloud.
- Using easy-to-deploy app connectors that take advantage of provider APIs, for visibility and governance of apps that you connect to.
- Helping you have continuous control by setting, and then continually fine-tuning, policies.

On collecting data from these sources, Cloud App Security runs sophisticated analysis on the data. It immediately alerts you to anomalous activities, and gives you deep visibility into your cloud environment. You can configure a policy in Cloud App Security and use it to protect everything in your cloud environment.

Third-party ATD capabilities through Azure Marketplace

Web Application Firewall

Web Application Firewall inspects inbound web traffic and blocks SQL injections, Cross-Site Scripting, malware uploads & application DDoS and other attacks targeted at your web applications. It also inspects the responses from the back-end web servers for Data Loss Prevention (DLP). The integrated access control engine enables administrators to create granular access control policies for Authentication, Authorization & Accounting (AAA), which gives organizations strong authentication and user control.

Highlights:

- Detects and blocks SQL injections, Cross-Site Scripting, malware uploads, application DDoS, or any other attacks against your application.

- Authentication and access control.
- Scans outbound traffic to detect sensitive data and can mask or block the information from being leaked out.
- Accelerates the delivery of web application contents, using capabilities such as caching, compression, and other traffic optimizations.

Following are example of Web Application firewalls available in Azure Market Place:

[Barracuda Web Application Firewall](#), [Brocade Virtual Web Application Firewall \(Brocade vWAF\)](#), [Imperva SecureSphere](#) & [The ThreatSTOP IP Firewall](#).

Next Steps

- [Azure Security Center detection capabilities](#)

Azure Security Center's advanced detection capabilities helps to identify active threats targeting your Microsoft Azure resources and provides you with the insights needed to respond quickly.

- [Azure SQL Database Threat Detection](#)

Azure SQL Database Threat Detection helped address their concerns about potential threats to their database.

Azure Logging and Auditing

6/27/2017 • 23 min to read • [Edit Online](#)

Introduction

Overview

To assist current and prospective Azure customers in understanding and using the various security-related capabilities available in and surrounding the Azure Platform, Microsoft has developed a series of white papers, security overviews, best practices, and checklists. The topics range in terms of breadth and depth and are updated periodically. This document is part of that series as summarized in the following Abstract section.

Azure Platform

Azure is an open and flexible cloud service platform that supports the broadest selection of operating systems, programming languages, frameworks, tools, databases, and devices.

For example, you can:

- Run Linux containers with Docker integration.
- Build apps with JavaScript, Python, .NET, PHP, Java, and Node.js
- Build back-ends for iOS, Android, and Windows devices.

Azure public cloud services support the same technologies millions of developers and IT professionals already rely on and trust.

When you build on, or migrate IT assets to, a cloud provider, you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security needs. In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your deployments. This document will help you meet these requirements.

Abstract

Auditing and logging of security-related events, and related alerts, are important components in an effective data protection strategy. Security logs and reports provide you with an electronic record of suspicious activities and help you detect patterns that may indicate attempted or successful external penetration of the network, as well as internal attacks. You can use auditing to monitor user activity, document regulatory compliance, perform forensic analysis, and more. Alerts provide immediate notification when security events occur.

Microsoft Azure services and products provide you with configurable security auditing and logging options to help you identify gaps in your security policies and mechanisms, and address those gaps to help prevent breaches.

Microsoft services offer some (and in some cases, all) of the following options: centralized monitoring, logging, and analysis systems to provide continuous visibility; timely alerts; and reports to help you manage the large amount of information generated by devices and services.

Microsoft Azure log data can be exported to Security Incident and Event Management (SIEM) systems for analysis and integrates with third-party auditing solutions.

This whitepaper provides an introduction for generating, collecting, and analyzing security logs from services

hosted on Azure, and it can help you gain security insights into your Azure deployments. The scope of this white paper is limited to applications and services built and deployed in Azure.

NOTE

Certain recommendations contained herein may result in increased data, network, or compute resource usage, and increase your license or subscription costs.

Types of logs in Azure

Cloud applications are complex with many moving parts. Logs provide data to ensure that your application stays up and running in a healthy state. It also helps you to stave off potential problems or troubleshoot past ones. In addition, you can use logging data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention.

Azure produces extensive logging for every Azure service. These logs are categorized by these main types:

- **Control/management logs** give visibility into the Azure Resource Manager CREATE, UPDATE, and DELETE operations. [Azure Activity Logs](#) is an example of this type of log.
- **Data plane logs** give visibility into the events raised as part of the usage of an Azure resource. Examples of this type of log are the Windows event System, Security, and Application logs in a virtual machine and the [Diagnostics Logs](#) configured through Azure Monitor
- **Processed events** give information about analyzed events/alerts that have been processed on your behalf. Examples of this type are [Azure Security Center Alerts](#) where [Azure Security Center](#) has processed and analyzed your subscription and provides concise security alerts

The following table lists most important type of logs available in Azure.

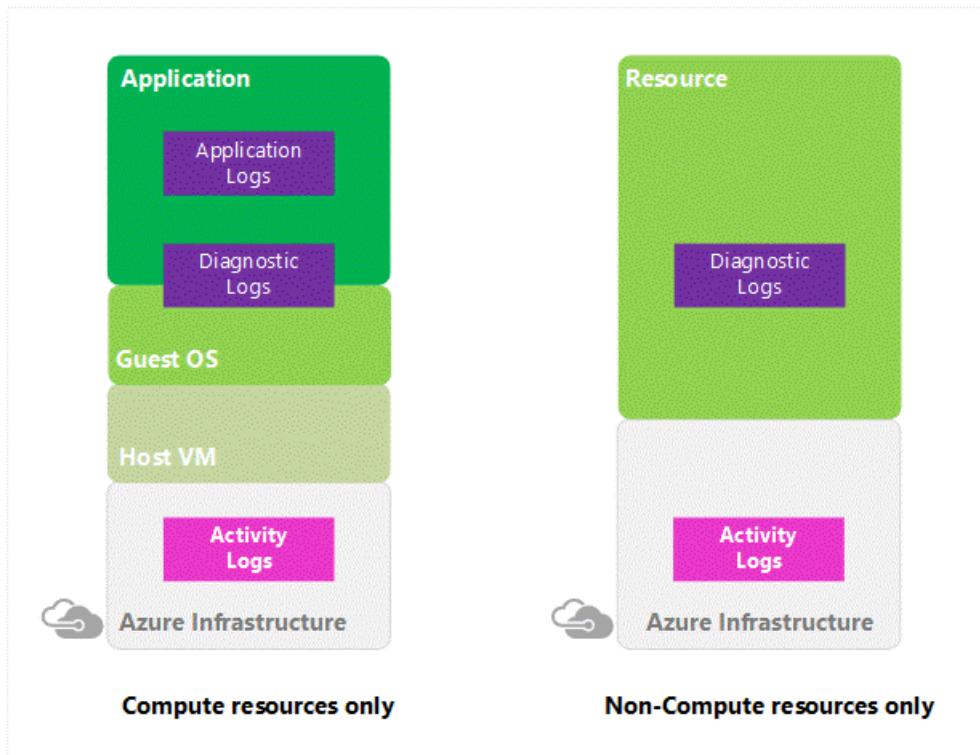
LOG CATEGORY	LOG TYPE	USAGES	INTEGRATION
Activity Logs	Control-plane events on Azure Resource Manager resources	Provide insight into the operations that were performed on resources in your subscription.	Rest API & Azure Monitor
Azure Diagnostic Logs	frequent data about the operation of Azure Resource Manager resources in subscription	Provide insight into operations that your resource performed itself	Azure Monitor, Stream
AAD Reporting	Logs and Reports	User sign-in activities & System activity information about users and group management	Graph API
Virtual Machine & Cloud Services	Windows Event log & Linux Syslog	Captures system data and logging data on the virtual machines and transfers that data into a storage account of your choice.	Windows using WAD (Windows Azure Diagnostics storage) and Linux in Azure monitor

LOG CATEGORY	LOG TYPE	USAGES	INTEGRATION
Storage Analytics	Storage logging and provides metrics data for a storage account	Provides insight into trace requests, analyze usage trends, and diagnose issues with your storage account.	REST API or the client library
NSG (Network Security Group) Flow Logs	JSON format and shows outbound and inbound flows on a per rule basis	View information about ingress and egress IP traffic through a Network Security Group	Network Watcher
Application insight	Logs, exceptions, and custom diagnostics	Application Performance Management (APM) service for web developers on multiple platforms.	REST API, Power BI
Process Data / Security Alert	Azure Security Center Alert, OMS Alert	Security information and alerts.	REST APIs, JSON

Activity Log

The [Azure Activity Log](#), provides insight into the operations that were performed on resources in your subscription. The Activity Log was previously known as "Audit Logs" or "Operational Logs," since it reports [control-plane events](#) for your subscriptions. Using the Activity Log, you can determine the "what, who, and when" for any write operations (PUT, POST, DELETE) taken on the resources in your subscription. You can also understand the status of the operation and other relevant properties. The Activity Log does not include read (GET) operations.

Here PUT, POST, DELETE refers to all the write operations activity log contains on the resources. For example, you can use the activity logs to find an error when troubleshooting or to monitor how a user in your organization modified a resource.



You can retrieve events from your Activity Log using the Azure portal, [CLI](#), PowerShell cmdlets, and [Azure Monitor REST API](#). Activity logs have 19-day data retention period.

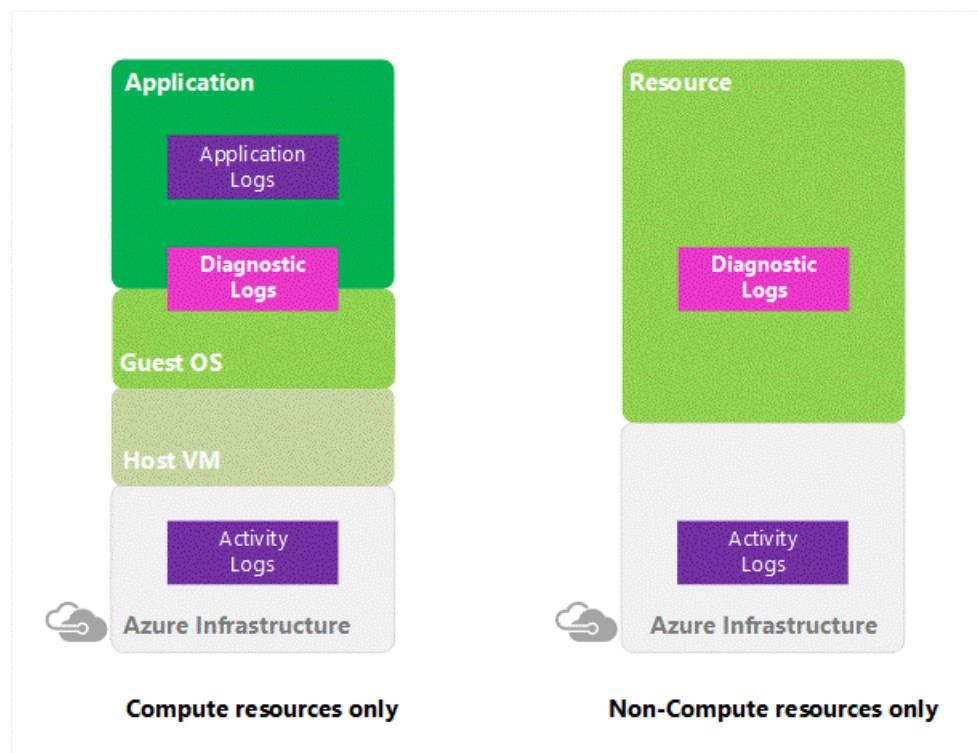
Integration Scenarios

- Create an email or webhook alert that triggers off an Activity Log event.
- Stream it to an Event Hub for ingestion by a third-party service or custom analytics solution such as PowerBI.
- Analyze it in PowerBI using the [PowerBI content pack](#).
- Save it to a Storage Account for archival or manual inspection. You can specify the retention time (in days) using Log Profiles.
- Query and view it in the Azure portal.
- Query it via PowerShell Cmdlet, CLI, or REST API.
- Export the Activity Log with Log Profiles to [log Analytics](#).

You can use a storage account or [event hub namespace](#) that is not in the same subscription as the one emitting log. The user who configures the setting must have the appropriate [RBAC](#) access to both subscriptions

Azure Diagnostic Logs

Azure Diagnostic Logs are emitted by a resource that provide rich, frequent data about the operation of that resource. The content of these logs varies by resource type (for example, [Windows event system logs](#) are one category of Diagnostic Log for VMs and [blob, table, and queue logs](#) are categories of Diagnostic Logs for storage accounts) and differ from the Activity Log, which provides insight into the operations that were performed on resources in your subscription.



Azure Diagnostics logs offer multiple configuration options that is,Azure portal, using PowerShell, Command-line interface (CLI),and REST API.

Integration Scenarios

- Save them to a [Storage Account](#) for auditing or manual inspection. You can specify the retention time (in days) using the Diagnostic Settings.
- Stream them to [Event Hubs](#) for ingestion by a third-party service or custom analytics solution such as [PowerBI](#).
- Analyze them with [OMS Log Analytics](#).

Supported services, schema for Diagnostic Logs and supported log categories per resource type

Service	Schema & Docs	Resource Type	Category
Load Balancer	Log analytics for Azure Load Balancer (Preview)	Microsoft.Network/loadBalancers	LoadBalancerAlertEvent
		Microsoft.Network/loadBalancers	LoadBalancerProbeHealthStatus
Network Security Groups	Log analytics for network security groups (NSGs)	Microsoft.Network/networkSecurityGroups	NetworkSecurityGroupEvent
		Microsoft.Network/networkSecurityGroups	NetworkSecurityGroupRuleCounter
Application Gateways	Diagnostics Logging for Application Gateway	Microsoft.Network/applicationGateways	ApplicationGatewayAccessLog
		Microsoft.Network/applicationGateways	ApplicationGatewayPerformanceLog
		Microsoft.Network/applicationGateways	ApplicationGatewayFirewallLog
Key Vault	Azure Key Vault Logging	Microsoft.KeyVault/vaults	AuditEvent
Azure Search	Enabling and using Search Traffic Analytics	Microsoft.Search/searchServices	OperationLogs
Data Lake Store	Accessing diagnostic logs for Azure Data Lake Store	Microsoft.DataLakeStore/accounts	Audit
Data Lake Analytics	Accessing diagnostic logs for Azure Data Lake Analytics	Microsoft.DataLakeAnalytics/accounts	Audit
		Microsoft.DataLakeAnalytics/accounts	Requests
		Microsoft.DataLakeStore/accounts	Requests
Logic Apps	Logic Apps B2B custom tracking schema	Microsoft.Logic/workflows	WorkflowRuntime
		Microsoft.Logic/integrationAccounts	IntegrationAccountTrackingEvents
Azure Batch	Azure Batch diagnostic logging	Microsoft.Batch/batchAccounts	ServiceLog
Azure Automation	Log analytics for Azure Automation	Microsoft.Automation/automationAccounts	JobLogs
		Microsoft.Automation/automationAccounts	JobStreams

Service	Schema & Docs	Resource Type	Category
Event Hubs	Azure Event Hubs diagnostic logs	Microsoft.EventHub/namespaces	ArchiveLogs
		Microsoft.EventHub/namespaces	OperationalLogs
Stream Analytics	Job diagnostic logs	Microsoft.StreamAnalytics/streamingjobs	Execution
		Microsoft.StreamAnalytics/streamingjobs	Authoring
Service Bus	Azure Service Bus diagnostic logs	Microsoft.ServiceBus/namespaces	OperationalLogs

Azure Active Directory Reporting

Azure Active Directory (Azure AD) includes security, activity, and audit reports for your directory. The [Azure Active Directory Audit Report](#) helps customers to identify privileged actions that occurred in their Azure Active Directory. Privileged actions include elevation changes (for example, role creation or password resets), changing policy configurations (for example password policies), or changes to directory configuration (for example, changes to domain federation settings).

The reports provide the audit record for the event name, the actor who performed the action, the target resource affected by the change, and the date and time (in UTC). Customers are able to retrieve the list of audit events for their Azure Active Directory via the [Azure portal](#), as described in [View your Audit Logs](#). Here's a list of the reports included:

Security Reports	Activity Reports	Audit Reports
Sign-ins from unknown sources	Application usage: summary	Directory audit report
Sign-ins after multiple failures	Application usage: detailed	
Sign-ins from multiple geographies	Application dashboard	
Sign-ins from IP addresses with suspicious activity	Account provisioning errors	
Irregular sign-in activity	Individual user devices	
Sign-ins from possibly infected devices	Individual user Activity	
Users with anomalous sign-in activity	Groups activity report	
	Password Reset Registration Activity Report	
	Password reset activity	

The data of these reports can be useful to your applications, such as SIEM systems, audit, and business intelligence tools. The Azure AD reporting APIs provide programmatic access to the data through a set of REST-based APIs. You can call these [APIs](#) from various programming languages and tools.

Events in the Azure AD Audit report are retained for 180 days.

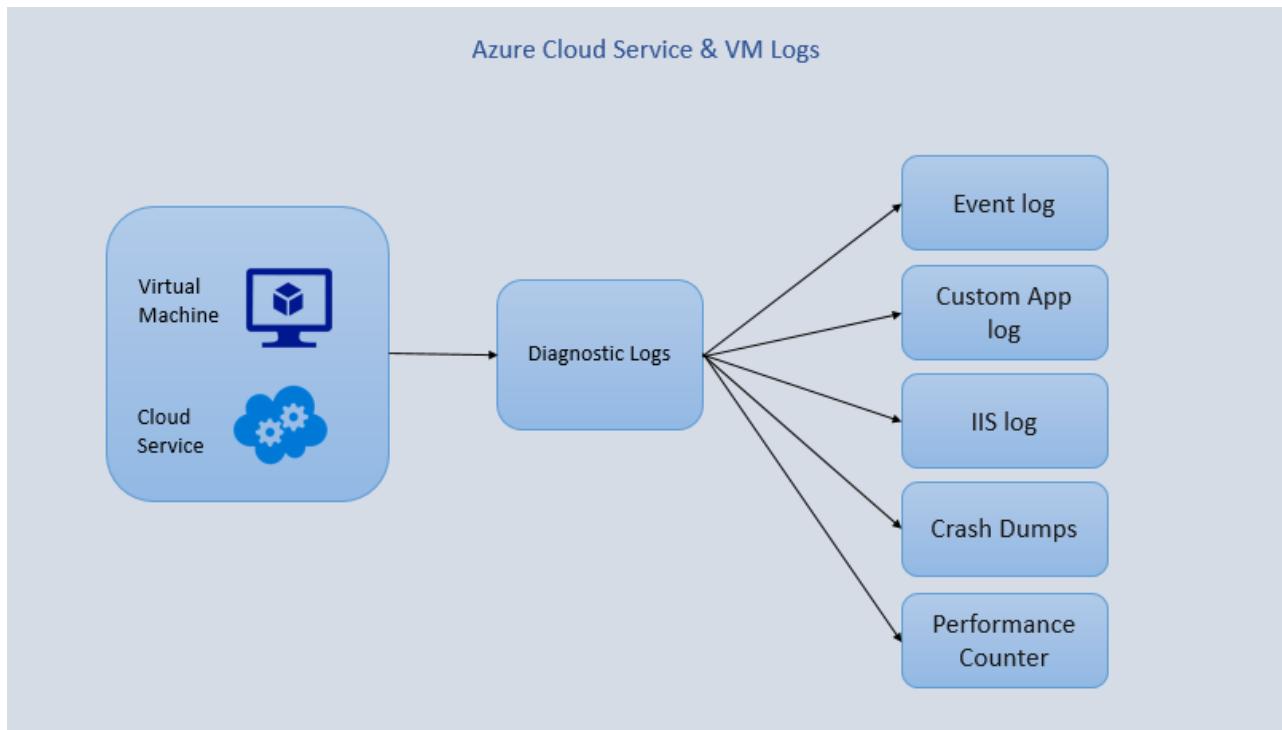
NOTE

For more information about retention on reports, see [Azure Active Directory Report Retention Policies](#).

For customers interested in storing their audit events for longer retention periods, the Reporting API can be used to regularly pull [audit events](#) into a separate data store.

Virtual Machine logs using Azure Diagnostics

[Azure Diagnostics](#) is the capability within Azure that enables the collection of diagnostic data on a deployed application. You can use the diagnostics extension from several different sources. Currently supported are [Azure Cloud Service Web and Worker Roles](#),



[Azure Virtual Machines](#) running Microsoft Windows and [Service Fabric](#).

You can enable Azure Diagnostic on a virtual machine using following:

- Using Visual Studio, see [Use Visual Studio to trace Azure Virtual Machines](#)
- [Set up Azure Diagnostics on an Azure Virtual Machine Remotely](#)
- [Use PowerShell to set up diagnostics on Azure Virtual Machines](#)
- [Create a Windows Virtual machine with monitoring and diagnostics using Azure Resource Manager Template](#)

Storage Analytics

[Azure Storage Analytics](#) performs logging and provides metrics data for a storage account. You can use this data to trace requests, analyze usage trends, and diagnose issues with your storage account. Storage Analytics logging is available for the [Blob, Queue, and Table services](#). Storage Analytics logs detailed information about successful and failed requests to a storage service.

This information can be used to monitor individual requests and to diagnose issues with a storage service.

Requests are logged on a best-effort basis. Log entries are created only if there are requests made against the service endpoint. For example, if a storage account has activity in its Blob endpoint but not in its Table or Queue

endpoints, only logs pertaining to the Blob service is created.

To use Storage Analytics, you must enable it individually for each service you want to monitor. You can enable it in the [Azure portal](#); for details, see [Monitor a storage account in the Azure portal](#). You can also enable Storage Analytics programmatically via the REST API or the client library. Use the Set Service Properties operation to enable Storage Analytics individually for each service.

The aggregated data is stored in a well-known blob (for logging) and in well-known tables (for metrics), which may be accessed using the Blob service and Table service APIs.

Storage Analytics has a 20-TB limit on the amount of stored data that is independent of the total limit for your storage account. All logs are stored in [block blobs](#) in a container named \$logs, which are automatically created when Storage Analytics is enabled for a storage account.

NOTE

For more information on billing and data retention policies, see [Storage Analytics and Billing](#).

NOTE

For more information on storage account limits, see [Azure Storage Scalability and Performance Targets](#).

The following types of authenticated and anonymous requests are logged.

AUTHENTICATED	ANONYMOUS
Successful requests	Successful requests
Failed requests, including timeout, throttling, network, authorization, and other errors	Requests using a Shared Access Signature (SAS), including failed and successful requests
Requests using a Shared Access Signature (SAS), including failed and successful requests	Time out errors for both client and server
Requests to analytics data	Failed GET requests with error code 304 (Not Modified)
Requests made by Storage Analytics itself, such as log creation or deletion, are not logged. A full list of the logged data is documented in the Storage Analytics Logged Operations and Status Messages and Storage Analytics Log Format topics.	All other failed anonymous requests are not logged. A full list of the logged data is documented in the Storage Analytics Logged Operations and Status Messages and Storage Analytics Log Format .

Azure networking logs

Network logging and monitoring in Azure is comprehensive and covers two broad categories:

- [Network Watcher](#) - Scenario-based network monitoring is provided with the features in Network Watcher. This service includes packet capture, next hop, IP flow verify, security group view, NSG flow logs. Scenario level monitoring provides an end to end view of network resources in contrast to individual network resource monitoring.
- [Resource monitoring](#) - Resource level monitoring comprises of four features, diagnostic logs, metrics, troubleshooting, and resource health. All these features are built at the network resource level.

The screenshot shows the Network Watcher - NSG flow logs interface. On the left, there's a navigation sidebar with sections like Overview, MONITORING (Topology), NETWORK DIAGNOSTIC TOOLS (IP flow verify, Next hop, Security group view, Packet capture), METRICS (Network subscription limit), LOGS (NSG flow logs, Diagnostic logs), and a search bar at the top. The main area has a header with 'Subscription' set to Microsoft Azure, 'Resource group' dropdown, 'Resource type' dropdown (set to 0 selected), and 'Type to start fl...'. Below this is a message: 'You can download flow logs from configured storage accounts.' A table lists network security groups:

NAME	RESOURCE TYPE	RESOURCE GROUP	STATUS	STORAGE ACCOUNT
webtestnsg-c3dxj32iloqq-	Network security group	ContosoAppGateway	Disabled	
webtestnsg-h7tfpj4hd...	Network security group	ContosoAppGateway	Enabled	webtestvhdc3dxj32iloqqo
fabrikmvm1-nsg	Network security group	FabrikamRG	Disabled	
fabrikamvm3-nsg	Network security group	FabrikamRG	Enabled	webtestvhdc3dxj32iloqqo
fabrikamvm4-nsg	Network security group	FabrikamRG	Disabled	
webtestnsg-r5wpjct4pltz...	Network security group	testresourcegroup	Disabled	
webtestnsg-xqpow6s7bp...	Network security group	testresourcegroup	Disabled	

Network Watcher is a regional service that enables you to monitor and diagnose conditions at a network scenario level in, to, and from Azure. Network diagnostic and visualization tools available with Network Watcher help you understand, diagnose, and gain insights to your network in Azure.

NSG Flow logging - Flow logs for Network Security Groups enable you to capture logs related to traffic that are allowed or denied by the security rules in the group. These flow logs are written in JSON format and show outbound and inbound flows on a per rule basis, the NIC the flow applies to, 5-tuple information about the flow (Source/Destination IP, Source/Destination Port, Protocol), and if the traffic was allowed or denied.

Network Security Group Flow Logging

[Network Security Group flow logs](#) are a feature of Network Watcher that allows you to view information about ingress and egress IP traffic through a Network Security Group. These flow logs are written in JSON format and show outbound and inbound flows on a per rule basis, the NIC the flow applies to, 5-tuple information about the flow (Source/Destination IP, Source/Destination Port, Protocol), and if the traffic was allowed or denied.

While flow logs target Network Security Groups, they are not displayed the same as the other logs. Flow logs are stored only within a storage account.

The same retention policies as seen on other logs apply to flow logs. Logs have a retention policy that can be set from 1 day to 365 days. If a retention policy is not set, the logs are maintained forever.

Diagnostic logs

Periodic and spontaneous events are created by network resources and logged in storage accounts, sent to an Event Hub, or Log Analytics. These logs provide insights into the health of a resource. These logs can be viewed in tools such as Power BI and Log Analytics. To learn how to view diagnostic logs, visit [Log Analytics](#).

The screenshot shows the Microsoft Azure Network Watcher interface for 'Diagnostic logs'. On the left, there's a navigation pane with sections like 'Overview', 'MONITORING' (Topology), 'NETWORK DIAGNOSTIC TOOLS' (IP flow verify, Next hop, Security group view, Packet capture), 'METRICS' (Network subscription limit), and 'LOGS' (NSG flow logs, Diagnostic logs). The 'Diagnostic logs' section is currently selected. The main area displays a table of resources:

NAME	RESOURCE TYPE	RESOURCE GROUP	DIAGNOSTICS STATUS	STORAGE ACCOUNT	EVENT HUB NAMES	LOG ANALYTICS
applicationGateway1	Application gateway	ContosoAppGateway	Disabled			gatetestworkspace
webtestnsg-c3dxj32lloqqo	Network security group	ContosoAppGateway	Enabled			
webtestnsg-h7trpjbl4hdmk	Network security group	ContosoAppGateway	Enabled		fabrikamrgdisks2...	
fabrikamvm1-nsg	Network security group	FabrikamRG	Disabled			
fabrikamvm3-nsg	Network security group	FabrikamRG	Enabled	webtestvhdc3dij...		
fabrikamvm4-nsg	Network security group	FabrikamRG	Disabled			
applicationGateway1	Application gateway	testresourcegroup	Disabled			
webtestnsg-i5wpjct4ptzm	Network security group	testresourcegroup	Disabled			
webtestnsg-xqpow6s7bpsg	Network security group	testresourcegroup	Disabled			

Diagnostic logs are available for [Load Balancer](#), [Network Security Groups](#), Routes, and [Application Gateway](#).

Network Watcher provides a diagnostic logs view. This view contains all networking resources that support diagnostic logging. From this view, you can enable and disable networking resources conveniently and quickly.

In addition to preceding logging capabilities, Network Watcher currently has the following capabilities:

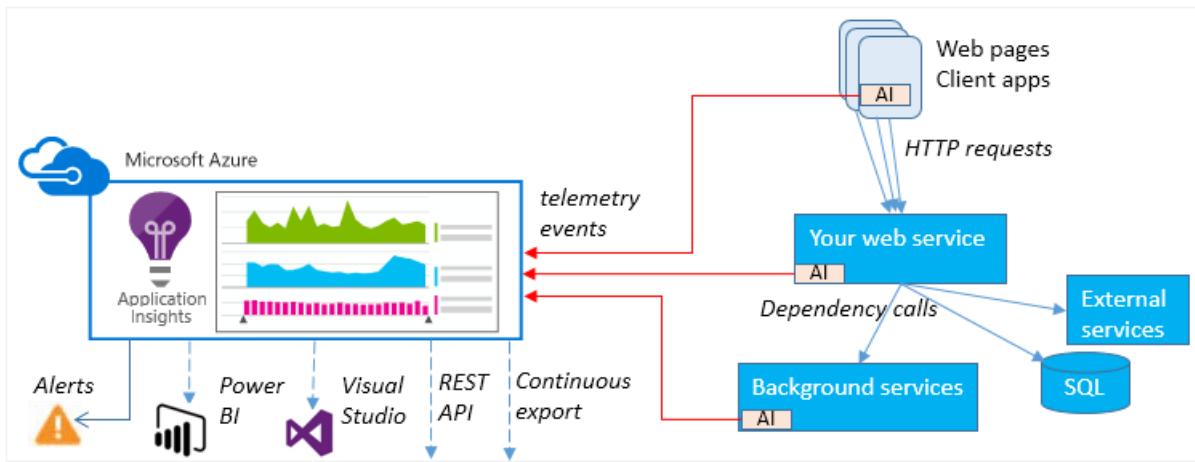
- [Topology](#) - Provides a network level view showing the various interconnections and associations between network resources in a resource group.
- [Variable Packet capture](#) - Captures packet data in and out of a virtual machine. Advanced filtering options and fine-tuned controls such as being able to set time and size limitations provide versatility. The packet data can be stored in a blob store or on the local disk in .cap format.
- [IP flow verifies](#) - Checks if a packet is allowed or denied based on flow information 5-tuple packet parameters (Destination IP, Source IP, Destination Port, Source Port, and Protocol). If the packet is denied by a security group, the rule and group that denied the packet is returned.
- [Next hop](#) - Determines the next hop for packets being routed in the Azure Network Fabric, enabling you to diagnose any misconfigured user-defined routes.
- [Security group view](#) - Gets the effective and applied security rules that are applied on a VM.
- [Virtual Network Gateway and Connection troubleshooting](#) - Provides the ability to troubleshoot Virtual Network Gateways and Connections.
- [Network subscription limits](#) - Enables you to view network resource usage against limits.

Application insight

[Application Insights](#) is an extensible Application Performance Management (APM) service for web developers on multiple platforms. Use it to monitor your live web application. It is automatically detect performance anomalies. It includes powerful analytics tools to help you diagnose issues and to understand what users actually do with your app.

It's designed to help you continuously improve performance and usability.

It works for apps on a wide variety of platforms including .NET, Node.js and J2EE, hosted on-premises or in the cloud. It integrates with your devOps process, and has connection points to various development tools.



Application Insights is aimed at the development team, to help you understand how your app is performing and how it's being used. It monitors:

- **Request rates, response times, and failure rates** - Find out which pages are most popular, at what times of day, and where your users are. See which pages perform best. If your response times and failure rates go high when there are more requests, then perhaps you have a resourcing problem.
- **Dependency rates, response times, and failure rates** - Find out whether external services are slowing you down.
- **Exceptions** - Analyze the aggregated statistics, or pick specific instances and drill into the stack trace and related requests. Both server and browser exceptions are reported.
- **Page views and load performance** - reported by your users' browsers.
- **AJAX calls** from web pages - rates, response times, and failure rates.
- **User and session counts**.
- **Performance counters** from your Windows or Linux server machines, such as CPU, memory, and network usage.
- **Host diagnostics** from Docker or Azure.
- **Diagnostic trace logs** from your app - so that you can correlate trace events with requests.
- **Custom events and metrics** that you write yourself in the client or server code, to track business events such as items sold or games won.

List Of Integration Scenarios and Description:

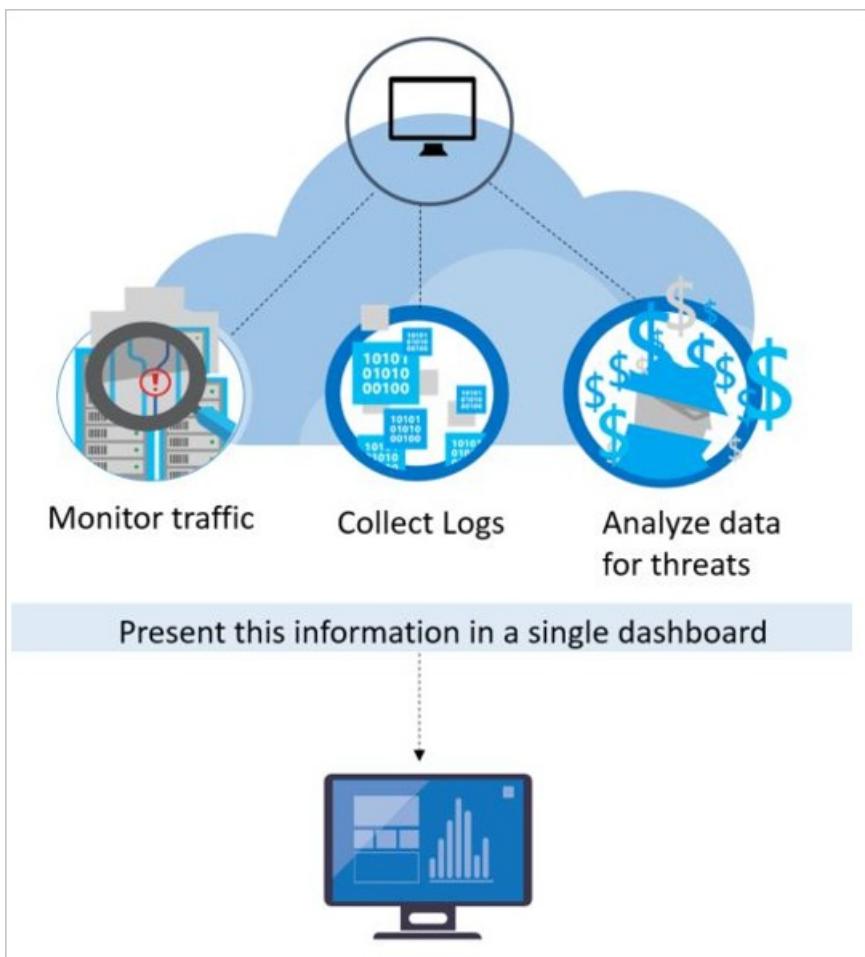
INTEGRATION SCENARIOS	DESCRIPTION
Application map	The components of your app, with key metrics and alerts.
Diagnostic search for instance data	Search and filter events such as requests, exceptions, dependency calls, log traces, and page views.
Metrics Explorer for aggregated data	Explore, filter, and segment aggregated data such as rates of requests, failures, and exceptions; response times, page load times.
Dashboards	Mash up data from multiple resources and share with others. Great for multi-component applications, and for continuous display in the team room.

INTEGRATION SCENARIOS	DESCRIPTION
Live Metrics Stream	When you deploy a new build, watch these near-real-time performance indicators to make sure everything works as expected.
Analytics	Answer tough questions about your app's performance and usage by using this powerful query language.
Automatic and manual alerts	Automatic alerts adapt to your app's normal patterns of telemetry and trigger when there's something outside the usual pattern. You can also set alerts on particular levels of custom or standard metrics.
Visual Studio	See performance data in the code. Go to code from stack traces.
Power BI	Integrate usage metrics with other business intelligence.
REST API	Write code to run queries over your metrics and raw data.
Continuous export	Bulk export of raw data to storage when it arrives.

Azure Security Center Alerts

[Azure Security Center](#) automatically collects, analyzes, and integrates log data from your Azure resources, the network, and connected partner solutions, like firewall and endpoint protection solutions, to detect real threats and reduce false positives. A list of prioritized security alerts is shown in Security Center along with the information you need to quickly investigate the problem and recommendations for how to remediate an attack.

Security Center threat detection works by automatically collecting security information from your Azure resources, the network, and connected partner solutions. It analyzes this information, often correlating information from multiple sources, to identify threats. Security alerts are prioritized in Security Center along with recommendations on how to remediate the threat.



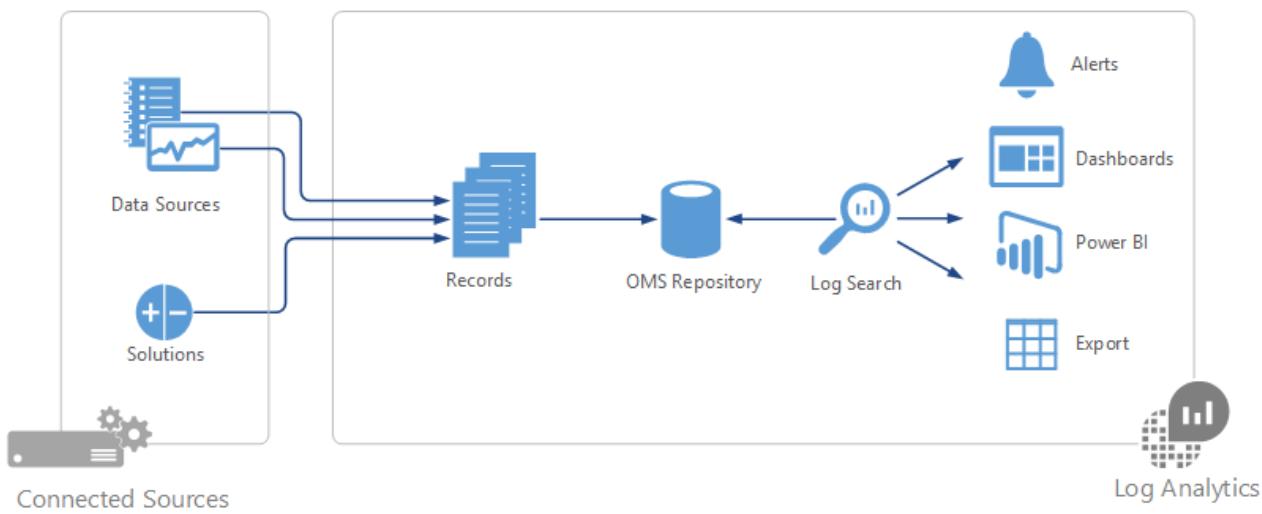
Security Center employs advanced security analytics, which go far beyond signature-based approaches. Breakthroughs in large data and [machine learning](#) technologies are applied to evaluate events across the entire cloud fabric – detecting threats that would be impossible to identify using manual approaches and predicting the evolution of attacks. These security analytics include:

- **Integrated threat intelligence:** looks for known bad actors by applying global threat intelligence from Microsoft products and services, the Microsoft Digital Crimes Unit (DCU), the Microsoft Security Response Center (MSRC), and external feeds.
- **Behavioral analytics:** applies known patterns to discover malicious behavior.
- **Anomaly detection:** uses statistical profiling to build a historical baseline. It alerts on deviations from established baselines that conform to a potential attack vector.

Many security operations and incident response teams rely on a Security Information and Event Management (SIEM) solution as the starting point for triaging and investigating security alerts. With Azure log integration, customers can sync Security Center alerts and virtual machine security events, collected by Azure Diagnostics and Azure Audit Logs, with their log analytics or SIEM solution in near real time.

Log Analytics

Log Analytics is a service in [Operations Management Suite \(OMS\)](#) that helps you collect and analyze data generated by resources in your cloud and on-premises environments. It gives you real-time insights using integrated search and custom dashboards to readily analyze millions of records across all your workloads and servers regardless of their physical location.



At the center of Log Analytics is the OMS repository, which is hosted in the Azure cloud. Data is collected into the repository from connected sources by configuring data sources and adding solutions to your subscription. Data sources and solutions will each create different record types that have their own set of properties but may still be analyzed together in queries to the repository. This allows you to use the same tools and methods to work with different kinds of data collected by different sources.

Connected sources are the computers and other resources that generate data collected by Log Analytics. This can include agents installed on [Windows](#) and [Linux](#) computers that connect directly or agents in [a connected System Center Operations Manager management group](#). Log Analytics can also collect data from [Azure storage](#).

[Data sources](#) are the different kinds of data collected from each connected source. This includes events and [performance data](#) from [Windows](#) and Linux agents in addition to sources such as [IIS logs](#), and [custom text logs](#). You configure each data source that you want to collect, and the configuration is automatically delivered to each connected source.

There are four different ways of [collecting logs and metrics for Azure services](#):

1. Azure diagnostics direct to Log Analytics (Diagnostics in the following table)
2. Azure diagnostics to Azure storage to Log Analytics (Storage in the following table)
3. Connectors for Azure services (Connectors in the following table)
4. Scripts to collect and then post data into Log Analytics (blanks in the following table and for services that are not listed)

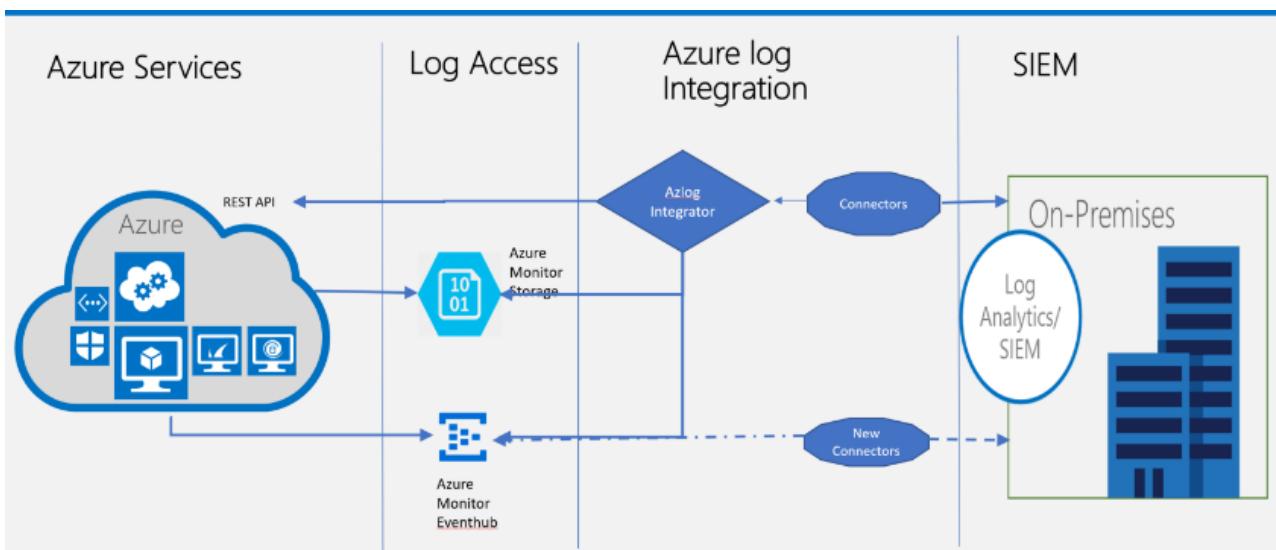
SERVICE	RESOURCE TYPE	LOGS	METRICS	SOLUTION
Application gateways	Microsoft.Network/applicationGateways	Diagnostics	Diagnostics	Azure Application Gateway Analytics
Application insights		Connector	Connector	Application Insights Connector (Preview)
Automation accounts	Microsoft.Automation/AutomationAccounts	Diagnostics		More information
Batch accounts	Microsoft.Batch/batchAccounts	Diagnostics	Diagnostics	
Classic cloud services		Storage		More information

Service	Resource Type	Logs	Metrics	Solution
Cognitive services	Microsoft.CognitiveServices/ accounts	Diagnostics		
Data Lake analytics	Microsoft.DataLakeAnalytics/ accounts	Diagnostics		
Data Lake store	Microsoft.DataLakeStore/ accounts	Diagnostics		
Event Hub namespace	Microsoft.EventHub/ namespaces	Diagnostics	Diagnostics	
IoT Hubs	Microsoft.Devices/ IoTHubs		Diagnostics	
Key Vault	Microsoft.KeyVault/ vaults	Diagnostics		KeyVault Analytics
Load Balancers	Microsoft.Network/ loadBalancers	Diagnostics		
Logic Apps	Microsoft.Logic/ workflows	Diagnostics	Diagnostics	
	Microsoft.Logic/ integrationAccounts			
Network Security Groups	Microsoft.Network/ networkSecurityGroups	Diagnostics		Azure Network Security Group Analytics
Recovery vaults	Microsoft.RecoveryServices/ vaults			Azure Recovery Services Analytics (Preview)
Search services	Microsoft.Search/ searchServices	Diagnostics	Diagnostics	
Service Bus namespace	Microsoft.ServiceBus/ namespaces	Diagnostics	Diagnostics	Service Bus Analytics (Preview)
Service Fabric		Storage		Service Fabric Analytics (Preview)
SQL (v12)	Microsoft.Sql/ servers/ databases		Diagnostics	
	Microsoft.Sql/ servers/ elasticPools			

Service	Resource Type	Logs	Metrics	Solution
Storage			Script	Azure Storage Analytics (Preview)
Virtual Machines	Microsoft.Compute/virtualMachines	Extension	Extension	
			Diagnostics	
Virtual Machines scale sets	Microsoft.Compute/virtualMachines		Diagnostics	
	Microsoft.Compute/virtualMachineScaleSets/virtualMachines			
Web Server farms	Microsoft.Web/serverfarms		Diagnostics	
Web Sites	Microsoft.Web/sites		Diagnostics	More information
	Microsoft.Web/sites/slots			

Log integration with on-premises SIEM systems

[Azure log integration](#) enables you to integrate raw logs from your Azure resources in to your on-premises **Security Information and Event Management (SIEM) systems**.



Azure log integration collects Azure Diagnostics from your Windows (WAD) virtual machines, Azure Activity Logs, Azure Security Center alerts, and Azure Resource Provider logs. This integration provides a unified dashboard for all your assets, on-premises or in the cloud, so that you can aggregate, correlate, analyze, and alert for security events.

Azure log integration currently supports integration of Azure Activity Logs, Windows Event log from Windows virtual machine in your Azure subscription, Azure Security Center alerts, Azure Diagnostic logs and Azure Active Directory audit logs.

LOG TYPE	LOG ANALYTICS SUPPORTING JSON (SPLUNK, ARCSIGHT, QRADAR)
AAD Audit logs	yes
Activity Logs	Yes
ASC Alerts	Yes
Diagnostics Logs (resource logs)	Yes
VM logs	Yes via Forwarded events and not through JSON

The following table explains the Log category and SIEM integration detail.

[Get started with Azure log integration](#) - Tutorial walks you through installation of Azure log integration and integrating logs from Azure WAD storage, Azure Activity Logs, Azure Security Center alerts, and Azure Active Directory audit logs.

Integration Scenarios

- [Partner configuration steps](#) – This blog post shows you how to configure Azure log integration to work with partner solutions Splunk, HP ArcSight, and IBM QRadar.
- [Azure log Integration frequently asked questions \(FAQ\)](#) - This FAQ answers questions about Azure log integration.
- [Integrating Security Center alerts with Azure log Integration](#) – This document shows you how to sync Security Center alerts, along with virtual machine security events collected by Azure Diagnostics and Azure Audit Logs, with your log analytics or SIEM solution.

Next Steps

- [Auditing and logging](#)

Protect data by maintaining visibility and responding quickly to timely security alerts

- [Security Logging and Audit Log Collection within Azure](#)

What settings you need to enforce to make sure your Azure instances are collecting the correct Security and Audit logs.

- [Configure audit settings for a site collection](#)

As a site collection administrator, one can retrieve the history of actions taken by a particular user and can also retrieve the history of actions taken during a particular date range.

- [Search the audit log in the Office 365 Security & Compliance Center](#)

One can use the Office 365 Security & Compliance Center to search the unified audit log to view user and administrator activity in your Office 365 organization.

Isolation in the Azure Public Cloud

8/21/2017 • 22 min to read • [Edit Online](#)

Introduction

Overview

To assist current and prospective Azure customers understand and utilize the various security-related capabilities available in and surrounding the Azure platform, Microsoft has developed a series of White Papers, Security Overviews, Best Practices, and Checklists. The topics range in terms of breadth and depth and are updated periodically. This document is part of that series as summarized in the Abstract section following.

Azure Platform

Azure is an open and flexible cloud service platform that supports the broadest selection of operating systems, programming languages, frameworks, tools, databases, and devices. For example, you can:

- Run Linux containers with Docker integration;
- Build apps with JavaScript, Python, .NET, PHP, Java, and Node.js; and
- Build back-ends for iOS, Android, and Windows devices.

Microsoft Azure supports the same technologies millions of developers and IT professionals already rely on and trust.

When you build on, or migrate IT assets to, a public cloud service provider, you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security needs. In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your deployments. This document helps you meet these requirements.

Abstract

Microsoft Azure allows you to run applications and virtual machines (VMs) on shared physical infrastructure. One of the prime economic motivations to running applications in a cloud environment is the ability to distribute the cost of shared resources among multiple customers. This practice of multi-tenancy improves efficiency by multiplexing resources among disparate customers at low costs. Unfortunately, it also introduces the risk of sharing physical servers and other infrastructure resources to run your sensitive applications and VMs that may belong to an arbitrary and potentially malicious user.

This article outlines how Microsoft Azure provides isolation against both malicious and non-malicious users and serves as a guide for architecting cloud solutions by offering various isolation choices to architects. This white paper focuses on the technology of Azure platform and customer-facing security controls, and does not attempt to address SLAs, pricing models, and DevOps practice considerations.

Tenant Level Isolation

One of the primary benefits of cloud computing is concept of a shared, common infrastructure across numerous customers simultaneously, leading to economies of scale. This concept is called multi-tenancy. Microsoft works continuously to ensure that the multi-tenant architecture of Microsoft Cloud Azure supports security, confidentiality, privacy, integrity, and availability standards.

In the cloud-enabled workplace, a tenant can be defined as a client or organization that owns and manages a specific instance of that cloud service. With the identity platform provided by Microsoft Azure, a tenant is simply a dedicated instance of Azure Active Directory (Azure AD) that your organization receives and owns when it signs up for a Microsoft cloud service.

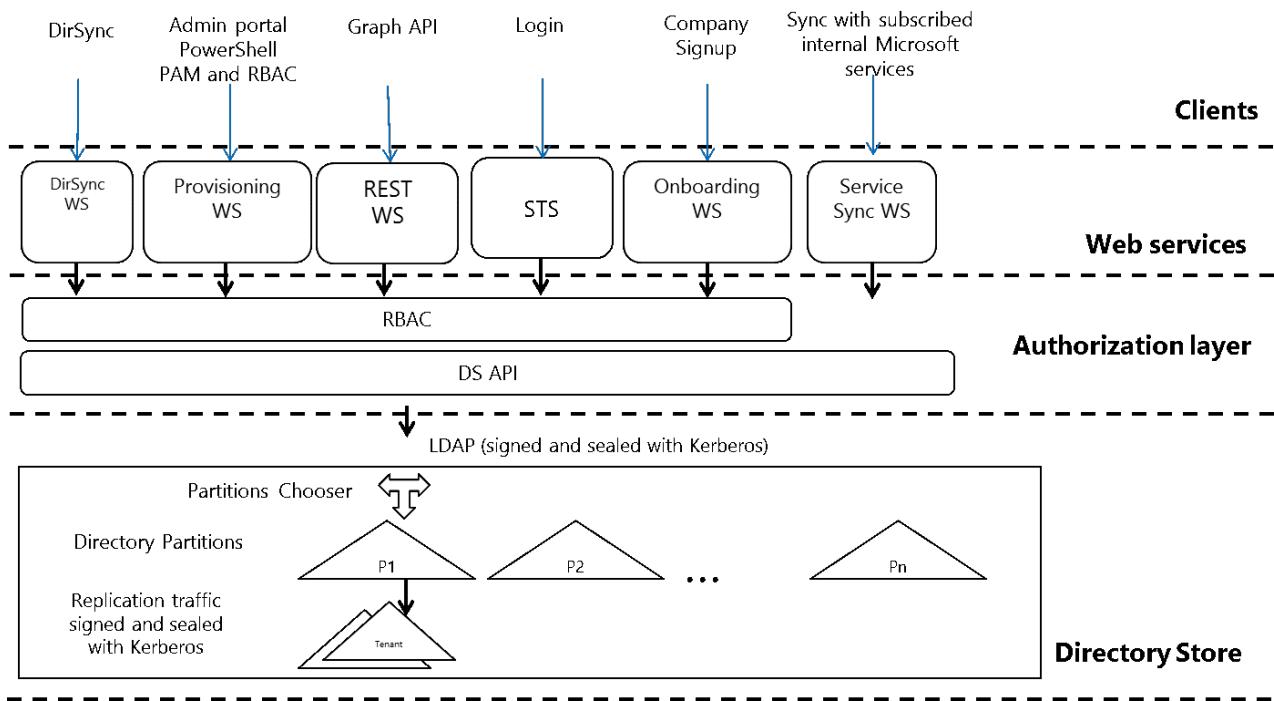
Each Azure AD directory is distinct and separate from other Azure AD directories. Just like a corporate office building is a secure asset specific to only your organization, an Azure AD directory was also designed to be a secure asset for use by only your organization. The Azure AD architecture isolates customer data and identity information from co-mingling. This means that users and administrators of one Azure AD directory cannot accidentally or maliciously access data in another directory.

Azure Tenancy

Azure tenancy (Azure Subscription) refers to a “customer/billing” relationship and a unique [tenant in Azure Active Directory](#). Tenant level isolation in Microsoft Azure is achieved using Azure Active Directory and [role-based controls](#) offered by it. Each Azure subscription is associated with one Azure Active Directory (AD) directory.

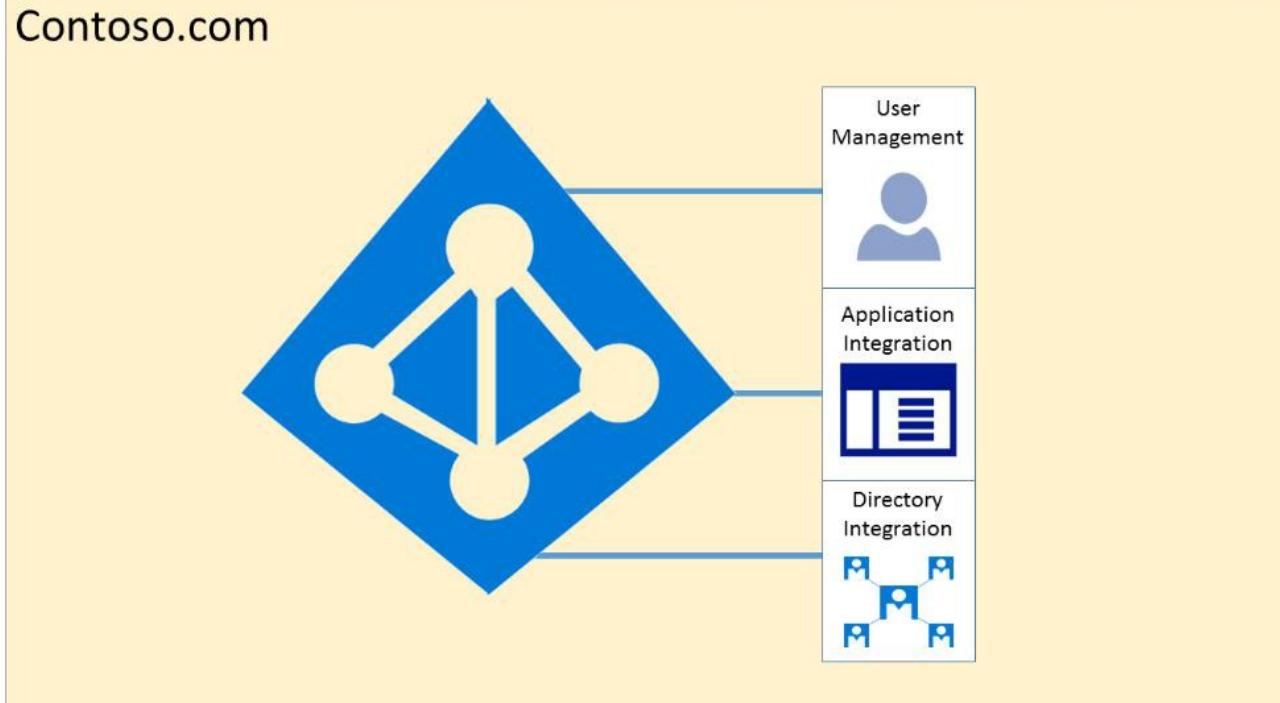
Users, groups, and applications from that directory can manage resources in the Azure subscription. You can assign these access rights using the Azure portal, Azure command-line tools, and Azure Management APIs. An Azure AD tenant is logically isolated using security boundaries so that no customer can access or compromise co-tenants, either maliciously or accidentally. Azure AD runs on “bare metal” servers isolated on a segregated network segment, where host-level packet filtering and Windows Firewall block unwanted connections and traffic.

- Access to data in Azure AD requires user authentication via a [security token service \(STS\)](#). Information on the user’s existence, enabled state, and role is used by the authorization system to determine whether the requested access to the target tenant is authorized for this user in this session.



- Tenants are discrete containers and there is no relationship between these.
- No access across tenants unless tenant admin grants it through federation or provisioning user accounts from other tenants.
- Physical access to servers that comprise the Azure AD service, and direct access to Azure AD’s back-end systems, is restricted.
- Azure AD users have no access to physical assets or locations, and therefore it is not possible for them to bypass the logical RBAC policy checks stated following.

For diagnostics and maintenance needs, an operational model that employs a just-in-time privilege elevation system is required and used. Azure AD Privileged Identity Management (PIM) introduces the concept of an eligible admin. [Eligible admins](#) should be users that need privileged access now and then, but not every day. The role is inactive until the user needs access, then they complete an activation process and become an active admin for a predetermined amount of time.



Azure Active Directory hosts each tenant in its own protected container, with policies and permissions to and within the container solely owned and managed by the tenant.

The concept of tenant containers is deeply ingrained in the directory service at all layers, from portals all the way to persistent storage.

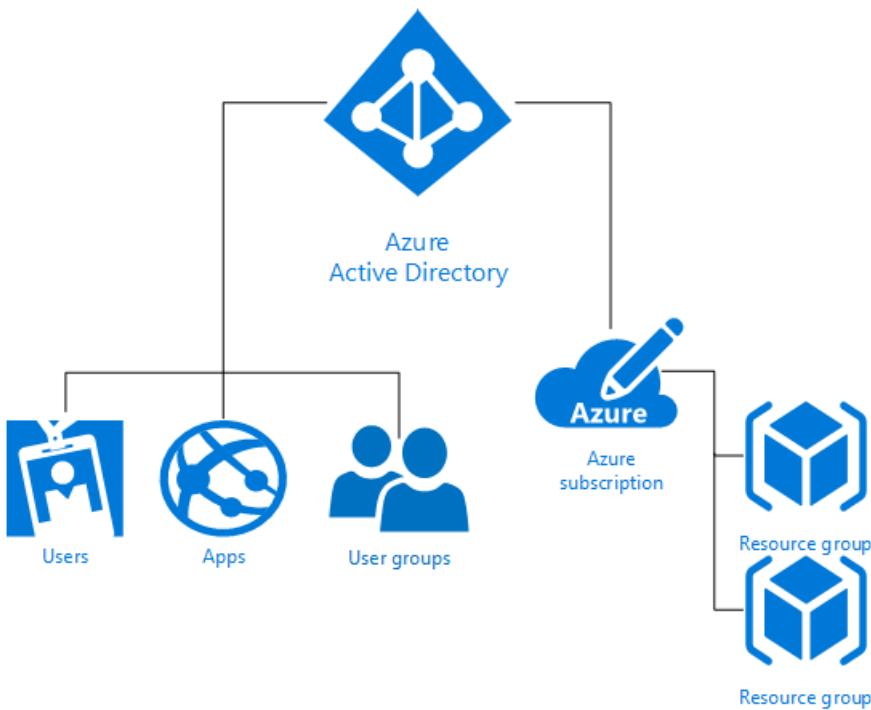
Even when metadata from multiple Azure Active Directory tenants is stored on the same physical disk, there is no relationship between the containers other than what is defined by the directory service, which in turn is dictated by the tenant administrator.

Azure Role-Based Access Control (RBAC)

[Azure Role-Based Access Control \(RBAC\)](#) helps you to share various components available within an Azure subscription by providing fine-grained access management for Azure. Azure RBAC enables you to segregate duties within your organization and grant access based on what users need to perform their jobs. Instead of giving everybody unrestricted permissions in Azure subscription or resources, you can allow only certain actions.

Azure RBAC has three basic roles that apply to all resource types:

- **Owner** has full access to all resources including the right to delegate access to others.
- **Contributor** can create and manage all types of Azure resources but can't grant access to others.
- **Reader** can view existing Azure resources.



The rest of the RBAC roles in Azure allow management of specific Azure resources. For example, the Virtual Machine Contributor role allows the user to create and manage virtual machines. It does not give them access to the Azure Virtual Network or the subnet that the virtual machine connects to.

[RBAC built-in roles](#) list the roles available in Azure. It specifies the operations and scope that each built-in role grants to users. If you're looking to define your own roles for even more control, see how to build [Custom roles in Azure RBAC](#).

Some other capabilities for Azure Active Directory include:

- Azure AD enables SSO to SaaS applications, regardless of where they are hosted. Some applications are federated with Azure AD, and others use password SSO. Federated applications can also support user provisioning and [password vaulting](#).
- Access to data in [Azure Storage](#) is controlled via authentication. Each storage account has a primary key ([storage account key](#), or SAK) and a secondary secret key (the shared access signature, or SAS).
- Azure AD provides Identity as a Service through federation by using [Active Directory Federation Services](#), synchronization, and replication with on-premises directories.
- [Azure Multi-Factor Authentication](#) is the multi-factor authentication service that requires users to verify sign-ins by using a mobile app, phone call, or text message. It can be used with Azure AD to help secure on-premises resources with the Azure Multi-Factor Authentication server, and also with custom applications and directories using the SDK.
- [Azure AD Domain Services](#) lets you join Azure virtual machines to an Active Directory domain without deploying domain controllers. You can sign in to these virtual machines with your corporate Active Directory credentials and administer domain-joined virtual machines by using Group Policy to enforce security baselines on all your Azure virtual machines.
- [Azure Active Directory B2C](#) provides a highly available global-identity management service for consumer-facing applications that scales to hundreds of millions of identities. It can be integrated across mobile and web platforms. Your consumers can sign in to all your applications through customizable experiences by using their existing social accounts or by creating credentials.

Isolation from Microsoft Administrators & Data Deletion

Microsoft takes strong measures to protect your data from inappropriate access or use by unauthorized persons.

These operational processes and controls are backed by the [Online Services Terms](#), which offer contractual commitments that govern access to your data.

- Microsoft engineers do not have default access to your data in the cloud. Instead, they are granted access, under management oversight, only when necessary. That access is carefully controlled and logged, and revoked when it is no longer needed.
- Microsoft may hire other companies to provide limited services on its behalf. Subcontractors may access customer data only to deliver the services for which, we have hired them to provide, and they are prohibited from using it for any other purpose. Further, they are contractually bound to maintain the confidentiality of our customers' information.

Business services with audited certifications such as ISO/IEC 27001 are regularly verified by Microsoft and accredited audit firms, which perform sample audits to attest that access, only for legitimate business purposes. You can always access your own customer data at any time and for any reason.

If you delete any data, Microsoft Azure deletes the data, including any cached or backup copies. For in-scope services, that deletion will occur within 90 days after the end of the retention period. (In-scope services are defined in the Data Processing Terms section of our [Online Services Terms](#).)

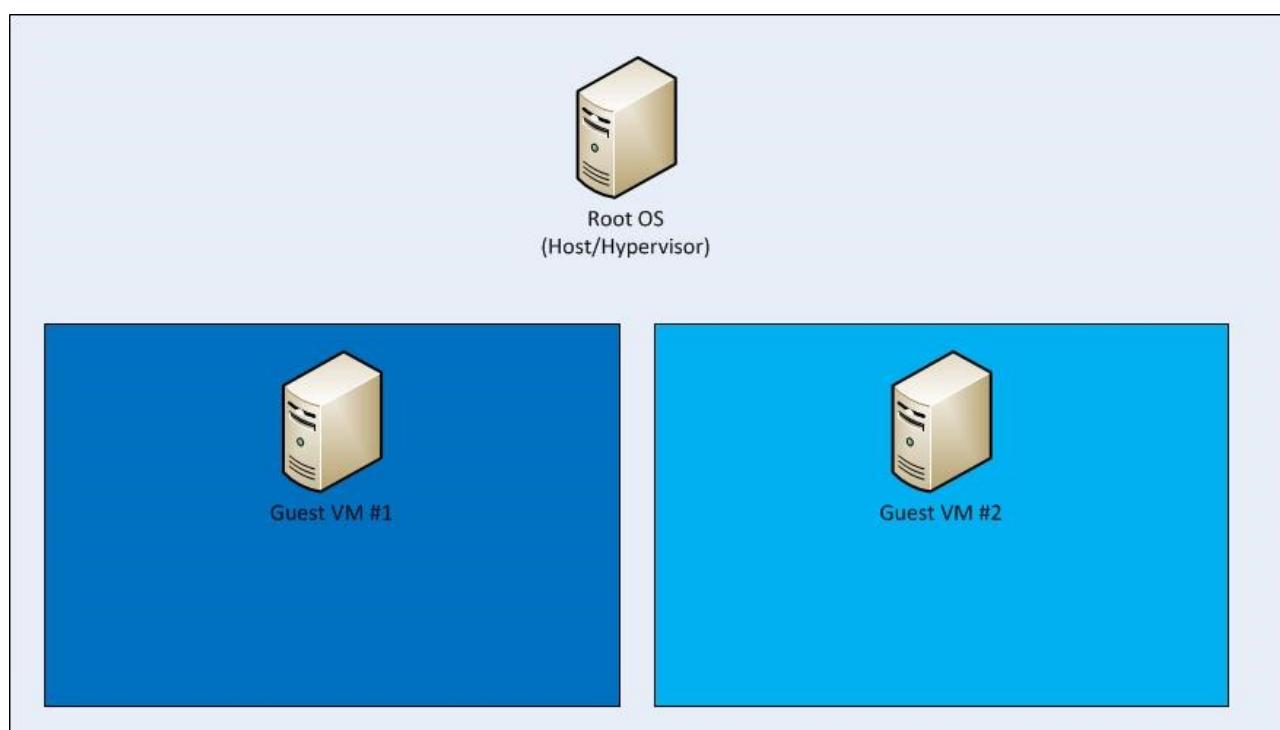
If a disk drive used for storage suffers a hardware failure, it is securely [erased or destroyed](#) before Microsoft returns it to the manufacturer for replacement or repair. The data on the drive is overwritten to ensure that the data cannot be recovered by any means.

Compute Isolation

Microsoft Azure provides various cloud-based computing services that include a wide selection of compute instances & services that can scale up and down automatically to meet the needs of your application or enterprise. These compute instance and service offer isolation at multiple levels to secure data without sacrificing the flexibility in configuration that customers demand.

Hyper-V & Root OS Isolation Between Root VM & Guest VMs

Azure's compute platform is based on machine virtualization—meaning that all customer code executes in a Hyper-V virtual machine. On each Azure node (or network endpoint), there is a Hypervisor that runs directly over the hardware and divides a node into a variable number of Guest Virtual Machines (VMs).



Each node also has one special Root VM, which runs the Host OS. A critical boundary is the isolation of the root VM from the guest VMs and the guest VMs from one another, managed by the hypervisor and the root OS. The hypervisor/root OS pairing leverages Microsoft's decades of operating system security experience, and more recent learning from Microsoft's Hyper-V, to provide strong isolation of guest VMs.

The Azure platform uses a virtualized environment. User instances operate as standalone virtual machines that do not have access to a physical host server, and this isolation is enforced by using physical processor (ring-0/ring-3) privilege levels.

Ring 0 is the most privileged and 3 is the least. The guest OS runs in a lesser-privileged Ring 1, and applications run in the least privileged Ring 3. This virtualization of physical resources leads to a clear separation between guest OS and hypervisor, resulting in additional security separation between the two.

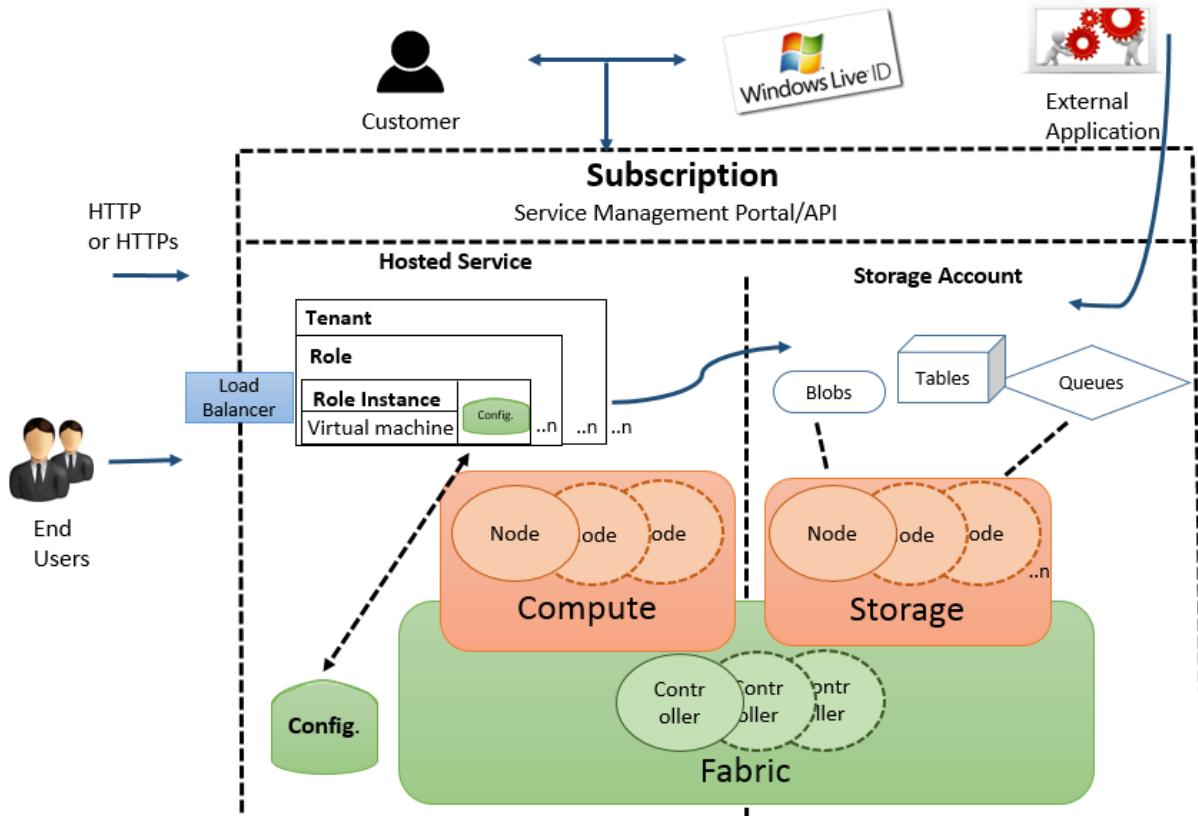
The Azure hypervisor acts like a micro-kernel and passes all hardware access requests from guest virtual machines to the host for processing by using a shared-memory interface called VMBus. This prevents users from obtaining raw read/write/execute access to the system and mitigates the risk of sharing system resources.

Advanced VM placement algorithm & protection from side channel attacks

Any cross-VM attack involves two steps: placing an adversary-controlled VM on the same host as one of the victim VMs, and then breaching the isolation boundary to either steal sensitive victim information or affect its performance for greed or vandalism. Microsoft Azure provides protection at both steps by using an advanced VM placement algorithm and protection from all known side channel attacks including noisy neighbor VMs.

The Azure Fabric Controller

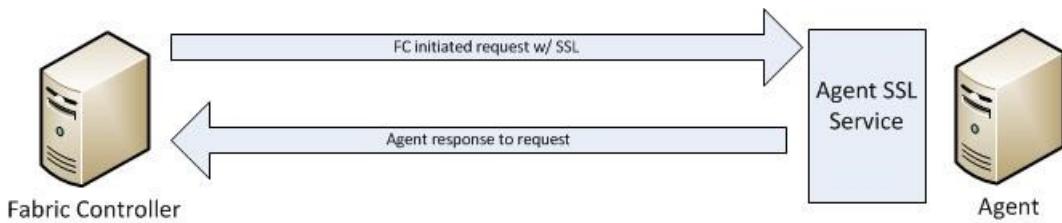
The Azure Fabric Controller is responsible for allocating infrastructure resources to tenant workloads, and it manages unidirectional communications from the host to virtual machines. The VM placing algorithm of the Azure fabric controller is highly sophisticated and nearly impossible to predict at physical host level.



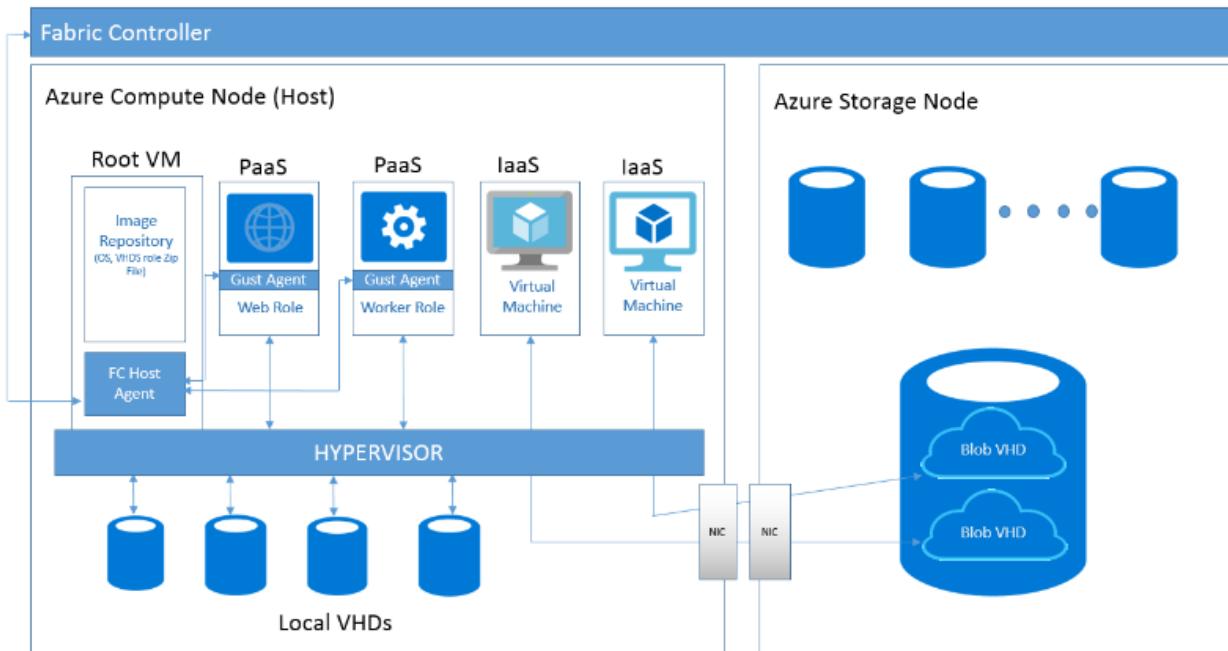
The Azure hypervisor enforces memory and process separation between virtual machines, and it securely routes network traffic to guest OS tenants. This eliminates possibility of and side channel attack at VM level.

In Azure, the root VM is special: it runs a hardened operating system called the root OS that hosts a fabric agent (FA). FAs are used in turn to manage guest agents (GA) within guest OSes on customer VMs. FAs also manage storage nodes.

The collection of Azure hypervisor, root OS/FA, and customer VMs/GAs comprises a compute node. FAs are managed by a fabric controller (FC), which exists outside of compute and storage nodes (compute and storage clusters are managed by separate FCs). If a customer updates their application's configuration file while it's running, the FC communicates with the FA, which then contacts GAs, which notify the application of the configuration change. In the event of a hardware failure, the FC will automatically find available hardware and restart the VM there.



Communication from a Fabric Controller to an agent is unidirectional. The agent implements an SSL-protected service that only responds to requests from the controller. It cannot initiate connections to the controller or other privileged internal nodes. The FC treats all responses as if they were untrusted.



Isolation extends from the Root VM from Guest VMs, and the Guest VMs from one another. Compute nodes are also isolated from storage nodes for increased protection.

The hypervisor and the host OS provide network packet - filters to help assure that untrusted virtual machines cannot generate spoofed traffic or receive traffic not addressed to them, direct traffic to protected infrastructure endpoints, or send/receive inappropriate broadcast traffic.

Additional Rules Configured by Fabric Controller Agent to Isolate VM

By default, all traffic is blocked when a virtual machine is created, and then the fabric controller agent configures the packet filter to add rules and exceptions to allow authorized traffic.

There are two categories of rules that are programmed:

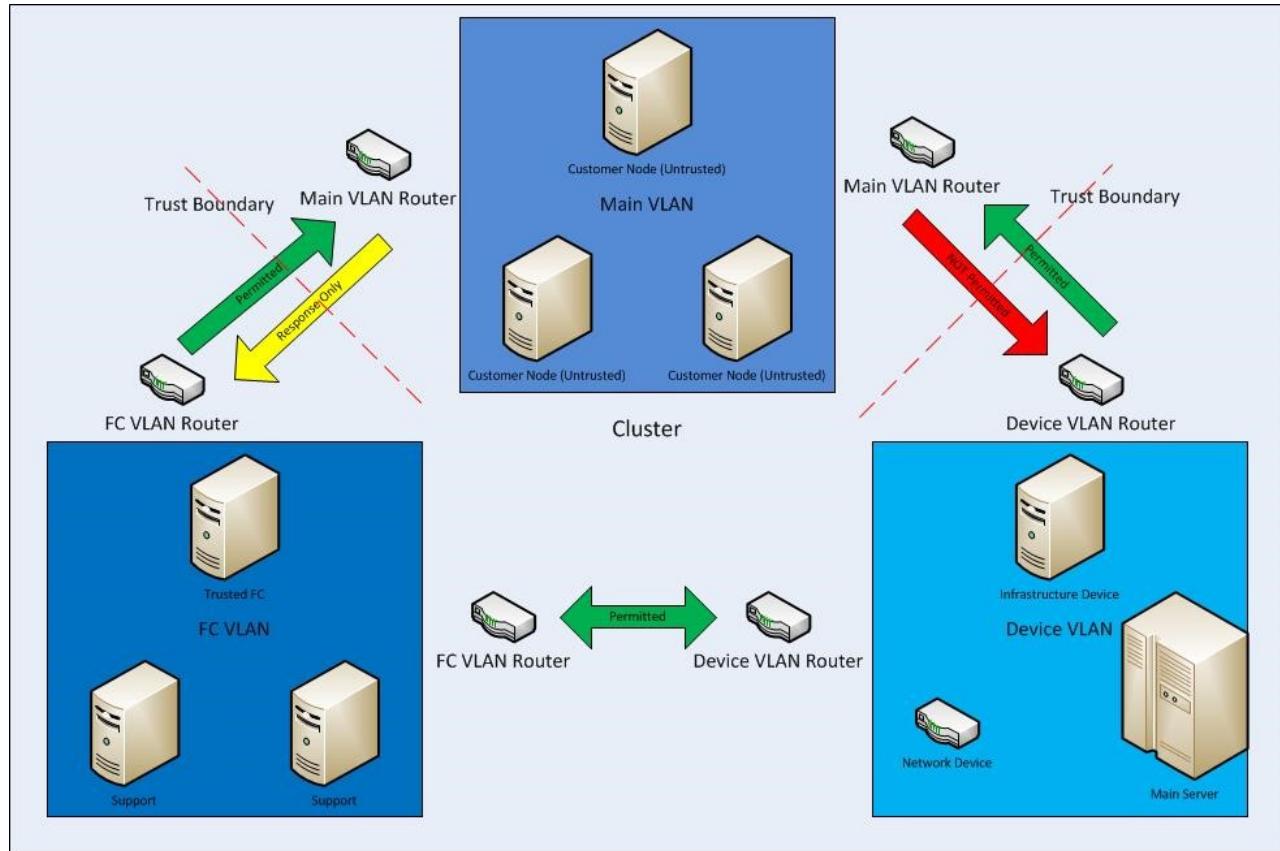
- **Machine configuration or infrastructure rules:** By default, all communication is blocked. There are exceptions to allow a virtual machine to send and receive DHCP and DNS traffic. Virtual machines can also send traffic to the "public" internet and send traffic to other virtual machines within the same Azure Virtual

Network and the OS activation server. The virtual machines' list of allowed outgoing destinations does not include Azure router subnets, Azure management, and other Microsoft properties.

- **Role configuration file:** This defines the inbound Access Control Lists (ACLs) based on the tenant's service model.

VLAN Isolation

There are three VLANs in each cluster:



- The main VLAN – interconnects untrusted customer nodes
- The FC VLAN – contains trusted FCs and supporting systems
- The device VLAN – contains trusted network and other infrastructure devices

Communication is permitted from the FC VLAN to the main VLAN, but cannot be initiated from the main VLAN to the FC VLAN. Communication is also blocked from the main VLAN to the device VLAN. This assures that even if a node running customer code is compromised, it cannot attack nodes on either the FC or device VLANs.

Storage Isolation

Logical Isolation Between Compute and Storage

As part of its fundamental design, Microsoft Azure separates VM-based computation from storage. This separation enables computation and storage to scale independently, making it easier to provide multi-tenancy and isolation.

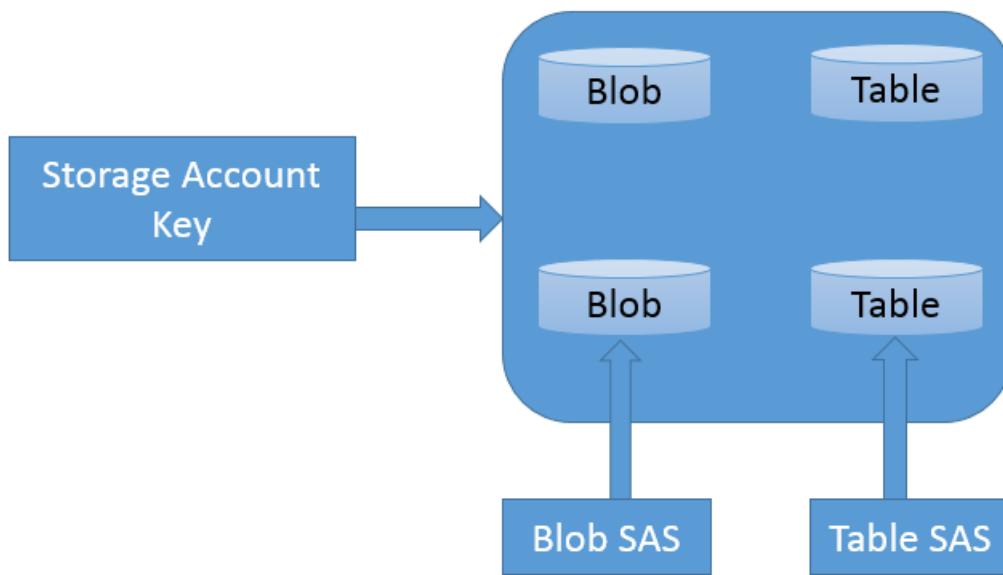
Therefore, Azure Storage runs on separate hardware with no network connectivity to Azure Compute except logically. [This](#) means that when a virtual disk is created, disk space is not allocated for its entire capacity. Instead, a table is created that maps addresses on the virtual disk to areas on the physical disk and that table is initially empty.

The first time a customer writes data on the virtual disk, space on the physical disk is allocated, and a pointer to it is placed in the table.

Isolation Using Storage Access control

Access Control in Azure Storage has a simple access control model. Each Azure subscription can create one or

more Storage Accounts. Each Storage Account has a single secret key that is used to control access to all data in that Storage Account.



Access to Azure Storage data (including Tables) can be controlled through a [SAS \(Shared Access Signature\)](#) token, which grants scoped access. The SAS is created through a query template (URL), signed with the [SAK \(Storage Account Key\)](#). That [signed URL](#) can be given to another process (that is, delegated), which can then fill in the details of the query and make the request of the storage service. A SAS enables you to grant time-based access to clients without revealing the storage account's secret key.

The SAS means that we can grant a client limited permissions, to objects in our storage account for a specified period of time and with a specified set of permissions. We can grant these limited permissions without having to share your account access keys.

IP Level Storage Isolation

You can establish firewalls and define an IP address range for your trusted clients. With an IP address range, only clients that have an IP address within the defined range can connect to [Azure Storage](#).

IP storage data can be protected from unauthorized users via a networking mechanism that is used to allocate a dedicated or dedicated tunnel of traffic to IP storage.

Encryption

Azure offers following types of Encryption to protect data:

- Encryption in transit
- Encryption at rest

Encryption in Transit

Encryption in transit is a mechanism of protecting data when it is transmitted across networks. With Azure Storage, you can secure data using:

- [Transport-level encryption](#), such as HTTPS when you transfer data into or out of Azure Storage.
- [Wire encryption](#), such as SMB 3.0 encryption for Azure File shares.

- [Client-side encryption](#), to encrypt the data before it is transferred into storage and to decrypt the data after it is transferred out of storage.

Encryption at Rest

For many organizations, [data encryption at rest](#) is a mandatory step towards data privacy, compliance, and data sovereignty. There are three Azure features that provide encryption of data that is “at rest”:

- [Storage Service Encryption](#) allows you to request that the storage service automatically encrypt data when writing it to Azure Storage.
- [Client-side Encryption](#) also provides the feature of encryption at rest.
- [Azure Disk Encryption](#) allows you to encrypt the OS disks and data disks used by an IaaS virtual machine.

Azure Disk Encryption

[Azure Disk Encryption](#) for virtual machines (VMs) helps you address organizational security and compliance requirements by encrypting your VM disks (including boot and data disks) with keys and policies you control in [Azure Key Vault](#).

The Disk Encryption solution for Windows is based on [Microsoft BitLocker Drive Encryption](#), and the Linux solution is based on [dm-crypt](#).

The solution supports the following scenarios for IaaS VMs when they are enabled in Microsoft Azure:

- Integration with Azure Key Vault
- Standard tier VMs: A, D, DS, G, GS, and so forth, series IaaS VMs
- Enabling encryption on Windows and Linux IaaS VMs
- Disabling encryption on OS and data drives for Windows IaaS VMs
- Disabling encryption on data drives for Linux IaaS VMs
- Enabling encryption on IaaS VMs that are running Windows client OS
- Enabling encryption on volumes with mount paths
- Enabling encryption on Linux VMs that are configured with disk striping (RAID) by using [mdadm](#)
- Enabling encryption on Linux VMs by using [LVM\(Logical Volume Manager\)](#) for data disks
- Enabling encryption on Windows VMs that are configured by using storage spaces
- All Azure public regions are supported

The solution does not support the following scenarios, features, and technology in the release:

- Basic tier IaaS VMs
- Disabling encryption on an OS drive for Linux IaaS VMs
- IaaS VMs that are created by using the classic VM creation method
- Integration with your on-premises Key Management Service
- Azure Files (shared file system), Network File System (NFS), dynamic volumes, and Windows VMs that are configured with software-based RAID systems

SQL Azure Database Isolation

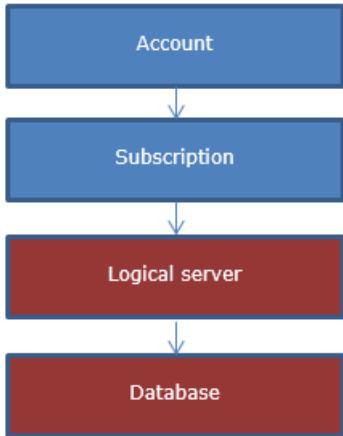
SQL Database is a relational database service in the Microsoft cloud based on the market-leading Microsoft SQL Server engine and capable of handling mission-critical workloads. SQL Database offers predictable data isolation at

account level, geography / region based and based on networking—all with near-zero administration.

SQL Azure Application Model

[Microsoft SQL Azure](#) Database is a cloud-based relational database service built on SQL Server technologies. It provides a highly available, scalable, multi-tenant database service hosted by Microsoft in cloud.

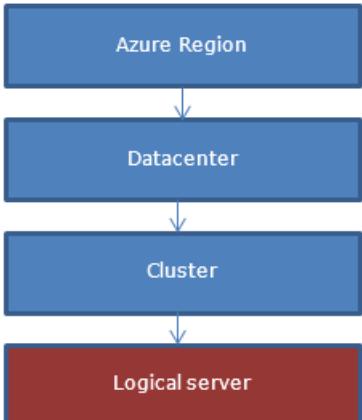
From an application perspective SQL Azure provides the following hierarchy: Each level has one-to-many containment of levels below.



The account and subscription are Microsoft Azure platform concepts to associate billing and management.

Logical servers and databases are SQL Azure-specific concepts and are managed by using SQL Azure, provided OData and TSQL interfaces or via SQL Azure portal that integrated into Azure portal.

SQL Azure servers are not physical or VM instances, instead they are collections of databases, sharing management and security policies, which are stored in so called “logical master” database.



Logical master databases include:

- SQL logins used to connect to the server
- Firewall rules

Billing and usage-related information for SQL Azure databases from the same logical server are not guaranteed to be on the same physical instance in SQL Azure cluster, instead applications must provide the target database name when connecting.

From a customer perspective, a logical server is created in a geo-graphical region while the actual creation of the server happens in one of the clusters in the region.

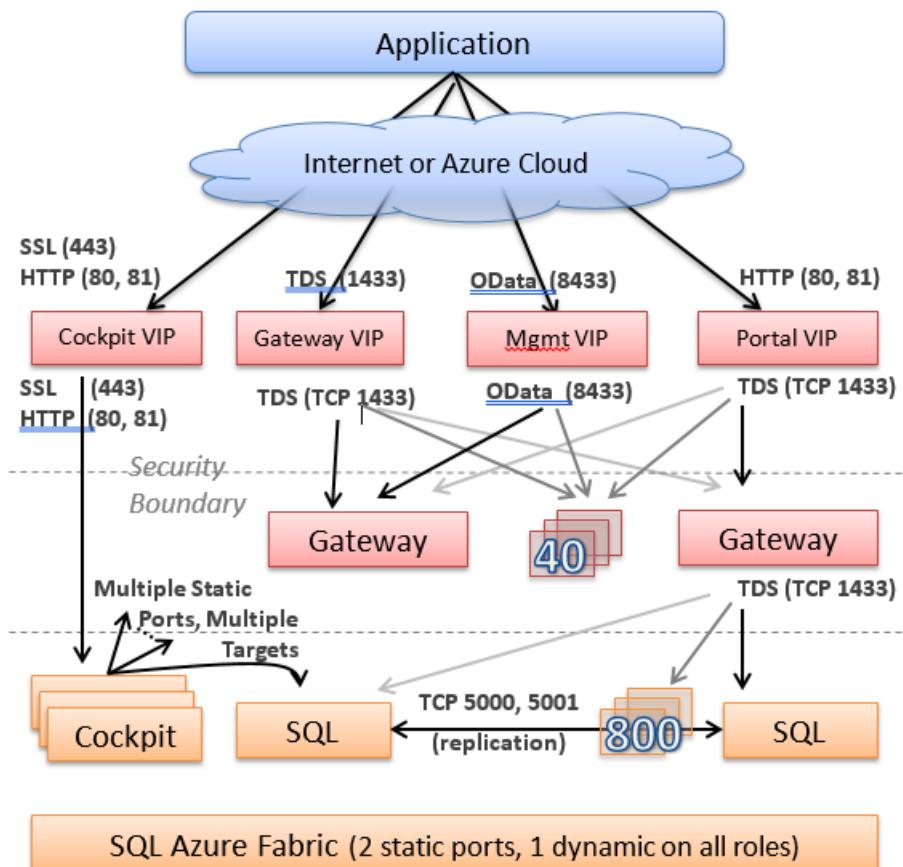
Isolation through Network Topology

When a logical server is created and its DNS name is registered, the DNS name points to the so called “Gateway

"VIP" address in the specific data center where the server was placed.

Behind the VIP (virtual IP address), we have a collection of stateless gateway services. In general, gateways get involved when there is coordination needed between multiple data sources (master database, user database, etc.). Gateway services implement the following:

- **TDS connection proxying.** This includes locating user database in the backend cluster, implementing the login sequence and then forwarding the TDS packets to the backend and back.
- **Database management.** This includes implementing a collection of workflows to do CREATE/ALTER/DROP database operations. The database operations can be invoked by either sniffing TDS packets or explicit OData APIs.
- CREATE/ALTER/DROP login/user operations
- Logical server management operations via OData API



The tier behind the gateways is called "back-end". This is where all the data is stored in a highly available fashion. Each piece of data is said to belong to a "partition" or "failover unit", each of them having at least three replicas. Replicas are stored and replicated by SQL Server engine and managed by a failover system often referred to as "fabric".

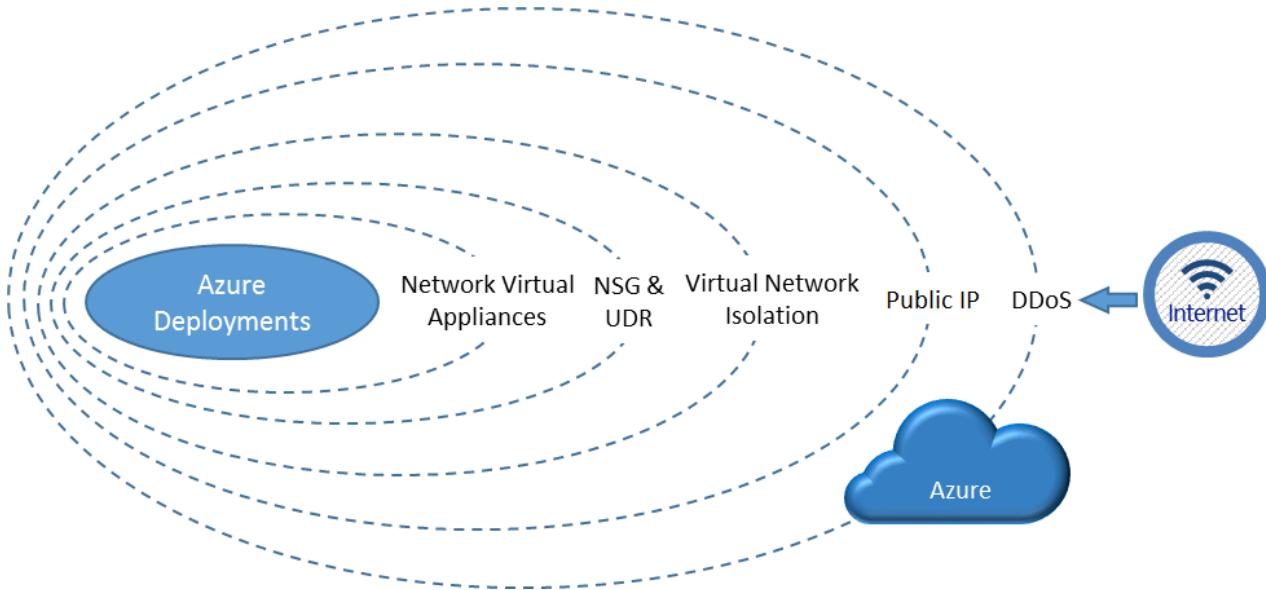
Generally, the back-end system does not communicate outbound to other systems as a security precaution. This is reserved to the systems in the front-end (gateway) tier. The gateway tier machines have limited privileges on the back-end machines to minimize the attack surface as a defense-in-depth mechanism.

Isolation by Machine Function and Access

SQL Azure is composed of services running on different machine functions. SQL Azure is divided into "backend" Cloud Database and "front-end" (Gateway/Management) environments, with the general principle of traffic only going into back-end and not out. The front-end environment can communicate to the outside world of other services and in general, has only limited permissions in the back-end (enough to call the entry points it needs to invoke).

Networking Isolation

Azure deployment has multiple layers of network isolation. The following diagram shows various layers of network isolation Azure provides to customers. These layers are both native in the Azure platform itself and customer-defined features. Inbound from the Internet, Azure DDoS provides isolation against large-scale attacks against Azure. The next layer of isolation is customer-defined public IP addresses (endpoints), which are used to determine which traffic can pass through the cloud service to the virtual network. Native Azure virtual network isolation ensures complete isolation from all other networks, and that traffic only flows through user configured paths and methods. These paths and methods are the next layer, where NSGs, UDR, and network virtual appliances can be used to create isolation boundaries to protect the application deployments in the protected network.



Traffic isolation: A [virtual network](#) is the traffic isolation boundary on the Azure platform. Virtual machines (VMs) in one virtual network cannot communicate directly to VMs in a different virtual network, even if both virtual networks are created by the same customer. Isolation is a critical property that ensures customer VMs and communication remains private within a virtual network.

[Subnet](#) offers an additional layer of isolation with in virtual network based on IP range. IP addresses in the virtual network, you can divide a virtual network into multiple subnets for organization and security. VMs and PaaS role instances deployed to subnets (same or different) within a VNet can communicate with each other without any extra configuration. You can also configure [network security group \(NSGs\)](#) to allow or deny network traffic to a VM instance based on rules configured in access control list (ACL) of NSG. NSGs can be associated with either subnets or individual VM instances within that subnet. When an NSG is associated with a subnet, the ACL rules apply to all the VM instances in that subnet.

Next Steps

- [Network Isolation Options for Machines in Windows Azure Virtual Networks](#)

This includes the classic front-end and back-end scenario where machines in a particular back-end network or subnetwork may only allow certain clients or other computers to connect to a particular endpoint based on a whitelist of IP addresses.

- [Compute Isolation](#)

Microsoft Azure provides a various cloud-based computing services that include a wide selection of compute instances & services that can scale up and down automatically to meet the needs of your application or enterprise.

- [Storage Isolation](#)

Microsoft Azure separates customer VM-based computation from storage. This separation enables computation

and storage to scale independently, making it easier to provide multi-tenancy and isolation. Therefore, Azure Storage runs on separate hardware with no network connectivity to Azure Compute except logically. All requests run over HTTP or HTTPS based on customer's choice.

Azure security technical capabilities

9/7/2017 • 31 min to read • [Edit Online](#)

To assist current and prospective Azure customers understand and utilize the various Security-related capabilities available in and surrounding the Azure Platform, Microsoft has developed a series of White Papers, Security Overviews, Best Practices, and Checklists. The topics range in terms of breadth and depth and are updated periodically. This document is part of that series as summarized in the Abstract section below. Further information on this Azure Security series can be found at (URL).

Azure platform

[Microsoft Azure](#) is a cloud platform comprised of infrastructure and application services, with integrated data services and advanced analytics, and developer tools and services, hosted within Microsoft's public cloud data centers. Customers use Azure for many different capacities and scenarios, from basic compute, networking, and storage, to mobile and web app services, to full cloud scenarios like Internet of Things, and can be used with open source technologies, and deployed as hybrid cloud or hosted within a customer's datacenter. Azure provides cloud technology as building blocks to help companies save costs, innovate quickly, and manage systems proactively. When you build on, or migrate IT assets to a cloud provider, you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Microsoft Azure is the only cloud computing provider that offers a secure, consistent application platform and infrastructure-as-a-service for teams to work within their different cloud skillsets and levels of project complexity, with integrated data services and analytics that uncover intelligence from data wherever it exists, across both Microsoft and non-Microsoft platforms, open frameworks and tools, providing choice for integrating cloud with on-premises as well deploying Azure cloud services within on-premises datacenters. As part of the Microsoft Trusted Cloud, customers rely on Azure for industry-leading security, reliability, compliance, privacy, and the vast network of people, partners, and processes to support organizations in the cloud.

With Microsoft Azure, you can:

- Accelerate innovation with the cloud.
- Power business decisions & apps with insights.
- Build freely and deploy anywhere.
- Protect their business.

Scope

The focal point of this whitepaper concerns security features and functionality supporting Microsoft Azure's core components, namely [Microsoft Azure Storage](#), [Microsoft Azure SQL Databases](#), [Microsoft Azure's virtual machine model](#), and the tools and infrastructure that manage it all. This white paper focus on Microsoft Azure technical capabilities available to you as customers to fulfil their role in protecting the security and privacy of their data.

The importance of understanding this shared responsibility model is essential for customers who are moving to the cloud. Cloud providers offer considerable advantages for security and compliance efforts, but these advantages do not absolve the customer from protecting their users, applications, and service offerings.

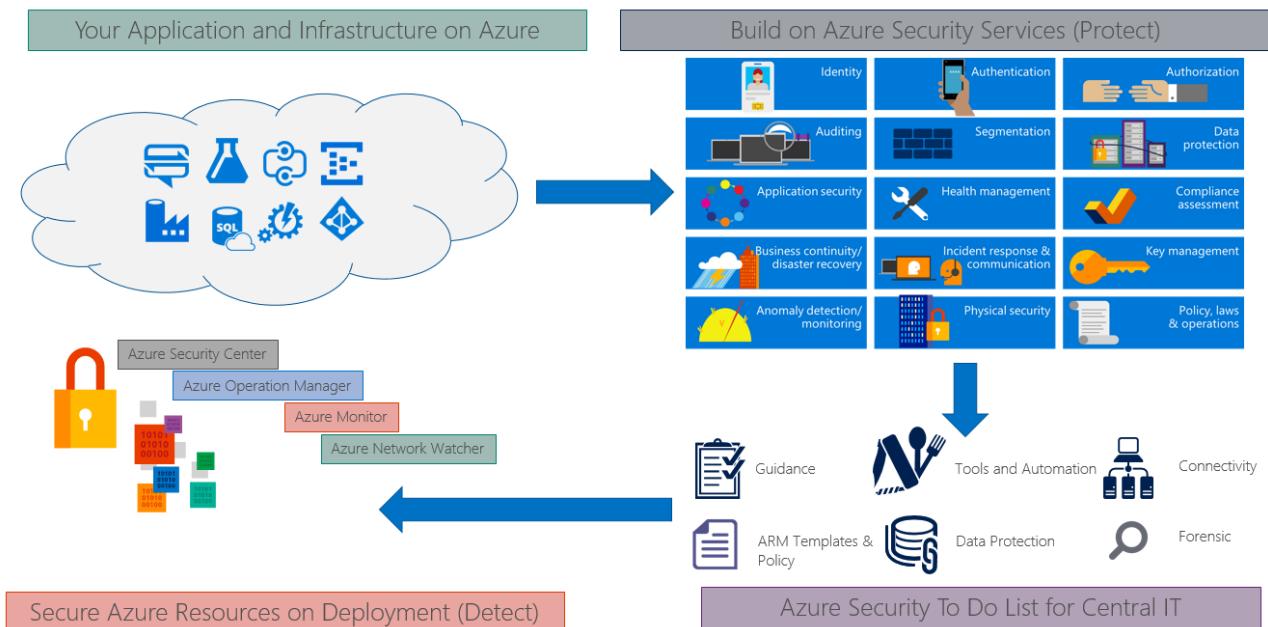
For IaaS solutions, the customer is responsible or has a shared responsibility for securing and managing the operating system, network configuration, applications, identity, clients, and data. PaaS solutions build on IaaS

deployments, the customer is still responsible or has a shared responsibility for securing and managing applications, identity, clients, and data. For SaaS solutions, Nonetheless, the customer continues to be accountable. They must ensure that data is classified correctly, and they share a responsibility to manage their users and end-point devices.

This document does not provide detailed coverage of any of the related Microsoft Azure platform components such as Azure Web Sites, Azure Active Directory, HDInsight, Media Services, and other services that are layered atop the core components. Although a minimum level of general information is provided, readers are assumed familiar with Azure basic concepts as described in other references provided by Microsoft and included in links provided in this white paper.

Available security technical capabilities to fulfil user (Customer) responsibility - Big picture

Microsoft Azure provides services that can help customers meet the security, privacy, and compliance needs. The Following picture helps explain various Azure services available for users to build a secure and compliant application infrastructure based on industry standards.



Manage and control identity and user access (Protect)

Azure helps you protect business and personal information by enabling you to manage user identities and credentials and control access.

Azure active directory

Microsoft identity and access management solutions help IT protect access to applications and resources across the corporate datacenter and into the cloud, enabling additional levels of validation such as multi-factor authentication and conditional access policies. Monitoring suspicious activity through advanced security reporting, auditing and alerting helps mitigate potential security issues. [Azure Active Directory Premium](#) provides single sign-on to thousands of cloud (SaaS) apps and access to web apps you run on-premises.

Security benefits of Azure Active Directory (AD) include the ability to:

- Create and manage a single identity for each user across your hybrid enterprise, keeping users, groups, and devices in sync.
- Provide single sign-on access to your applications including thousands of pre-integrated SaaS apps.
- Enable application access security by enforcing rules-based Multi-Factor Authentication for both on-

premises and cloud applications.

- Provision secure remote access to on-premises web applications through Azure AD Application Proxy.

Azure active directory portal is available a part of azure portal. From this dashboard, you can get an overview of the state of your organization, and easily dive into managing the directory, users, or application access.

The screenshot shows the Azure Active Directory admin center dashboard. At the top, there's a header with a back arrow, forward arrow, refresh button, and a URL bar showing https://aad.portal.azure.com/. The title is "Azure Active Directory admin center". On the right, there's a user profile for "IlanaS@WoodGroveO... WOODGROVE". Below the header, there's a navigation bar with icons for Home, Groups, Users, Applications, and Reports. The main content area has several sections: 1) A "Welcome to the Azure AD admin center" section with a blue diamond icon and text: "Azure AD helps you protect your business and empower your users." It also includes a link to "Learn more about Azure AD". 2) A "Quick tasks" sidebar with links: "Add a user", "Add a guest user", "Add a group", "Find a user", "Find a group", and "Find an enterprise app". 3) A "Recommended" section with a purple icon and text: "Sync with Windows Server AD" and "Sync users and groups from your on-premises directory to your Azure AD". 4) A "Self-service password reset" section with a green arrow icon and text: "Enable your users to reset their forgotten passwords". 5) A "Company branding" section with a yellow and green bar icon and text: "Customize the text and graphics your users see when they sign in to your Azure AD". 6) A "Users Sign-ins" chart showing sign-in activity from April 16 to May 7. 7) A "Azure AD Connect" section with a sync icon and "Sync enabled" status. 8) An "Audit Logs" section with a blue square icon and a "View activity" button. The overall theme is branded with "WOODGROVE" and "WOODGROVEONLINE.COM".

Following are core Azure Identity management capabilities:

- Single sign-on
- Multi-factor authentication
- Security monitoring, alerts, and machine learning-based reports
- Consumer identity and access management
- Device registration
- Privileged identity management
- Identity protection

Single sign-on

[Single sign-on \(SSO\)](#) means being able to access all the applications and resources that you need to do business, by signing in only once using a single user account. Once signed in, you can access all the applications you need without being required to authenticate (for example, type a password) a second time.

Many organizations rely upon software as a service (SaaS) applications such as Office 365, Box and Salesforce for end-user productivity. Historically, IT staff needed to individually create and update user accounts in each SaaS application, and users had to remember a password for each SaaS application.

[Azure AD extends on-premises Active Directory into the cloud](#), enabling users to use their primary organizational account to not only sign in to their domain-joined devices and company resources, but also all the web and SaaS applications needed for their job.

Not only do users not have to manage multiple sets of usernames and passwords, application access can be automatically provisioned or de-provisioned based on organizational groups and their status as an employee.

Azure AD introduces security and access governance controls that enable you to centrally manage users' access across SaaS applications.

Multi-factor authentication

Azure Multi-factor authentication (MFA) is a method of authentication that requires the use of more than one verification method and adds a critical second layer of security to user sign-ins and transactions. MFA helps safeguard access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication via a range of verification options—phone call, text message, or mobile app notification or verification code and third-party OAuth tokens.

Security monitoring, alerts, and machine learning-based reports

Security monitoring and alerts and machine learning-based reports that identify inconsistent access patterns can help you protect your business. You can use Azure Active Directory's access and usage reports to gain visibility into the integrity and security of your organization's directory. With this information, a directory admin can better determine where possible security risks may lie so that they can adequately plan to mitigate those risks.

In the Azure classic portal or through [Azure Active directory portal, reports](#) are categorized in the following ways:

- Anomaly reports – contain sign in events that we found to be anomalous. Our goal is to make you aware of such activity and enable you to be able to decide about whether an event is suspicious.
- Integrated Application reports – provide insights into how cloud applications are being used in your organization. Azure Active Directory offers integration with thousands of cloud applications.
- Error reports – indicate errors that may occur when provisioning accounts to external applications.
- User-specific reports – display device/sign in activity data for a specific user.
- Activity logs – contain a record of all audited events within the last 24 hours, last 7 days, or last 30 days, and group activity changes, and password reset and registration activity.

Consumer identity and access management

[Azure Active Directory B2C](#) is a highly available, global, identity management service for consumer-facing applications that scales to hundreds of millions of identities. It can be integrated across mobile and web platforms. Your consumers can log on to all your applications through customizable experiences by using their existing social accounts or by creating new credentials.

In the past, application developers who wanted to [sign up and sign in consumers](#) into their applications would have written their own code. And they would have used on-premises databases or systems to store usernames and passwords. Azure Active Directory B2C offers your organization a better way to integrate consumer identity management into applications with the help of a secure, standards-based platform, and a large set of extensible policies.

When you use Azure Active Directory B2C, your consumers can sign up for your applications by using their existing social accounts (Facebook, Google, Amazon, LinkedIn) or by creating new credentials (email address and password, or username and password).

Device registration

[Azure AD Device Registration](#) is the foundation for device-based [conditional access](#) scenarios. When a device is registered, Azure Active Directory Device Registration provides the device with an identity that is used to authenticate the device when the user signs in. The authenticated device, and the attributes of the device, can then be used to enforce conditional access policies for applications that are hosted in the cloud and on-premises.

When combined with a [mobile device management \(MDM\)](#) solution such as Intune, the device attributes in Azure Active Directory are updated with additional information about the device. This allows you to create conditional access rules that enforce access from devices to meet your standards for security and compliance.

Privileged identity management

Azure Active Directory (AD) Privileged Identity Management lets you manage, control, and monitor your privileged identities and access to resources in Azure AD as well as other Microsoft online services like Office 365 or Microsoft Intune.

Sometimes users need to carry out privileged operations in Azure or Office 365 resources, or other SaaS apps. This often means organizations have to give them permanent privileged access in Azure AD. This is a growing security risk for cloud-hosted resources because organizations can't sufficiently monitor what those users are doing with their admin privileges. Additionally, if a user account with privileged access is compromised, that one breach could impact their overall cloud security. Azure AD Privileged Identity Management helps to resolve this risk.

Azure AD Privileged Identity Management lets you:

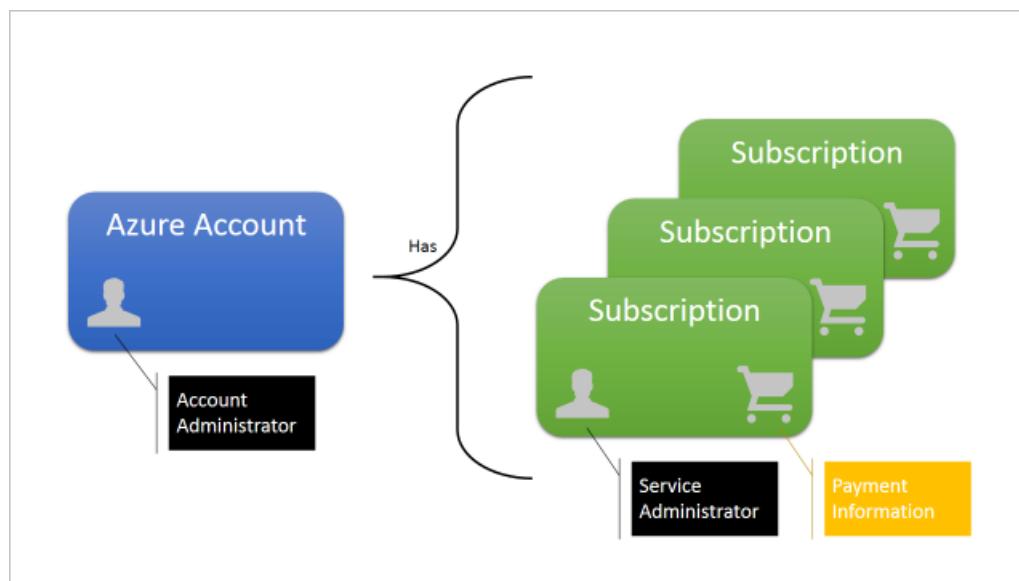
- See which users are Azure AD admins
- Enable on-demand, "just in time" administrative access to Microsoft Online Services like Office 365 and Intune
- Get reports about administrator access history and changes in administrator assignments
- Get alerts about access to a privileged role

Identity protection

Azure AD Identity Protection is a security service that provides a consolidated view into risk events and potential vulnerabilities affecting your organization's identities. Identity Protection uses existing Azure Active Directory's anomaly detection capabilities (available through Azure AD's Anomalous Activity Reports), and introduces new risk event types that can detect anomalies in real-time.

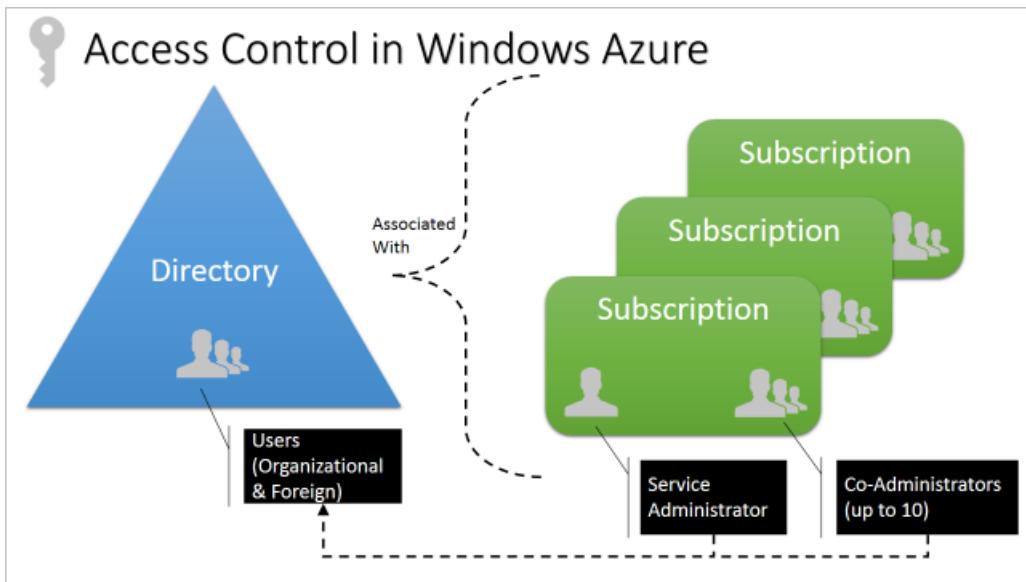
Secured resource access in Azure

Access control in Azure starts from a billing perspective. The owner of an Azure account, accessed by visiting the [Azure Accounts Center](#), is the Account Administrator (AA). Subscriptions are a container for billing, but they also act as a security boundary: each subscription has a Service Administrator (SA) who can add, remove, and modify Azure resources in that subscription by using the [Azure classic portal](#). The default SA of a new subscription is the AA, but the AA can change the SA in the Azure Accounts Center.

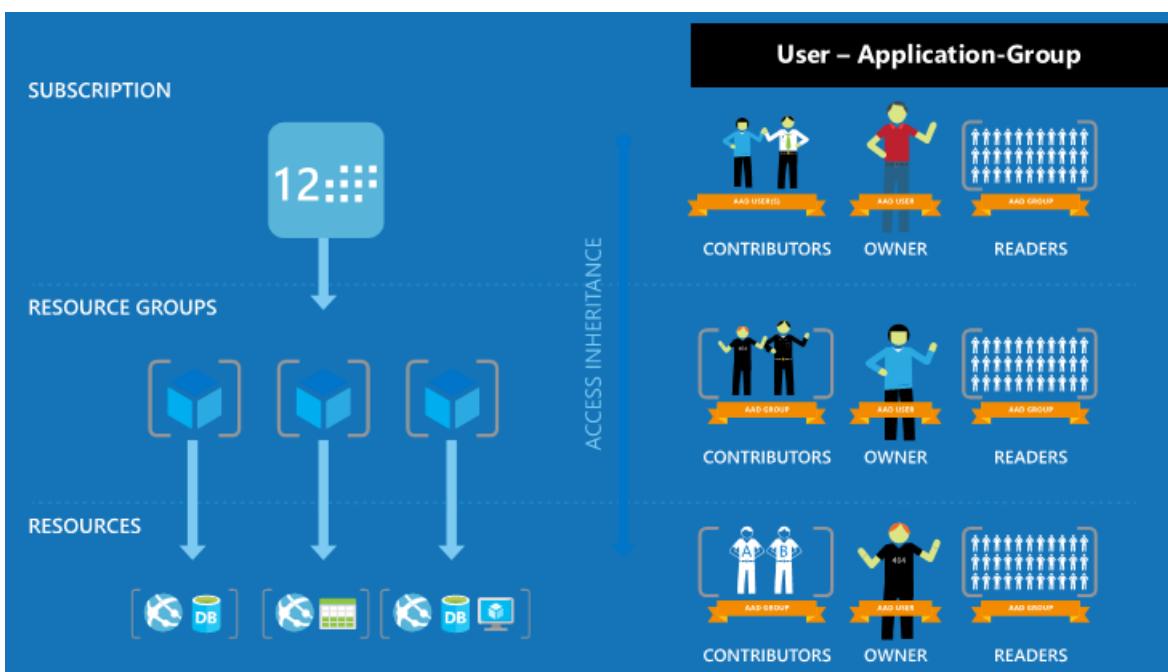


Subscriptions also have an association with a directory. The directory defines a set of users. These can be users from the work or school that created the directory, or they can be external users (that is, Microsoft Accounts). Subscriptions are accessible by a subset of those directory users who have been assigned as either Service Administrator (SA) or Co-Administrator (CA); the only exception is that, for legacy reasons, Microsoft Accounts (formerly Windows Live ID) can be assigned as SA or CA without being present in the directory.

Security-oriented companies should focus on giving employees the exact permissions they need. Too many permissions can expose an account to attackers. Too few permissions mean that employees can't get their work done efficiently. [Azure Role-Based Access Control \(RBAC\)](#) helps address this problem by offering fine-grained access management for Azure.



Using RBAC, you can segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Instead of giving everybody unrestricted permissions in your Azure subscription or resources, you can allow only certain actions. For example, use RBAC to let one employee manage virtual machines in a subscription, while another can manage SQL databases within the same subscription.



Azure data security and encryption (protect)

One of the keys to data protection in the cloud is accounting for the possible states in which your data may occur, and what controls are available for that state. For Azure data security and encryption best practices the recommendations be around the following data's states.

- At-rest: This includes all information storage objects, containers, and types that exist statically on physical media, be it magnetic or optical disk.
- In-Transit: When data is being transferred between components, locations or programs, such as over the network, across a service bus (from on-premises to cloud and vice-versa, including hybrid connections such

as ExpressRoute), or during an input/output process, it is thought of as being in-motion.

Encryption @ rest

To achieve Encryption at Rest each of the following:

Support at least one of the recommended encryption models detailed in the following table to encrypt data.

ENCRYPTION MODELS			
Server Encryption	Server Encryption	Server Encryption	Client Encryption
Server-Side Encryption using Service Managed Keys	Server-side encryption using Customer-Managed Keys in Azure Key Vault	Server-side encryption using on-prem customer managed keys	
<ul style="list-style-type: none">• Azure Resource Providers perform the encryption and decryption operations• Microsoft manages the keys• Full cloud functionality	<ul style="list-style-type: none">• Azure Resource Providers perform the encryption and decryption operations• Customer controls keys via Azure Key Vault• Full cloud functionality	<ul style="list-style-type: none">• Azure Resource Providers perform the encryption and decryption operations• Customer controls keys On-Prem• Full cloud functionality	<ul style="list-style-type: none">• Azure services cannot see decrypted data• Customers keep keys on-premises (or in other secure stores). Keys are not available to Azure services• Reduced cloud functionality

Enabling encryption at rest

Identify All Locations Your Stores Data

The goal of Encryption at Rest is to encrypt all data. Doing so eliminates the possibility of missing important data or all persisted locations. Enumerate all data stored by your application.

NOTE

Not just "application data" or "PII" but any data relating to application including account metadata (subscription mappings, contract info, PII).

Consider what stores you are using to store data. For example:

- External storage (for example, SQL Azure, Document DB, HDInsights, Data Lake, etc.)
- Temporary storage (any local cache that includes tenant data)
- In-memory cache (could be put into the page file.)

Leverage the existing encryption at rest support in Azure

For each store you use, leverage the existing Encryption at Rest support.

- Azure Storage: See [Azure Storage Service Encryption for Data at Rest](#),
- SQL Azure: See [Transparent Data Encryption \(TDE\)](#), [SQL Always Encrypted](#)
- VM & Local disk storage ([Azure Disk Encryption](#))

For VM and Local disk storage use Azure Disk Encryption where supported:

IaaS

Services with IaaS VMs (Windows or Linux) should use [Azure Disk Encryption](#) to encrypt volumes containing customer data.

PaaS v2

Services running on PaaS v2 using Service Fabric can use Azure disk encryption for Virtual Machine Scale Set [VMSS] to encrypt their PaaS v2 VMs.

PaaS v1

Azure Disk Encryption currently is not supported on PaaS v1. Therefore, you must use application level encryption to encrypt persisted data at rest. This includes, but is not limited to, application data, temporary files, logs, and crash dumps.

Most services should attempt to leverage the encryption of a storage resource provider. Some services have to do explicit encryption, for example, any persisted key material (Certificates, root / master keys) must be stored in Key Vault.

If you support service-side encryption with customer-managed keys there needs to be a way for the customer to get the key to us. The supported and recommended way to do that by integrating with Azure Key Vault (AKV). In this case customers can add and manage their keys in Azure Key Vault. A customer can learn how to use AKV via [Getting Started with Key Vault](#).

To integrate with Azure Key Vault, you'd add code to request a key from AKV when needed for decryption.

- See [Azure Key Vault – Step by Step](#) for info on how to integrate with AKV.

If you support customer managed keys, you need to provide a UX for the customer to specify which Key Vault (or Key Vault URI) to use.

As Encryption at Rest involves the encryption of host, infrastructure and tenant data, the loss of the keys due to system failure or malicious activity could mean all the encrypted data is lost. It is therefore critical that your Encryption at Rest solution has a comprehensive disaster recovery story resilient to system failures and malicious activity.

Services that implement Encryption at Rest are usually still susceptible to the encryption keys or data being left unencrypted on the host drive (for example, in the page file of the host OS.) Therefore, services must ensure the host volume for their services is encrypted. To facilitate this Compute team has enabled the deployment of Host Encryption, which uses [BitLocker](#) NKP and extensions to the DCM service and agent to encrypt the host volume.

Most services are implemented on standard Azure VMs. Such services should get [Host Encryption](#) automatically when Compute enables it. For services running in Compute managed clusters host encryption is enabled automatically as Windows Server 2016 is rolled out.

Encryption in-transit

Protecting data in transit should be essential part of your data protection strategy. Since data is moving back and forth from many locations, the general recommendation is that you always use SSL/TLS protocols to exchange data across different locations. In some circumstances, you may want to isolate the entire communication channel between your on-premises and cloud infrastructure by using a virtual private network (VPN).

For data moving between your on-premises infrastructure and Azure, you should consider appropriate safeguards such as HTTPS or VPN.

For organizations that need to secure access from multiple workstations located on-premises to Azure, use [Azure site-to-site VPN](#).

For organizations that need to secure access from one workstation located on-premises to Azure, use [Point-to-Site VPN](#).

Larger data sets can be moved over a dedicated high-speed WAN link such as [ExpressRoute](#). If you choose to use ExpressRoute, you can also encrypt the data at the application-level using [SSL/TLS](#) or other protocols for added protection.

If you are interacting with Azure Storage through the Azure Portal, all transactions occur via HTTPS. [Storage REST](#)

[API](#) over HTTPS can also be used to interact with [Azure Storage](#) and [Azure SQL Database](#).

Organizations that fail to protect data in transit are more susceptible for [man-in-the-middle attacks](#), [eavesdropping](#), and session hijacking. These attacks can be the first step in gaining access to confidential data.

You can learn more about Azure VPN option by reading the article [Planning and design for VPN Gateway](#).

Enforce file level data encryption

[Azure RMS](#) uses encryption, identity, and authorization policies to help secure your files and email. Azure RMS works across multiple devices — phones, tablets, and PCs by protecting both within your organization and outside your organization. This capability is possible because Azure RMS adds a level of protection that remains with the data, even when it leaves your organization's boundaries.

When you use Azure RMS to protect your files, you are using industry-standard cryptography with full support of [FIPS 140-2](#). When you leverage Azure RMS for data protection, you have the assurance that the protection stays with the file, even if it is copied to storage that is not under the control of IT, such as a cloud storage service. The same occurs for files shared via e-mail, the file is protected as an attachment to an email message, with instructions how to open the protected attachment. When planning for Azure RMS adoption we recommend the following:

- Install the [RMS sharing app](#). This app integrates with Office applications by installing an Office add-in so that users can easily protect files directly.
- Configure applications and services to support Azure RMS
- Create [custom templates](#) that reflect your business requirements. For example: a template for top secret data that should be applied in all top secret related emails.

Organizations that are weak on [data classification](#) and file protection may be more susceptible to data leakage. Without proper file protection, organizations won't be able to obtain business insights, monitor for abuse and prevent malicious access to files.

NOTE

You can learn more about Azure RMS by reading the article [Getting Started with Azure Rights Management](#).

Secure your application (protect)

While Azure is responsible for securing the infrastructure and platform that your application runs on, it is your responsibility to secure your application itself. In other words, you need to develop, deploy, and manage your application code and content in a secure way. Without this, your application code or content can still be vulnerable to threats.

Web application firewall (WAF)

[Web application firewall \(WAF\)](#) is a feature of [Application Gateway](#) that provides centralized protection of your web applications from common exploits and vulnerabilities.

Web application firewall is based on rules from the [OWASP core rule sets](#) 3.0 or 2.2.9. Web applications are increasingly targets of malicious attacks that exploit common known vulnerabilities. Common among these exploits are SQL injection attacks, cross site scripting attacks to name a few. Preventing such attacks in application code can be challenging and may require rigorous maintenance, patching and monitoring at multiple layers of the application topology. A centralized web application firewall helps make security management much simpler and gives better assurance to application administrators against threats or intrusions. A WAF solution can also react to a security threat faster by patching a known vulnerability at a central location versus securing each of individual web applications. Existing application gateways can be converted to a web application firewall enabled application gateway easily.

Some of the common web vulnerabilities which web application firewall protects against includes:

- SQL injection protection
- Cross site scripting protection
- Common Web Attacks Protection such as command injection, HTTP request smuggling, HTTP response splitting, and remote file inclusion attack
- Protection against HTTP protocol violations
- Protection against HTTP protocol anomalies such as missing host user-agent and accept headers
- Prevention against bots, crawlers, and scanners
- Detection of common application misconfigurations (that is, Apache, IIS, etc.)

NOTE

For a more detailed list of rules and their protections see the following [Core rule sets](#):

Azure also provides several easy-to-use features to help secure both inbound and outbound traffic for your app. Azure also helps customers secure their application code by providing externally provided functionality to scan your web application for vulnerabilities.

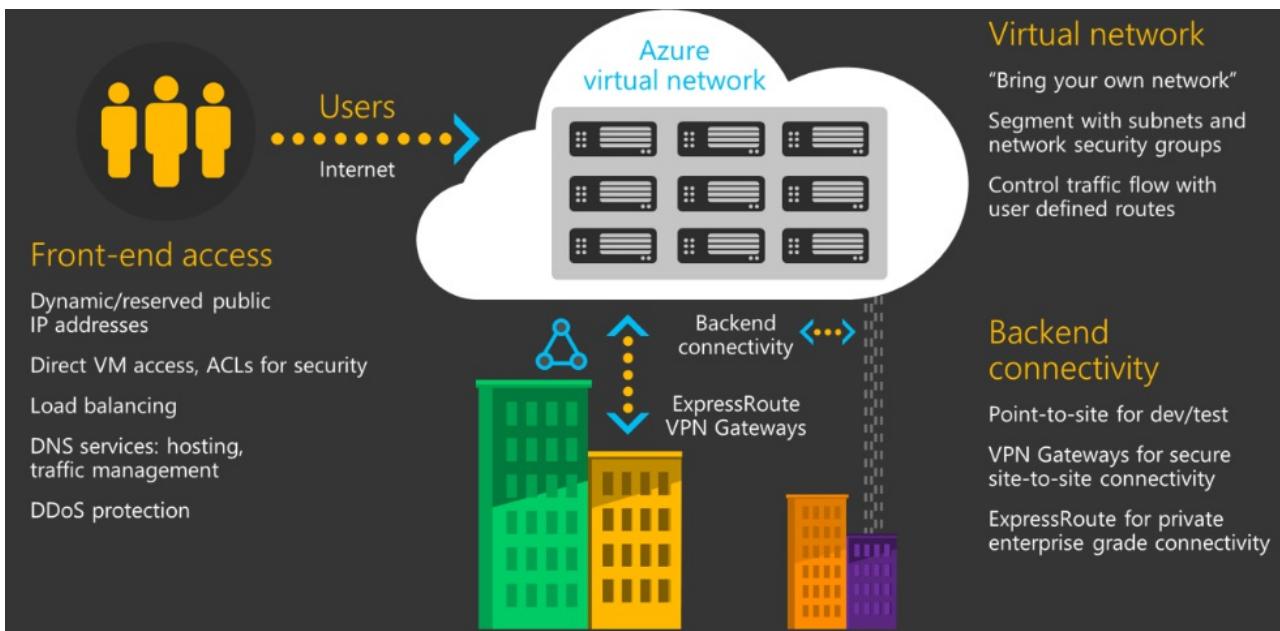
- [Secure your web app using various means of authentication and authorization](#)
 - [Setup Azure Active Directory authentication for your app](#)
- [Secure traffic to your app by enabling Transport Layer Security \(TLS/SSL\) - HTTPS](#)
 - [Force all incoming traffic over HTTPS connection](#)
 - [Enable Strict Transport Security \(HSTS\)](#)
- [Restrict access to your app by client's IP address](#)
- [Restrict access to your app by client's behavior - request frequency and concurrency](#)
- [Scan your web app code for vulnerabilities using Tinfoil Security Scanning](#)
- [Configure TLS mutual authentication to require client certificates to connect to your web app](#)
- [Configure a client certificate for use from your app to securely connect to external resources](#)
- [Remove standard server headers to avoid tools from fingerprinting your app](#)
- [Securely connect your app with resources in a private network using Point-To-Site VPN](#)
- [Securely connect your app with resources in a private network using Hybrid Connections](#)

Azure App Service uses the same Antimalware solution used by Azure Cloud Services and Virtual Machines. To learn more about this refer to our [Antimalware documentation](#).

Secure your network (protect)

Microsoft Azure includes a robust networking infrastructure to support your application and service connectivity requirements. Network connectivity is possible between resources located in Azure, between on-premises and Azure hosted resources, and to and from the Internet and Azure.

The [Azure network infrastructure](#) enables you to securely connect Azure resources to each other with [virtual networks \(VNets\)](#). A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure cloud network dedicated to your subscription. You can connect VNets to your on-premises networks.



If you need basic network level access control (based on IP address and the TCP or UDP protocols), then you can use [Network Security Groups](#). A Network Security Group (NSG) is a basic stateful packet filtering firewall and it enables you to control access based on a [5-tuple](#).

Azure networking supports the ability to customize the routing behavior for network traffic on your Azure Virtual Networks. You can do this by configuring [User-Defined Routes](#) in Azure.

[Forced tunneling](#) is a mechanism you can use to ensure that your services are not allowed to initiate a connection to devices on the Internet.

Azure supports dedicated WAN link connectivity to your on-premises network and an Azure Virtual Network with [ExpressRoute](#). The link between Azure and your site uses a dedicated connection that does not go over the public Internet. If your Azure application is running in multiple datacenters, you can use [Azure Traffic Manager](#) to route requests from users intelligently across instances of the application. You can also route traffic to services not running in Azure if they are accessible from the Internet.

Virtual machine security (protect)

[Azure Virtual Machines](#) lets you deploy a wide range of computing solutions in an agile way. With support for Microsoft Windows, Linux, Microsoft SQL Server, Oracle, IBM, SAP, and Azure BizTalk Services, you can deploy any workload and any language on nearly any operating system.

With Azure, you can use [antimalware software](#) from security vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky to protect your virtual machines from malicious files, adware, and other threats.

Microsoft Antimalware for Azure Cloud Services and Virtual Machines is a real-time protection capability that helps identify and remove viruses, spyware, and other malicious software. Microsoft Antimalware provides configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems.

[Azure Backup](#) is a scalable solution that protects your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications. With Azure Backup, your virtual machines running Windows and Linux are protected.

[Azure Site Recovery](#) helps orchestrate replication, failover, and recovery of workloads and apps so that they are available from a secondary location if your primary location goes down.

Ensure compliance: Cloud services due diligence checklist (protect)

Microsoft developed [the Cloud Services Due Diligence Checklist](#) to help organizations exercise due diligence as

they consider a move to the cloud. It provides a structure for an organization of any size and type—private businesses and public-sector organizations, including government at all levels and nonprofits—to identify their own performance, service, data management, and governance objectives and requirements. This allows them to compare the offerings of different cloud service providers, ultimately forming the basis for a cloud service agreement.

The checklist provides a framework that aligns clause-by-clause with a new international standard for cloud service agreements, ISO/IEC 19086. This standard offers a unified set of considerations for organizations to help them make decisions about cloud adoption, and create a common ground for comparing cloud service offerings.

The checklist promotes a thoroughly vetted move to the cloud, providing structured guidance and a consistent, repeatable approach for choosing a cloud service provider.

Cloud adoption is no longer simply a technology decision. Because checklist requirements touch on every aspect of an organization, they serve to convene all key internal decision-makers—the CIO and CISO as well as legal, risk management, procurement, and compliance professionals. This increases the efficiency of the decision-making process and ground decisions in sound reasoning, thereby reducing the likelihood of unforeseen roadblocks to adoption.

In addition, the checklist:

- Exposes key discussion topics for decision-makers at the beginning of the cloud adoption process.
- Supports thorough business discussions about regulations and the organization's own objectives for privacy, personally identifiable information (PII), and data security.
- Helps organizations identify any potential issues that could affect a cloud project.
- Provides a consistent set of questions, with the same terms, definitions, metrics, and deliverables for each provider, to simplify the process of comparing offerings from different cloud service providers.

Azure infrastructure and application security validation (detect)

[Azure Operational Security](#) refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure.

Insight & Analytics

- Quickly diagnose root cause across the full stack of modern applications and underlying infrastructure
- Monitor and alert on key metrics and KPIs in real time to rapidly identify problems
- Collect, process and analyze petabytes of data
- Create and share data insights across your company in minutes
- Integrate with and extend the value of existing monitoring tools

Protection & Recovery

- Protection of Cloud Assets (DR/Backup for IAAS, Backup of SQL PaaS)
- Enhanced Capacity Planning and Monitoring with Log Analytics
- Enterprise coverage with Linux distros, SQL AG, Encryption at rest
- Faster, Cheaper, Compact Backup Storage ([Xcool](#), [De-dup](#), [ReFS](#))
- Centralized hybrid backup monitoring and reporting in Azure
- Workload protection for public, hybrid, and private cloud
- Enterprise grade VMware VM Backup

Automation & Control

- Trigger immediate action in response to issues automatically or on-demand
- Maintain the state of IT resources and resolve configuration drifts
- Keep IT systems updated with minimal downtime
- Track and manage changes with ease

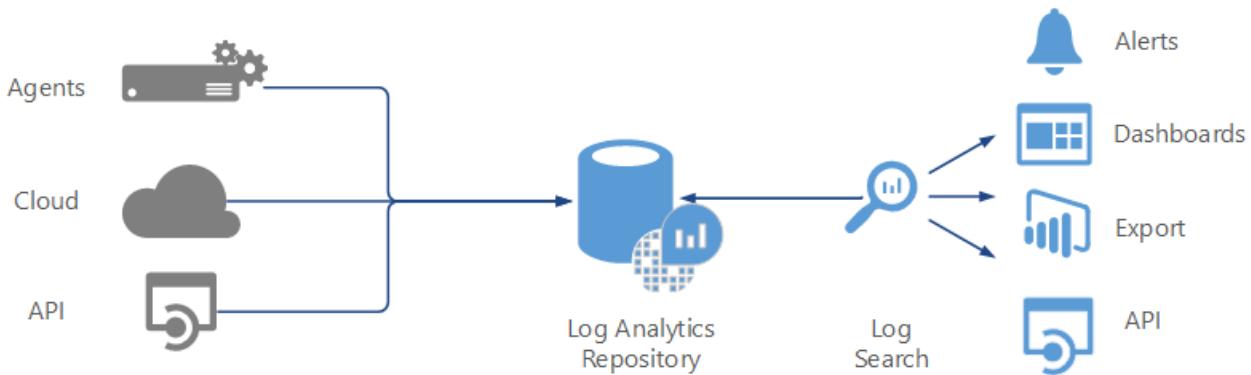
Security & Compliance

- Collection of security data from virtually any source
- Insight into security status (antimalware, system updates)
- Correlations to detect malicious activities and search for rapid investigation
- Integrates operational and security management
- Threat detection using advanced analytics

Azure Operational Security is built on a framework that incorporates the knowledge gained through a various capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Centre program, and deep awareness of the cybersecurity threat landscape.

Microsoft operations management suite(OMS)

Microsoft Operations Management Suite (OMS) is the IT management solution for the hybrid cloud. Used alone or to extend your existing System Center deployment, OMS gives you the maximum flexibility and control for cloud-based management of your infrastructure.



With OMS, you can manage any instance in any cloud, including on-premises, Azure, AWS, Windows Server, Linux, VMware, and OpenStack, at a lower cost than competitive solutions. Built for the cloud-first world, OMS offers a new approach to managing your enterprise that is the fastest, most cost-effective way to meet new business challenges and accommodate new workloads, applications and cloud environments.

Log analytics

[Log Analytics](#) provides monitoring services for OMS by collecting data from managed resources into a central repository. This data could include events, performance data, or custom data provided through the API. Once collected, the data is available for alerting, analysis, and export.

DESCRIPTION	RESOURCE	STATE	SEVERITY
Enable VM Agent	3 virtual mac...	Open	High
Install Endpoint Protection	8 virtual mac...	Open	High
Add a web application firewall	2 web applic...	Open	High
Add a Next Generation Firewall	6 endpoints	Open	High
Finalize Internet facing endpoint protec...	VM3-RDP-M...	Open	High
Enable Network Security Groups on sub...	3 subnets	Open	High
Enable Network Security Groups on virt...	vm1classic	Open	High
Route traffic through NGFW only	vm3	Open	High
Enable Auditing & Threat detection on...	sqlserver1as...	Open	High
Remediate vulnerabilities (by Qualys)	2 virtual mac...	Open	High
Enable Auditing & Threat detection on...	2 SQL datab...	Open	High
Apply a Just-In-Time network access co...	7 virtual mac...	Open	High
Apply system updates	3 virtual mac...	Open	High
Apply disk encryption	12 virtual ma...	Open	High
Enable encryption for Azure Storage Ac...	19 storage a...	Open	High
Restrict access through Internet facing...	6 virtual mac...	Open	Medium
Add a vulnerability assessment solution	8 virtual mac...	Open	Medium

This method allows you to consolidate data from a variety of sources, so you can combine data from your Azure services with your existing on-premises environment. It also clearly separates the collection of the data from the action taken on that data so that all actions are available to all kinds of data.

Azure security center

[Azure Security Center](#) helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Security Center analyzes the security state of your Azure resources to identify potential security vulnerabilities. A list of recommendations guides you through the process of configuring needed controls.

Examples include:

- Provisioning antimalware to help identify and remove malicious software
- Configuring network security groups and rules to control traffic to VMs
- Provisioning of web application firewalls to help defend against attacks that target your web applications
- Deploying missing system updates
- Addressing OS configurations that do not match the recommended baselines

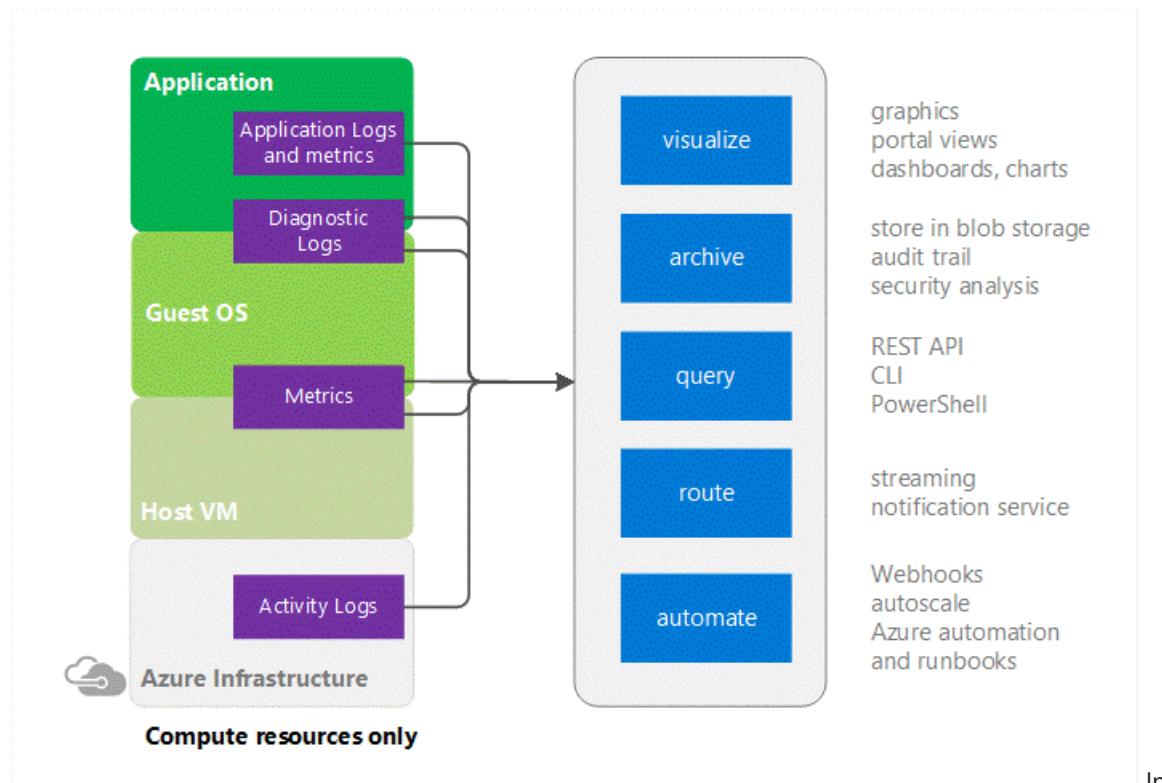
Security Center automatically collects, analyzes, and integrates log data from your Azure resources, the network, and partner solutions like antimalware programs and firewalls. When threats are detected, a security alert is created. Examples include detection of:

- Compromised VMs communicating with known malicious IP addresses
- Advanced malware detected by using Windows error reporting
- Brute force attacks against VMs
- Security alerts from integrated antimalware programs and firewalls

Azure monitor

[Azure Monitor](#) provides pointers to information on specific types of resources. It offers visualization, query, routing, alerting, auto scale, and automation on data both from the Azure infrastructure (Activity Log) and each individual Azure resource (Diagnostic Logs).

Cloud applications are complex with many moving parts. Monitoring provides data to ensure that your application stays up and running in a healthy state. It also helps you to stave off potential problems or troubleshoot past ones.



In addition, you can use monitoring data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention.

Auditing your network security is vital for detecting network vulnerabilities and ensuring compliance with your IT security and regulatory governance model. With Security Group view, you can retrieve the configured Network Security Group and security rules, as well as the effective security rules. With the list of rules applied, you can determine the ports that are open and ss network vulnerability.

Network watcher

[Network Watcher](#) is a regional service that enables you to monitor and diagnose conditions at a network level in, to, and from Azure. Network diagnostic and visualization tools available with Network Watcher help you understand, diagnose, and gain insights to your network in Azure. This service includes packet capture, next hop, IP flow verify, security group view, NSG flow logs. Scenario level monitoring provides an end to end view of network resources in contrast to individual network resource monitoring.

Storage analytics

[Storage Analytics](#) can store metrics that include aggregated transaction statistics and capacity data about requests to a storage service. Transactions are reported at both the API operation level as well as at the storage service level, and capacity is reported at the storage service level. Metrics data can be used to analyze storage service usage, diagnose issues with requests made against the storage service, and to improve the performance of applications that use a service.

Application insights

[Application Insights](#) is an extensible Application Performance Management (APM) service for web developers on multiple platforms. Use it to monitor your live web application. It will automatically detect performance anomalies. It includes powerful analytics tools to help you diagnose issues and to understand what users do with your app. It's designed to help you continuously improve performance and usability. It works for apps on a wide variety of platforms including .NET, Node.js and J2EE, hosted on-premises or in the cloud. It integrates with your devOps process, and has connection points to a various development tools.

It monitors:

- **Request rates, response times, and failure rates** - Find out which pages are most popular, at what times of day, and where your users are. See which pages perform best. If your response times and failure rates go high when there are more requests, then perhaps you have a resourcing problem.
- **Dependency rates, response times, and failure rates** - Find out whether external services are slowing you down.
- **Exceptions** - Analyze the aggregated statistics, or pick specific instances and drill into the stack trace and related requests. Both server and browser exceptions are reported.
- **Page views and load performance** - reported by your users' browsers.
- **AJAX calls from web pages** - rates, response times, and failure rates.
- **User and session counts**.
- **Performance counters** from your Windows or Linux server machines, such as CPU, memory, and network usage.
- **Host diagnostics** from Docker or Azure.
- **Diagnostic trace logs** from your app - so that you can correlate trace events with requests.
- **Custom events and metrics** that you write yourself in the client or server code, to track business events such as items sold, or games won. The infrastructure for your application is typically made up of many components – maybe a virtual machine, storage account, and virtual network, or a web app, database, database server, and 3rd party services. You do not see these components as separate entities, instead you see them as related and interdependent parts of a single entity. You want to deploy, manage, and monitor them as a group. [Azure Resource Manager](#) enables you to work with the resources in your solution as a group.

You can deploy, update, or delete all the resources for your solution in a single, coordinated operation. You use a template for deployment and that template can work for different environments such as testing, staging, and

production. Resource Manager provides security, auditing, and tagging features to help you manage your resources after deployment.

The benefits of using Resource Manager

Resource Manager provides several benefits:

- You can deploy, manage, and monitor all the resources for your solution as a group, rather than handling these resources individually.
- You can repeatedly deploy your solution throughout the development lifecycle and have confidence your resources are deployed in a consistent state.
- You can manage your infrastructure through declarative templates rather than scripts.
- You can define the dependencies between resources, so they are deployed in the correct order.
- You can apply access control to all services in your resource group because Role-Based Access Control (RBAC) is natively integrated into the management platform.
- You can apply tags to resources to logically organize all the resources in your subscription.
- You can clarify your organization's billing by viewing costs for a group of resources sharing the same tag.

NOTE

Resource Manager provides a new way to deploy and manage your solutions. If you used the earlier deployment model and want to learn about the changes, see [Understanding Resource Manager Deployment and classic deployment](#).

Next steps

Find out more about security by reading some of our in-depth security topics:

- [Auditing and logging](#)
- [Cybercrime](#)
- [Design and operational security](#)
- [Encryption](#)
- [Identity and access management](#)
- [Network security](#)
- [Threat management](#)

Governance in Azure

8/10/2017 • 31 min to read • [Edit Online](#)

We know that security is job one in the cloud and how important it is that you find accurate and timely information about Azure security. One of the best reasons to use Azure for your applications and services is to take advantage of its wide array of security tools and capabilities. These tools and capabilities help make it possible to create secure solutions on the secure Azure platform.

To help you better understand the array of Governance controls implemented within Microsoft Azure from both the customer's and Microsoft operations' perspectives, this article, "Governance in Azure", is written that provides a comprehensive look at the Governance features available with Microsoft Azure.

Azure platform

Azure is a public cloud service platform that supports a broad selection of operating systems, programming languages, frameworks, tools, databases and devices. It can run Linux containers with Docker integration; build apps with JavaScript, Python, .NET, PHP, Java and Node.js; build back-ends for iOS, Android and Windows devices. Azure public cloud services support the same technologies millions of developers and IT professionals already rely on and trust.

When you build on, or migrate IT assets to, a public cloud service provider you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security requirements. In addition, Azure provides you many security options and the ability to control them so that you can customize security to meet the unique requirements of your organization's deployments.

This document will help you understand how Azure Governance capabilities can help you fulfill these requirements.

Abstract

Microsoft Azure cloud governance provides an integrated audit and consulting approach for reviewing and advising organizations on their usage of the Azure platform. Microsoft Azure cloud governance refers to the decision-making processes, criteria and policies involved in the planning, architecture, acquisition, deployment, operation and management of a Cloud computing.

To create a plan for Microsoft Azure cloud governance, you need to take an in-depth look at the people, processes, and technologies currently in place, and then build frameworks that make it easy for IT to consistently support business needs while providing end users with the flexibility to use the powerful features of Microsoft Azure.

This paper describes how you can achieve an elevated level of governance of your IT resources in Microsoft Azure. This paper can help you understand the security and governance features built in to Microsoft Azure.

The following are main the governance issues discussed in this paper:

- Implementation of policies, processes and procedures as per organization goals.
- Security and continuous compliance with organization standards
- Alerting and Monitoring

Implementation of policies, processes and procedures

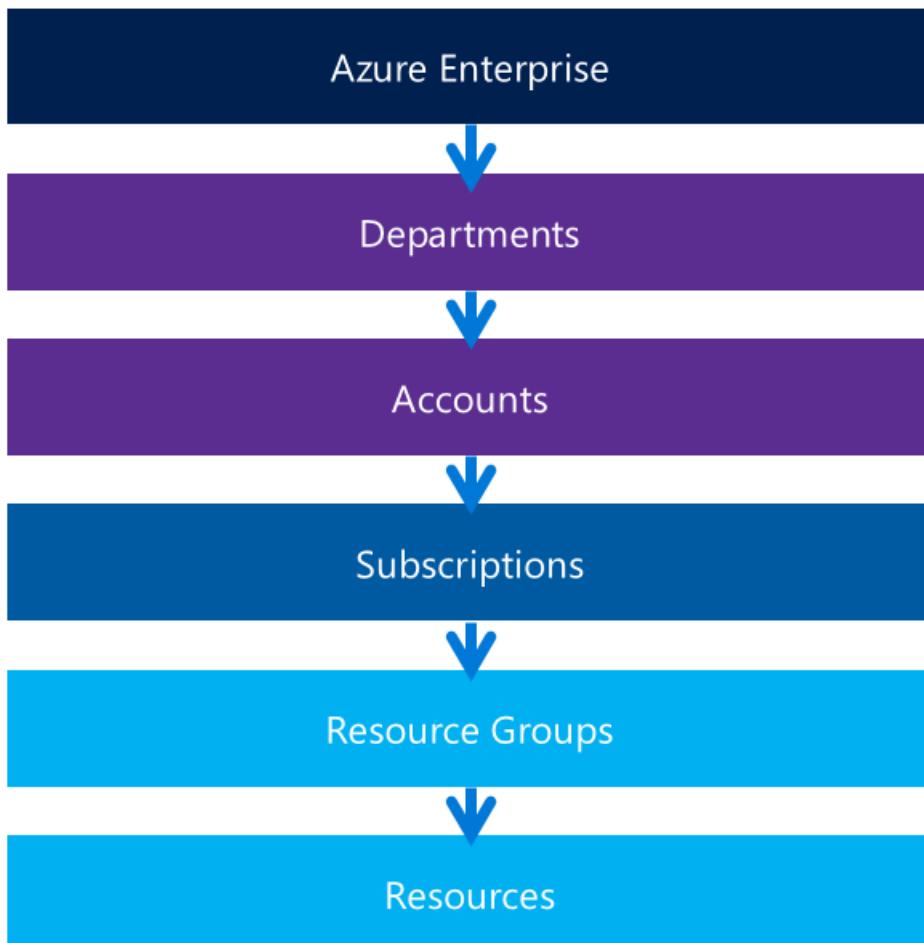
Management has established roles and responsibilities to oversee implementation of the information security policy and operational continuity across Azure. Microsoft Azure management is responsible for overseeing security and continuity practices within their respective teams (including third parties), and facilitating compliance with security policies, processes and standards.

Here are the factors evolved:

- Account Provisioning
- Subscription Controls
- Role Based access controls
- Resource Management
- Resource tracking
- Critical Resource Control
- API Access to Billing Information
- Networking Controls

Account provisioning

Defining account hierarchy is a major step to use and structure Azure services within a company and is the core governance structure. In case of customers with the enterprise agreement, customers can further subdivide the environment into departments, accounts, and finally, subscriptions.



If you do not have an enterprise agreement, consider using [Azure tags](#) at subscription level to define hierarchy. An

Azure subscription is the basic unit where all resources are contained. It also defines several limits within Azure, such as number of cores, resources, etc. Subscriptions can contain [Resource Groups](#), which can contain Resources. [RBAC](#) principles apply on those three levels.

Every enterprise is different and the hierarchy using Azure Tags in case of non-enterprise customers allows for significant flexibility in how Azure is organized within the company. Before deploying resources in Microsoft Azure, you should model hierarchy and understand the impact on billing, resource access, and complexity.

Subscription controls

Subscription controls how resources usage is reported and billed. Subscriptions can be setup for separate billing and payment. As mentioned earlier under one Azure account we can have multiple subscriptions. Subscriptions can be used to determine the Azure resource usage of multiple departments in a company.

For example, if a company has IT, HR and Marketing departments and these departments have different projects running. Based on the usage of Azure resources like virtual machines by each department, they can be billed accordingly. By this we can control the finances of each department.

Azure subscriptions establish three parameters:

- a unique subscriber ID
- a billing location
- Set of available resources

For an individual, that would include one Microsoft account ID, a credit card number and the full suite of Azure services -- although, Microsoft enforces consumption limits, depending on the subscription type.

Azure enrollment hierarchies define how services are structured within an Enterprise Agreement. The Enterprise Portal allows customers to divide access to Azure resources associated with an Enterprise Agreement based on flexible hierarchies customizable to an organization's unique needs. The hierarchy pattern should match an organization's management and geographic structure so that the associated billing and resource access can be accurately accounted for.

The three high-level patterns are functional, business unit, and geographic, using departments as an administrative construct for account groupings. Within each department, accounts can be assigned subscriptions, which create silos for billing and several key limits in Azure (e.g., number of VMs, storage accounts, etc.).

Azure Resource	Resource Manager API	Service Management API
Cores per subscription	20/10,000 per region	20/10,000 Global
Co-administrators per subscription	Unlimited	200/200 Global, with no RBAC model
Storage accounts per subscription	100/100 (250 by contacting support)	100/150 (250 by contacting support)
Cloud Services per subscription	N/A	20/200
Virtual networks per subscription	50/500	50/100
Local networks per subscription	10/500	20/Contact support
Reserved IPs per subscription	20/100	20/Contact support
Public IP addresses (dynamic)	60/Contact Support	400 Global
Reserved public IP addresses	20/Contact support	256 Global
Resource Groups per subscription	800/800	500 Global
Virtual machines per subscription	20/10,000 per region	50/50 per cloud service

For organizations with an Enterprise Agreement, Azure subscriptions follow a four-level hierarchy:

- enterprise enrolment administrator

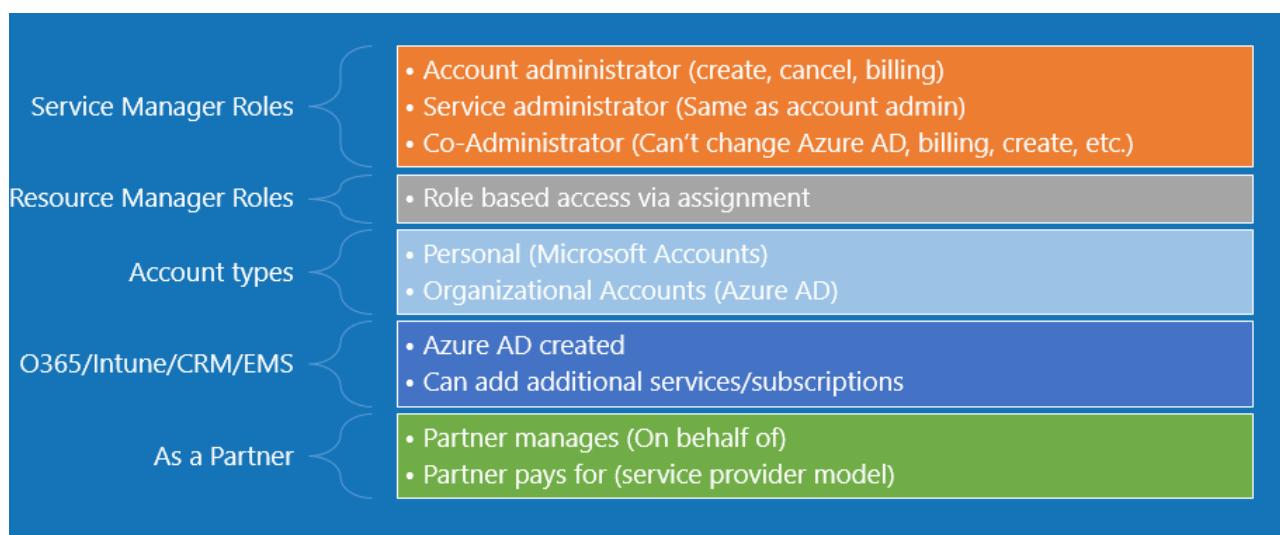
- department administrator
- account owner
- Service administrator

This hierarchy governs the following:

- Billing relationship
- Account administration
- Role Based Access Control (RBAC) to artifacts
- Boundaries/Limits
- Boundaries
 - Usage and billing (rate card based on offer numbers)
 - Limits
 - Virtual Network
- Attached to 1 AAD (1 AAD be associated with many subscriptions)
- Associated to an enterprise enrollment account

Role-based access controls

When Azure was initially released, access controls to a subscription were basic: Administrator or Co-Administrator. Access to a subscription in the Classic model implied access to all the resources in the portal. This lack of fine-grained control led to the proliferation of subscriptions to provide a level of reasonable access control for an Azure Enrollment.



This proliferation of subscriptions is no longer needed. With role-based access control, you can assign users to standard roles (such as common "reader" and "writer" types of roles). You can also define custom roles.

[Azure Role-Based Access Control \(RBAC\)](#) enables fine-grained access management for Azure. Using RBAC, you can grant only the amount of access that users need to perform their jobs. Security-oriented companies should focus on giving employees the exact permissions they need. Too many permissions expose an account to attackers. Too few permissions mean that employees can't get their work done efficiently. Azure Role-Based Access Control (RBAC) helps address this problem by offering fine-grained access management for Azure. RBAC helps you to segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Instead of giving everybody unrestricted permissions in your Azure subscription or resources, you can allow

only certain actions.

For example, use RBAC to let one employee manage virtual machines in a subscription, while another can manage SQL databases within the same subscription.

Azure RBAC has three basic roles that apply to all resource types:

- **Owner** has full access to all resources including the right to delegate access to others.
- **Contributor** can create and manage all types of Azure resources but can't grant access to others.
- **Reader** can view existing Azure resources.

The rest of the RBAC roles in Azure allow management of specific Azure resources. For example, the Virtual Machine Contributor role allows the user to create and manage virtual machines. It does not give them access to the virtual network or the subnet that the virtual machine connects to.

[RBAC built-in roles](#) lists the roles available in Azure. It specifies the operations and scope that each built-in role grants to users.

Grant access by assigning the appropriate RBAC role to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource. A role assigned at a parent scope also grants access to the children contained within it.

For example, a user with access to a resource group can manage all the resources it contains, like websites, virtual machines, and subnets.

Azure RBAC only supports management operations of the Azure resources in the Azure portal and Azure Resource Manager APIs. It cannot authorize all data level operations for Azure resources. For example, you can authorize someone to manage Storage Accounts, but not to the blobs or tables within a Storage Account cannot. Similarly, a SQL database can be managed, but not the tables within it.

If you want more details about how RBAC helps you manage access, see [What is Role-Based Access Control](#).

You can also [create a custom role](#) in Azure Role-Based Access Control (RBAC) if none of the built-in roles meet your specific access needs. Custom roles can be created using [Azure PowerShell](#), [Azure Command-Line Interface \(CLI\)](#), and the [REST API](#). Just like built-in roles, custom roles can be assigned to users, groups, and applications at subscription, resource group, and resource scopes.

Within each subscription, you can grant up to 2000 role assignments.

Resource management

Azure originally provided only the classic deployment model. In this model, each resource existed independently; there was no way to group related resources together. Instead, you had to manually track which resources made up your solution or application, and remember to manage them in a coordinated approach.

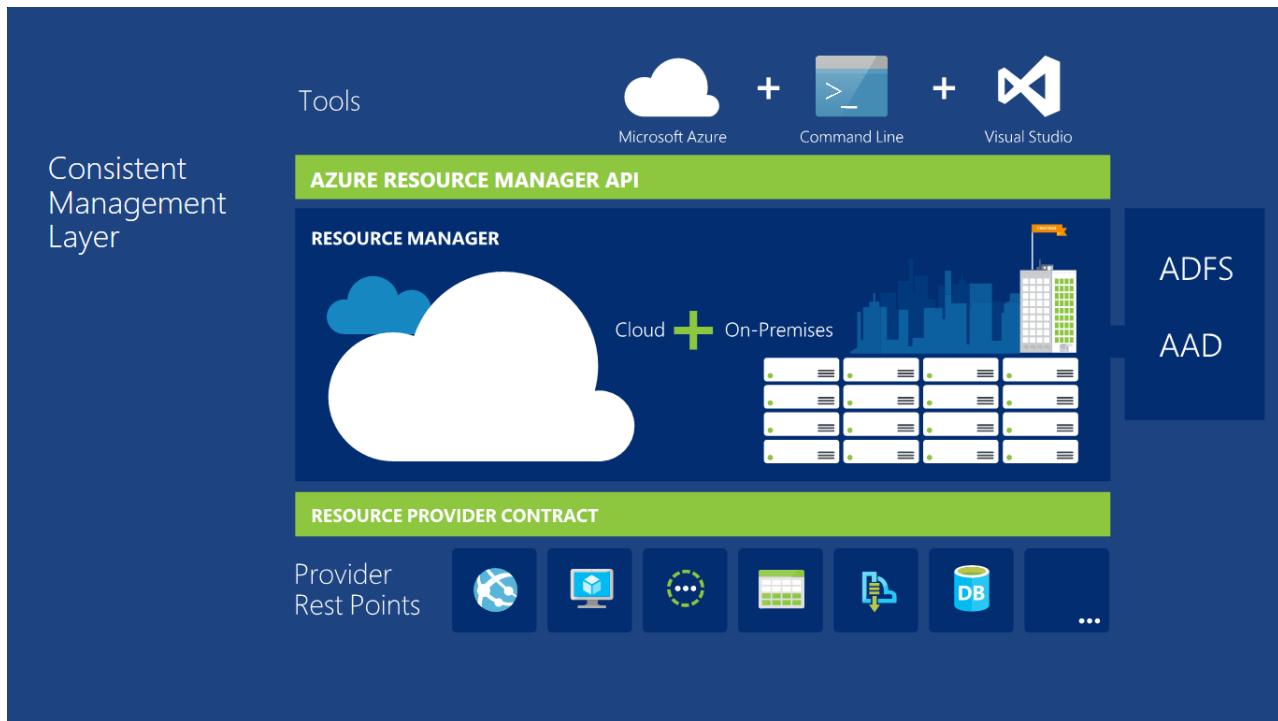
To deploy a solution, you had to either create each resource individually through the classic portal or create a script that deployed all the resources in the correct order. To delete a solution, you had to delete each resource individually. You could not easily apply and update access control policies for related resources. Finally, you could not apply tags to resources to label them with terms that help you monitor your resources and manage billing.

In 2014, Azure introduced Resource Manager, which added the concept of a resource group. A resource group is a container for resources that share a common lifecycle. The Resource Manager deployment model provides several benefits:

- You can deploy, manage, and monitor all the services for your solution as a group, rather than handling these services individually.
- You can repeatedly deploy your solution throughout its lifecycle and have confidence your resources are

deployed in a consistent state.

- You can apply access control to all resources in your resource group, and those policies are automatically applied when new resources are added to the resource group.
- You can apply tags to resources to logically organize all the resources in your subscription.
- You can use JavaScript Object Notation (JSON) to define the infrastructure for your solution. The JSON file is known as a Resource Manager template.
- You can define the dependencies between resources so they are deployed in the correct order.



Resource Manager enables you to put resources into meaningful groups for management, billing, or natural affinity. As mentioned earlier, Azure has two deployment models. In the earlier [Classic model](#), the basic unit of management was the subscription. It was difficult to break down resources within a subscription, which led to the creation of large numbers of subscriptions. With the Resource Manager model, we saw the introduction of resource groups.

A resource group is a container that holds related resources for an Azure solution. [The resource group](#) can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization.

For recommendations about templates, see [Best practices for creating Azure Resource Manager templates](#).

Azure Resource Manager analyzes dependencies to ensure resources are created in the correct order. If one resource relies on a value from another resource (such as a virtual machine needing a storage account for disks), you set a dependency.

NOTE

For more information, see [Defining dependencies in Azure Resource Manager templates](#).

You can also use the template for updates to the infrastructure. For example, you can add a resource to your solution and add configuration rules for the resources that are already deployed. If the template specifies creating a resource but that resource already exists, Azure Resource Manager performs an update instead of creating a new asset. Azure Resource Manager updates the existing asset to the same state as it would be as new.

Resource Manager provides extensions for scenarios when you need additional operations such as installing software that is not included in the setup.

Resource tracking

As users in your organization add resources to the subscription, it becomes increasingly important to associate resources with the appropriate department, customer, and environment. You can attach metadata to resources through tags. You use [tags](#) to provide information about the resource or the owner. Tags enable you to not only aggregate and group resources in several ways, but use that data for the purposes of chargeback.

Use tags when you have a complex collection of resource groups and resources, and need to visualize those assets in the way that makes the most sense to you. For example, you could tag resources that serve a similar role in your organization or belong to the same department.

Without tags, users in your organization can create multiple resources that may be difficult to later identify and manage. For example, you may wish to delete all the resources for a project. If those resources are not tagged for the project, you must manually find them. Tagging can be an important way for you to reduce unnecessary costs in your subscription.

Resources do not need to reside in the same resource group to share a tag. You can create your own tag taxonomy to ensure that all users in your organization use common tags rather than users inadvertently applying slightly different tags (such as "dept" instead of "department").

Resource policies enable you to create standard rules for your organization. You can create policies that ensure resources are tagged with the appropriate values.

NOTE

For more information, see [Apply resource policies for tags](#).

You can also view tagged resources through the Azure portal.

The [usage report](#) for your subscription includes tag names and values, which enables you to break out costs by tags.

NOTE

For more information about tags, see [Using tags to organize your Azure resources](#).

The following limitations apply to tags:

- Each resource or resource group can have a maximum of 15 tag key/value pairs. This limitation only applies to tags directly applied to the resource group or resource. A resource group can contain many resources that each have 15 tag key/value pairs.
- The tag name is limited to 512 characters.
- The tag value is limited to 256 characters.
- Tags applied to the resource group are not inherited by the resources in that resource group.

If you have more than 15 values that you need to associate with a resource, use a JSON string for the tag value. The JSON string can contain many values that are applied to a single tag key.

Tags and billing

Tags enable you to group your billing data. For example, if you are running multiple VMs for different

organizations, use the tags to group usage by cost center. You can also use tags to categorize costs by runtime environment; such as, the billing usage for VMs running in production environment.

You can retrieve information about tags through the [Azure Resource Usage and RateCard APIs](#) or the usage comma-separated values (CSV) file. You download the usage file from the [Azure accounts portal](#) or EA portal.

NOTE

For more information about programmatic access to billing information, see [Gain insights into your Microsoft Azure resource consumption](#). For REST API operations, see [Azure Billing REST API Reference](#).

When you download the usage CSV for services that support tags with billing, the tags appear in the Tags column.

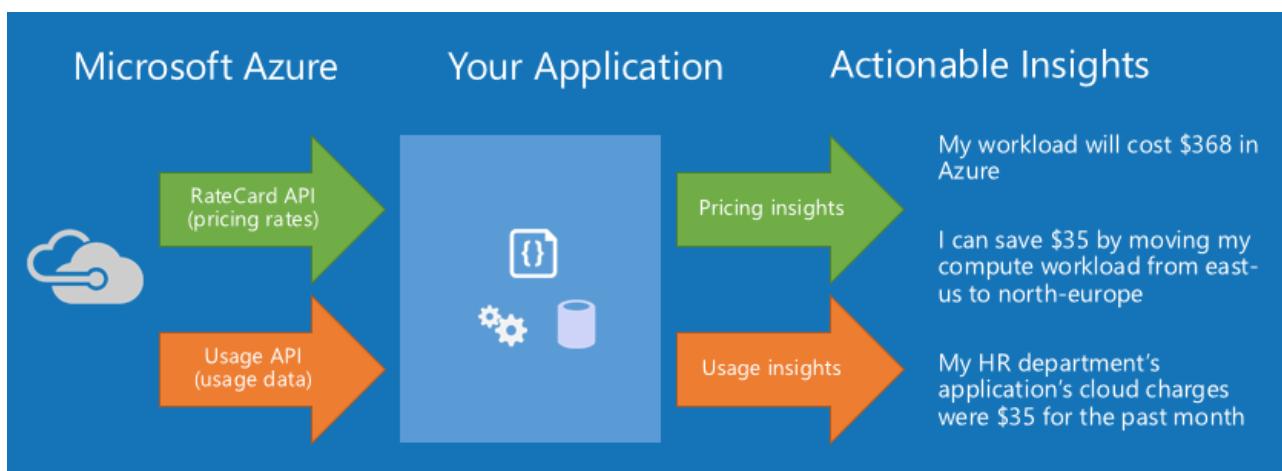
Critical resource controls

As your organization adds core services to the subscription, it becomes increasingly important to ensure that those services are available to avoid business disruption. [Resource locks](#) enable you to restrict operations on high-value resources where modifying or deleting them would have a significant impact on your applications or cloud infrastructure. You can apply locks to a subscription, resource group, or resource. Typically, you apply locks to foundational resources such as virtual networks, gateways, and storage accounts.

Resource locks currently support two values: CanNotDelete and ReadOnly. CanNotDelete means that users (with the appropriate rights) can still read or modify a resource but cannot delete it. ReadOnly means that authorized users can't delete or modify a resource.

Resource Manager Locks apply only to operations that happen in the management plane, which consists of operations sent to <https://management.azure.com>. The locks do not restrict how resources perform their own functions. Resource changes are restricted, but resource operations are not restricted. For example, a ReadOnly lock on a SQL Database prevents you from deleting or modifying the database, but it does not prevent you from creating, updating, or deleting data in the database.

Applying **ReadOnly** can lead to unexpected results because some operations that seem like read operations require additional actions. For example, placing a **ReadOnly** lock on a storage account prevents all users from listing the keys. The list keys operation is handled through a POST request because the returned keys are available for write operations.



For another example, placing a **ReadOnly** lock on an App Service resource prevents Visual Studio Server Explorer from displaying files for the resource because that interaction requires write access.

Unlike role-based access control, you use management locks to apply a restriction across all users and roles. To learn about setting permissions for users and roles, see [Azure Role-based Access Control](#).

When you apply a lock at a parent scope, all resources within that scope inherit the same lock. Even resources you

add later inherit the lock from the parent. The most restrictive lock in the inheritance takes precedence.

To create or delete management locks, you must have access to `Microsoft.Authorization/` or `Microsoft.Authorization/locks/` actions. Of the built-in roles, only **Owner** and **User Access Administrator** are granted those actions.

API access to billing information

Use Azure Billing APIs to pull usage and resource data into your preferred data analysis tools. The Azure Resource Usage and RateCard APIs can help you accurately predict and manage your costs. The APIs are implemented as a Resource Provider and part of the family of APIs exposed by the Azure Resource Manager.

Azure resource usage API (Preview)

Use the Azure [Resource Usage API](#) to get your estimated Azure consumption data. The API includes:

- **Azure Role-based Access Control** - Configure access policies on the [Azure portal](#) or through [Azure PowerShell cmdlets](#) to specify which users or applications can get access to the subscription's usage data. Callers must use standard Azure Active Directory tokens for authentication. Add the caller to either the Billing Reader, Reader, Owner, or Contributor role to get access to the usage data for a specific Azure subscription.
- **Hourly or Daily Aggregations** - Callers can specify whether they want their Azure usage data in hourly buckets or daily buckets. The default is daily.
- **Instance metadata (includes resource tags)** – Get instance-level detail like the fully qualified resource uri (`/subscriptions/{subscription-id} /.`), the resource group information, and resource tags. This metadata helps you deterministically and programmatically allocate usage by the tags, for use-cases like cross-charging.
- **Resource metadata** - Resource details such as the meter name, meter category, meter sub category, unit, and region give the caller a better understanding of what was consumed. We're also working to align resource metadata terminology across the Azure portal, Azure usage CSV, EA billing CSV, and other public-facing experiences, to let you correlate data across experiences.
- **Usage for all offer types** – Usage data is available for all offer types like Pay-as-you-go, MSDN, Monetary commitment, Monetary credit, and EA.

Azure resource RateCard API (Preview)

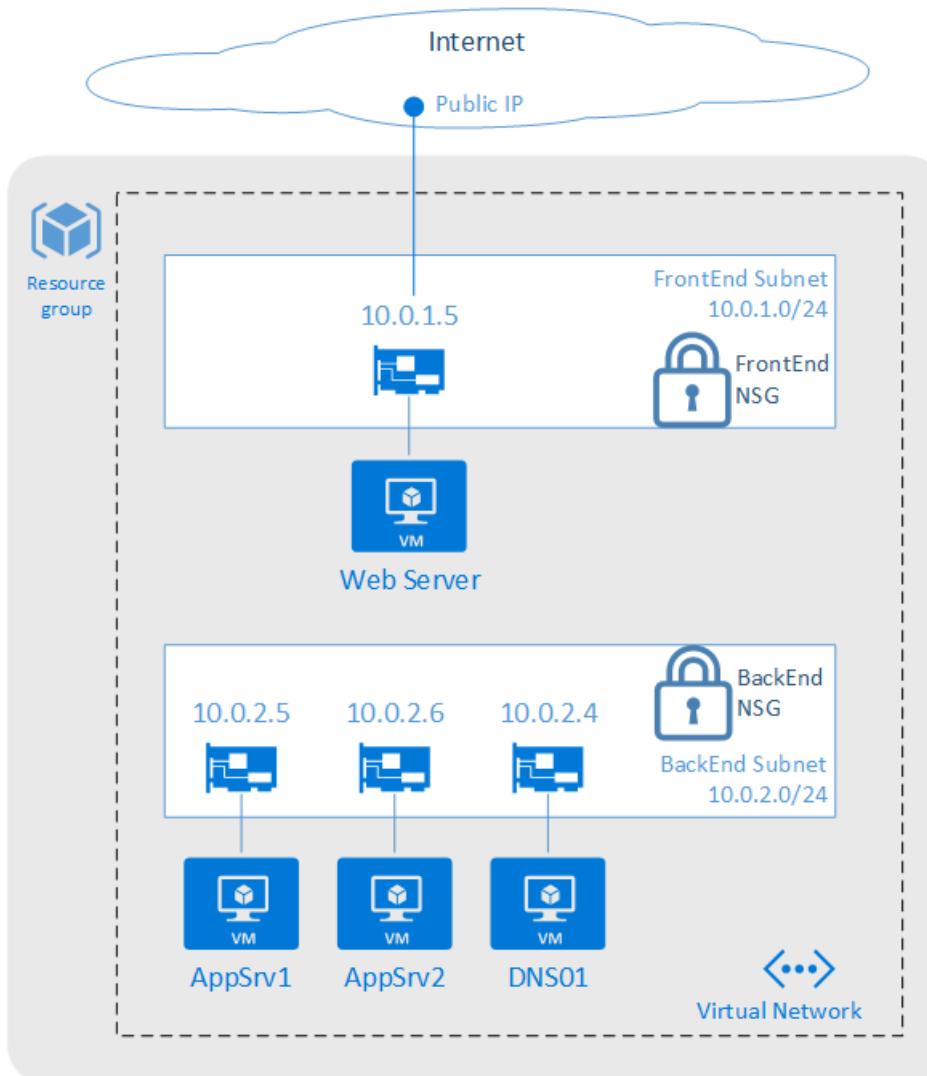
Use the Azure Resource RateCard API to get the list of available Azure resources and estimated pricing information for each. The API includes:

- **Azure Role-based Access Control** - Configure your access policies on the Azure portal or through Azure PowerShell cmdlets to specify which users or applications can get access to the RateCard data. Callers must use standard Azure Active Directory tokens for authentication. Add the caller to either the Reader, Owner, or Contributor role to get access to the usage data for a particular Azure subscription.
- **Support for Pay-as-you-go, MSDN, Monetary commitment, and Monetary credit offers (EA not supported)** - This API provides Azure offer-level rate information. The caller of this API must pass in the offer information to get resource details and rates. We're currently unable to provide EA rates because EA offers have customized rates per enrollment. Here are some of the scenarios that are made possible with the combination of the Usage and the RateCard APIs:
 - **Azure spend during the month** - Use the combination of the Usage and RateCard APIs to get better insights into your cloud spend during the month. You can analyze the hourly and daily buckets of usage and charge estimates.
 - **Set up alerts** – Use the Usage and the RateCard APIs to get estimated cloud consumption and charges, and set up resource-based or monetary-based alerts.

- **Predict bill** – Get your estimated consumption and cloud spend, and apply machine learning algorithms to predict what the bill would be at the end of the billing cycle.
- **Pre-consumption cost analysis** – Use the RateCard API to predict how much your bill would be for your expected usage when you move your workloads to Azure. If you have existing workloads in other clouds or private clouds, you can also map your usage with the Azure rates to get a better estimate of Azure spend. This estimate gives you the ability to pivot on offer, and compare between the different offer types beyond Pay-As-You-Go, like monetary commitment and monetary credit. The API also gives you the ability to see cost differences by region and allows you to do a what-if cost analysis to help you make deployment decisions.
- **What-if analysis** - You can determine whether it is more cost-effective to run workloads in another region, or on another configuration of the Azure resource. Azure resource costs may differ based on the Azure region you're using.
- You can also determine if another Azure offer type gives a better rate on an Azure resource.

Networking controls

Access to resources can be either internal (within the corporation's network) or external (through the internet). It is easy for users in your organization to inadvertently put resources in the wrong spot, and potentially open them to malicious access. As with on premises/ devices, enterprises must add appropriate controls to ensure that Azure users make the right decisions.



For subscription governance, we identify core resources that provide basic control of access. The core resources consist of:

Network connectivity

[Virtual Networks](#) are container objects for subnets. Though not strictly necessary, it is often used when connecting applications to internal corporate resources. The Azure Virtual Network service enables you to securely connect Azure resources to each other with virtual networks (VNets).

A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure cloud dedicated to your subscription. You can also connect VNets to your on-premises network.

Following are capabilities for Azure Virtual Networks:

- **Isolation:** VNets are isolated from one another. You can create separate VNets for development, testing, and production that use the same CIDR address blocks. Conversely, you can create multiple VNets that use different CIDR address blocks and connect networks together. You can segment a VNet into multiple subnets. Azure provides internal name resolution for VMs and Cloud Services role instances connected to a VNet. You can optionally configure a VNet to use your own DNS servers, instead of using Azure internal name resolution.
- **Internet connectivity:** All Azure Virtual Machines (VM) and Cloud Services role instances connected to a VNet have access to the Internet, by default. You can also enable inbound access to specific resources, as needed.
- **Azure resource connectivity:** Azure resources such as Cloud Services and VMs can be connected to the same VNet. The resources can connect to each other using private IP addresses, even if they are in different subnets. Azure provides default routing between subnets, VNets, and on-premises networks, so you don't have to configure and manage routes.
- **VNet connectivity:** VNets can be connected to each other, enabling resources connected to any VNet to communicate with any resource on any other VNet.
- **On-premises connectivity:** VNets can be connected to on-premises networks through private network connections between your network and Azure, or through a site-to-site VPN connection over the Internet.
- **Traffic filtering:** VM and Cloud Services role instances network traffic can be filtered inbound and outbound by source IP address and port, destination IP address and port, and protocol.
- **Routing:** You can optionally override Azure's default routing by configuring your own routes, or using BGP routes through a network gateway.

Network access controls

[Network security groups](#) are like a firewall and provide rules for how a resource can "talk" over the network. They provide granular control over how/if a subnet (or virtual machine) can connect to the Internet or other subnets in the same virtual network.

A network security group (NSG) contains a list of security rules that allow or deny network traffic to resources connected to Azure Virtual Networks (VNet). NSGs can be associated to subnets, individual VMs (classic), or individual network interfaces (NIC) attached to VMs (Resource Manager).

When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet. Traffic can further be restricted by also associating an NSG to a VM or NIC.

Security and continuous compliance with organizational standards

Every business has different needs, and every business will reap distinct benefits from cloud solutions. Still, customers of all kinds have the same basic concerns about moving to the cloud. They want to retain control of their data, and they want that data to be kept secure and private, all while maintaining transparency and compliance.

What customers want from cloud providers is:

- **Secure our data** while acknowledging that the cloud can provide increased data security and administrative control, IT leaders are still concerned that migrating to the cloud will leave them more vulnerable to hackers than their current in-house solutions.
- **Keep our data** private Cloud services raise unique privacy challenges for businesses. As companies look to the cloud to save on infrastructure costs and improve their flexibility, they also worry about losing control of where their data is stored, who is accessing it, and how it gets used.
- **Give us control** Even as they take advantage of the cloud to deploy more innovative solutions, companies are very concerned about losing control of their data. The recent disclosures of government agencies accessing customer data, through both legal and extra-legal means, make some CIOs wary of storing their data in the cloud.
- **Promote transparency** While security, privacy, and control are important to business decision-makers, they also want the ability to independently verify how their data is being stored, accessed, and secured.
- **Maintain compliance** as companies expand their use of cloud technologies, the complexity and scope of standards and regulations continue to evolve. Companies need to know that their compliance standards will be met, and that compliance will evolve as regulations change over time.

Security configuration, monitoring and alerting

Azure subscribers may manage their cloud environments from multiple devices, including management workstations, developer PCs, and even privileged end-user devices that have task-specific permissions. In some cases, administrative functions are performed through web-based consoles such as the Azure portal. In other cases, there may be direct connections to Azure from on-premises systems over Virtual Private Networks (VPNs), Terminal Services, client application protocols, or (programmatically) the Azure Service Management API (SMAPI). Additionally, client endpoints can be either domain joined or isolated and unmanaged, such as tablets or smartphones.

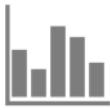
Although multiple access and management capabilities provide a rich set of options, this variability can add significant risk to a cloud deployment. It can be difficult to manage, track, and audit administrative actions. This variability may also introduce security threats through unregulated access to client endpoints that are used for managing cloud services. Using general or personal workstations for developing and managing infrastructure opens unpredictable threat vectors such as web browsing (for example, watering hole attacks) or email (for example, social engineering and phishing).

Monitoring, logging, and auditing provide a basis for tracking and understanding administrative activities, but it may not always be feasible to audit all actions in complete detail due to the amount of data generated. Auditing the effectiveness of the management policies is a best practice, however.

Azure security Governance from AD DS GPOs to control all the administrators' Windows interfaces, such as file sharing. Include management workstations in auditing, monitoring, and logging processes. Track all administrator and developer access and usage.

Azure security center

The [Azure Security Center](#) provides a central view of the security status of resources in the subscriptions, and provides recommendations that help prevent compromised resources. It can enable more granular policies (for example, applying policies to specific resource groups that allow the enterprise to tailor their posture to the risk they are addressing).



Visibility

- Security Resource Health
- Event Detection
- Recommendations



Control

- Policy
- Solutions



Analytics

- PowerBI Content Pack
- Operations Management Suite (OMS)

Security Center provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions. After you enable [security policies](#) for a subscription's resources, Security Center analyzes the security of your resources to identify potential vulnerabilities. Information about your network configuration is available instantly.

Azure Security Center represents a combination of best practice analysis and security policy management for all resources within an Azure subscription. This powerful and easy to use tool allows security teams and risk officers to prevent, detect, and respond to security threats as it automatically collects and analyzes security data from your Azure resources, the network, and partner solutions like anti-malware programs and firewalls.

In addition, Azure Security Center applies advanced analytics, including machine learning and behavioral analysis while leveraging global threat intelligence from Microsoft products and services, the Microsoft Digital Crimes Unit (DCU), the Microsoft Security Response Center (MSRC), and external feeds. [Security governance](#) can be applied broadly at the subscription level or narrowed down to specific, granular requirements applied to individual resources through policy definition.

Finally, Azure Security Center analyzes resource security health based on those policies and uses this to provide insightful dashboards and alerting for events such as malware detection or malicious IP connection attempts.

NOTE

For more information about how to apply recommendations, read [Implementing security recommendations in Azure Security Center](#).

Security Center collects data from your virtual machines to assess their security state, provide security recommendations, and alert you to threats. When you first access Security Center, data collection is enabled on all virtual machines in your subscription. Data collection is recommended but you can opt-out by [disabling data collection](#) in the Security Center policy.

Finally, Azure Security Center is an open platform that enables Microsoft partners and independent software vendors to create software that plugs into Azure Security Center to enhance its capabilities.

Azure Security Center monitors the following Azure resources:

- Virtual machines (VMs) (including Cloud Services)
- Azure Virtual Networks
- Azure SQL service
- Partner solutions integrated with your Azure subscription such as a web application firewall on VMs and on [App Service Environment](#).

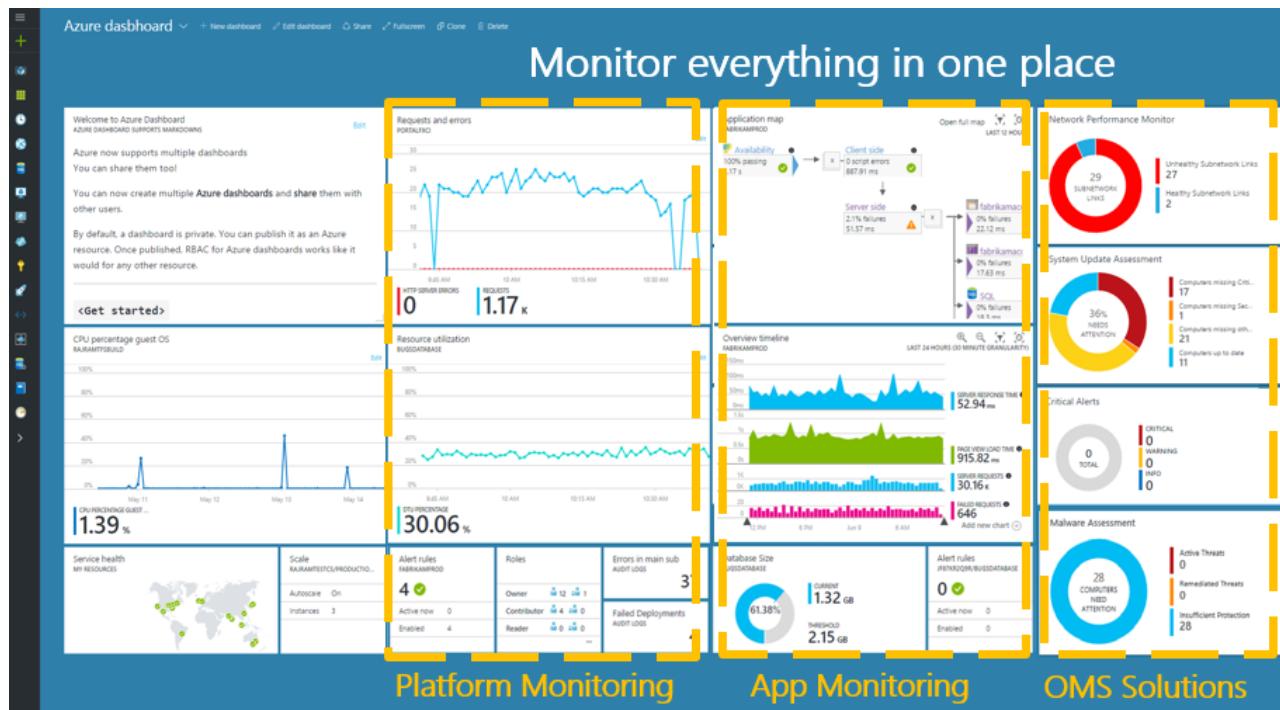
Operations Management Suite

The OMS software development and service team's information security and [governance program](#) supports its business requirements and adheres to laws and regulations as described at [Microsoft Azure Trust Center](#) and [Microsoft Trust Center Compliance](#). How OMS establish security requirements, identifies security controls, manages, and monitors risks are also described there. Annually, we review polices, standards, procedures, and guidelines.

Each OMS development team member receives formal application security training. Internally, we use a version control system for software development. Each software project is protected by the version control system.

Microsoft has a security and compliance team that oversees and assesses all services in Microsoft. Information security officers make up the team and they are not associated with the engineering departments that develop OMS. The security officers have their own management chain and conduct independent assessments of products and services to ensure security and compliance.

Operations Management Suite (also known as OMS) is a collection of management services that were designed in the cloud from the start. Rather than deploying and managing on premises resources, OMS components are entirely hosted in Azure. Configuration is minimal, and you can be up and running literally in a matter of minutes.



Just because OMS services run in the cloud doesn't mean that they can't effectively manage your on-premises environment.

Put an agent on any Windows or Linux computer in your data center, and it will send data to Log Analytics where it can be analyzed along with all other data collected from cloud or on premises services. Use Azure Backup and Azure Site Recovery to leverage the cloud for backup and high availability for on premises resources.

Runbooks in the cloud can't typically access your on-premises resources, but you can install an agent on one or more computers too that will host runbooks in your data center. When you start a runbook, you simply specify whether you want it to run in the cloud or on a local worker.

The core functionality of OMS is provided by a set of services that run in Azure. Each service provides a specific management function, and you can combine services to achieve different management scenarios.

Service	Description
 Log Analytics	Monitor and analyze the availability and performance of different resources including physical and virtual machines.
 Automation	Automate manual processes and enforce configurations for physical and virtual machines.
 Backup	Backup and restore critical data.
 Site Recovery	Provide high availability for critical applications.

Azure operation manager extends its functionalities by providing management solutions. [Management Solutions](#) are prepackaged sets of logic that implement a management scenario leveraging one or more OMS services.

Solutions Gallery

 Antimalware Assessment Owned View status of antivirus and antimalware scans across your servers.	 Automation Hybrid Worker Owned Create Hybrid Runbook Workers to run Automation runbooks on your on-premises servers.	 Backup Owned Manage Azure IaaS VM backup and Windows Server backup status for your backup vault.	 Upgrade Analytics (Preview) Owned Use a data-driven approach to streamline and accelerate Windows upgrades.	 Network Performance Monitor (Preview) Owned Offers near real time monitoring of network performance parameters like loss and latency.	 Security and Audit Owned Provides the ability to explore security related data and helps identify security breaches.	 Service Map Owned Automatically discover and map servers and their dependencies in real-time.	 SQL Assessment Owned Assess the risk and health of SQL Server environments.
 Activity Log Analytics Owned Track all create, update and delete activities occurring in your Azure subscriptions.	 Azure Networking Analytics (Preview) Owned Gain insight into your Azure Network Security Group and Application Gateway logs.	 Change Tracking Owned Track configuration changes across your servers.	 Containers Owned See Docker container performance metrics and logs from containers across your public or private cloud environments.	 Office 365 Analytics (Preview) Owned Get full visibility into your Office 365 user activities, perform forensics as well as audit and compliance.	 Service Fabric Analytics Owned Identify and troubleshoot issues across your Service Fabric cluster.	 Protection & Recovery Owned Monitor virtual machine replication status for your Azure Site Recovery Vault.	 Surface Hub Owned Provides the ability to monitor Microsoft Surface Hub devices.

Different solutions are available from Microsoft and from partners that you can easily add to your Azure subscription to increase the value of your investment in OMS.

As a partner, you can create your own solutions to support your applications and services and provide them to users through the Azure Marketplace or Quick Start Templates.

Performance alerting and monitoring

Alerting

Alerts are a method of monitoring Azure resource metrics, events, or logs and being notified when a condition you specify is met.

Alerts in different Azure services

Alerts are available across different services, including:

- Application Insights: Enables web test and metric alerts.

NOTE

See [Set alerts in Application Insights](#) and [Monitor availability and responsiveness of any website](#).

- Log Analytics (Operations Management Suite): Enables the routing of Activity and Diagnostic Logs to Log Analytics. Operations Management Suite allows metric, log, and other alert types.

NOTE

For more information, see Alerts in [Log Analytics](#).

- Azure Monitor: Enables alerts based on both metric values and activity log events. You can use the [Azure Monitor REST API](#) to manage alerts.

NOTE

For more information, see [Using the Azure portal, PowerShell, or the command-line interface to create alerts](#).

Monitoring

Performance issues in your cloud app can impact your business. With multiple interconnected components and frequent releases, degradations can happen at any time. And if you're developing an app, your users usually discover issues that you didn't find in testing. You should know about these issues immediately, and have tools for diagnosing and fixing the problems. Microsoft Azure has a range of tools for identifying these problems.

How do I monitor my Azure cloud apps?

There is a range of tools for monitoring Azure applications and services. Some of their features overlap. This is partly for historical reasons and partly due to the blurring between development and operation of an application.

Here are the principal tools:

- **Azure Monitor** is basic tool for monitoring services running on Azure. It gives you infrastructure-level data about the throughput of a service and the surrounding environment. If you are managing your apps all in Azure, deciding whether to scale up or down resources, then Azure Monitor gives you what you use to start.
- **Application Insights** can be used for development and as a production monitoring solution. It works by installing a package into your app, and so gives you a more internal view of what's going on. Its data includes response times of dependencies, exception traces, debugging snapshots, execution profiles. It provides powerful smart tools for analyzing all this telemetry both to help you debug an app and to help you understand what users are doing with it. You can tell whether a spike in response times is due to something in an app, or some external resourcing issue. If you use Visual Studio and the app is at fault, you can be taken right to the problem line(s) of code so you can fix it.
- **Log Analytics** is for those who need to tune performance and plan maintenance on applications running in production. It is based in Azure. It collects and aggregates data from many sources, though with a delay of 10 to 15 minutes. It provides a holistic IT management solution for Azure, on-premises, and third-party cloud-based infrastructure (such as Amazon Web Services). It provides richer tools to analyze data across more sources, allows complex queries across all logs, and can proactively alert on specified conditions. You can even collect custom data into its central repository so you can query and visualize it.
- **System Center Operations Manager (SCOM)** is for managing and monitoring large cloud installations.

You might be already familiar with it as a management tool for on-premises Windows Sever and Hyper-V based-clouds, but it can also integrate with and manage Azure apps. Among other things, it can install Application Insights on existing live apps. If an app goes down, it tells you in seconds.

Next steps

- [Best practices for creating Azure Resource Manager templates.](#)
- [Examples of implementing Azure subscription governance.](#)
- [Microsoft Azure Government.](#)

Azure Data Encryption-at-Rest

9/6/2017 • 19 min to read • [Edit Online](#)

There are multiple tools within Microsoft Azure to safeguard data according to your company's security and compliance needs. This paper focuses on how data is protected at rest across Microsoft Azure, discusses the various components taking part in the data protection implementation, and reviews pros and cons of the different key management protection approaches.

Encryption at Rest is a common security requirement. A benefit of Microsoft Azure is that organizations can achieve Encryption at Rest without having the cost of implementation and management and the risk of a custom key management solution. Organizations have the option of letting Azure completely manage Encryption at Rest. Additionally, organizations have various options to closely manage encryption or encryption keys.

What is encryption at rest?

Encryption at Rest refers to the cryptographic encoding (encryption) of data when it is persisted. The Encryption at Rest designs in Azure use symmetric encryption to encrypt and decrypt large amounts of data quickly according to a simple conceptual model:

- A symmetric encryption key is used to encrypt data as it is persisted
- The same encryption key is used to decrypt that data as it is readied for use in memory
- Data may be partitioned, and different keys may be used for each partition
- Keys must be stored in a secure location with access control policies limiting access to certain identities and logging key usage. Data encryption keys are often encrypted with asymmetric encryption to further limit access (discussed in the *Key Hierarchy*, later in this article)

The above describes the common high-level elements of Encryption at Rest. In practice, key management and control scenarios, as well as scale and availability assurances, require additional constructs. Microsoft Azure Encryption at Rest concepts and components are described below.

The purpose of encryption at rest

Encryption at rest is intended to provide data protection for data at-rest (as described above.) Attacks against data at-rest include attempts to obtain physical access to the hardware on which the data is stored, and then compromise the contained data. In such an attack, a server's hard drive may have been mishandled during maintenance allowing an attacker to remove the hard drive. Later the attacker would put the hard drive into a computer under their control to attempt to access the data.

Encryption at rest is designed to prevent the attacker from accessing the unencrypted data by ensuring the data is encrypted when on disk. If an attacker were to obtain a hard drive with such encrypted data, and no access to the encryption keys, the attacker would not compromise the data without great difficulty. In such a scenario, an attacker would have to attempt attacks against encrypted data which are much more complex and resource consuming than accessing unencrypted data on a hard drive. For this reason, encryption at rest is highly recommended and is a high priority requirement for many organizations.

In some cases, encryption at rest is also required by an organization's need for data governance and compliance efforts. Industry and government regulations such as HIPAA, PCI and FedRAMP, and international regulatory requirements, lay out specific safeguards through processes and policies regarding data protection and encryption requirements. For many of those regulations encryption at rest is a mandatory measure required for compliant data management and protection.

In addition to compliance and regulatory requirements, encryption at rest should be perceived as a defense-in-depth platform capability. While Microsoft provides a compliant platform for services, applications, and data, comprehensive facility and physical security, data access control and auditing, it is important to provide additional “overlapping” security measures in case one of the other security measures fails. Encryption at rest provides such an additional defense mechanism.

Microsoft is committed to providing encryption at rest options across cloud services and to provide customers suitable manageability of encryption keys and access to logs showing when encryption keys are used. Additionally, Microsoft is working towards the goal of making all customer data encrypted at rest by default.

Azure Encryption at Rest Components

As described previously, the goal of encryption at rest is that data that is persisted on disk is encrypted with a secret encryption key. To achieve that goal secure key creation, storage, access control and management of the encryption keys must be provided. Though details may vary, Azure services Encryption at Rest implementations can be described in terms of the below concepts which are then illustrated in the following diagram.



Azure Key Vault

The storage location of the encryption keys and access control to those keys is central to an encryption at rest model. The keys need to be highly secured but manageable by specified users and available to specific services. For Azure services, Azure Key Vault is the recommended key storage solution and provides a common management experience across services. Keys are stored and managed in key vaults, and access to a key vault can be given to users or services. Azure Key Vault supports customer creation of keys or import of customer keys for use in customer-managed encryption key scenarios.

Azure Active Directory

Permissions to use the keys stored in Azure Key Vault, either to manage or to access them for Encryption at Rest encryption and decryption, can be given to Azure Active Directory accounts.

Key Hierarchy

Usually more than one encryption key is used in an encryption at rest implementation. Asymmetric encryption is useful for establishing the trust and authentication needed for key access and management. Symmetric encryption is more efficient for bulk encryption and decryption, allowing for stronger encryption and better performance. Additionally, limiting the use of a single encryption key decreases the risk that the key will be compromised and the cost of re-encryption when a key must be replaced. To leverage the benefits of asymmetric and symmetric encryption and limit the use and exposure of a single key, Azure encryption at rest models use a key hierarchy made up of the following types of keys:

- **Data Encryption Key (DEK)** – A symmetric AES256 key used to encrypt a partition or block of data. A single resource may have many partitions and many Data Encryption Keys. Encrypting each block of data with a

different key makes crypto analysis attacks more difficult. Access to DEKs is needed by the resource provider or application instance that is encrypting and decrypting a specific block. When a DEK is replaced with a new key only the data in its associated block must be re-encrypted with the new key.

- **Key Encryption Key (KEK)** – An asymmetric encryption key used to encrypt the Data Encryption Keys. Use of a Key Encryption Key allows the data encryption keys themselves to be encrypted and controlled. The entity that has access to the KEK may be different than the entity that requires the DEK. This allows an entity to broker access to the DEK for the purpose of ensuring limited access of each DEK to specific partition. Since the KEK is required to decrypt the DEKs, the KEK is effectively a single point by which DEKs can be effectively deleted by deletion of the KEK.

The Data Encryption Keys, encrypted with the Key Encryption Keys are stored separately and only an entity with access to the Key Encryption Key can get any Data Encryption Keys encrypted with that key. Different models of key storage are supported. We will discuss each model in more detail later in the next section.

Data Encryption Models

Understanding the various encryption models, and their pros and cons is essential for understanding how the various resource providers in Azure implement encryption at Rest. These definitions are shared across all resource providers in Azure to ensure common language and taxonomy.

There are three scenarios for server-side encryption:

- Server-side encryption using Service Managed keys
 - Azure Resource Providers perform the encryption and decryption operations
 - Microsoft manages the keys
 - Full cloud functionality
- Server-side encryption using customer-managed keys in Azure Key Vault
 - Azure Resource Providers perform the encryption and decryption operations
 - Customer controls keys via Azure Key Vault
 - Full cloud functionality
- Server-side encryption using customer-managed keys on customer controlled hardware
 - Azure Resource Providers perform the encryption and decryption operations
 - Customer controls keys on customer controlled hardware
 - Full cloud functionality

For client-side encryption, consider the following:

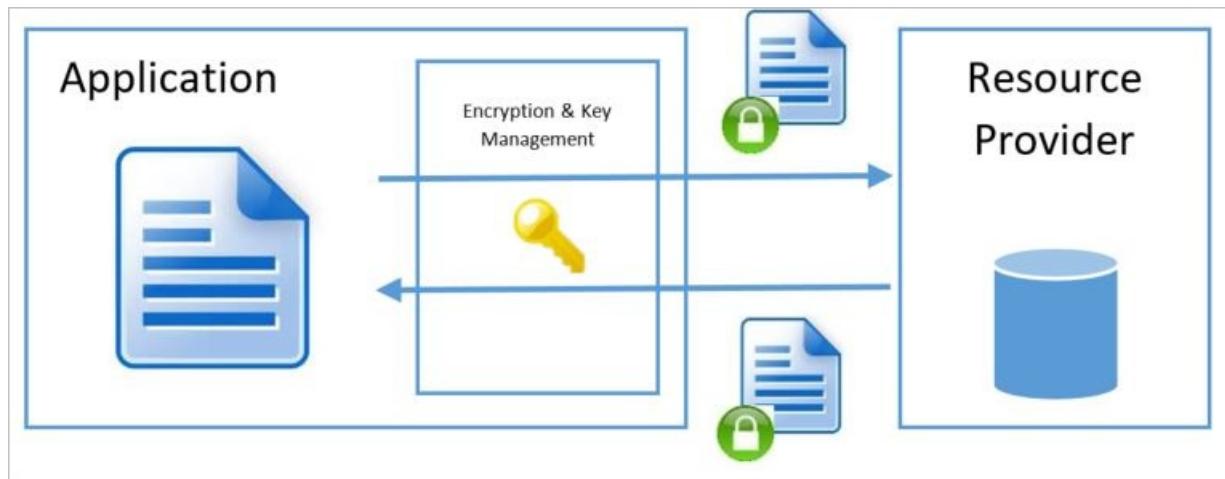
- Azure services cannot see decrypted data
- Customers manage and store keys on-premises (or in other secure stores). Keys are not available to Azure services
- Reduced cloud functionality

The supported encryption models in Azure split into two main groups: "Client Encryption" and "Server-side Encryption" as mentioned previously. Note that, independent of the encryption at rest model used, Azure services always recommend the use of a secure transport such as TLS or HTTPS. Therefore, encryption in transport should be addressed by the transport protocol and should not be a major factor in determining which encryption at rest model to use.

Client encryption model

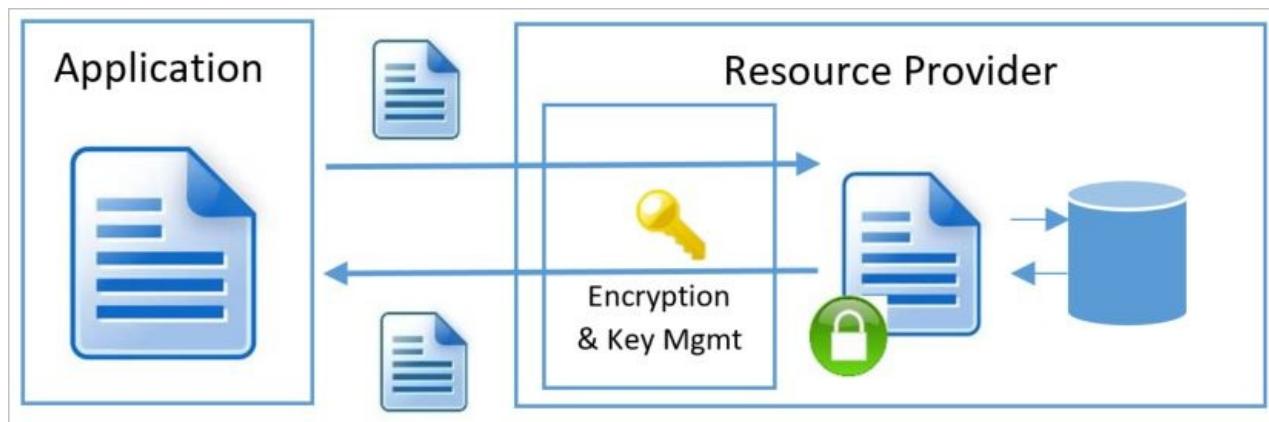
Client Encryption model refers to encryption that is performed outside of the Resource Provider or Azure by the service or calling application. The encryption can be performed by the service application in Azure, or by an application running in the customer data center. In either case, when leveraging this encryption model, the Azure

Resource Provider receives an encrypted blob of data without the ability to decrypt the data in any way or have access to the encryption keys. In this model, the key management is done by the calling service/application and is completely opaque to the Azure service.



Server-side encryption model

Server-side Encryption models refer to encryption that is performed by the Azure service. In that model, the Resource Provider performs the encrypt and decrypt operations. For example, Azure Storage may receive data in plain text operations and will perform the encryption and decryption internally. The Resource Provider might use encryption keys that are managed by Microsoft or by the customer depending on the provided configuration.

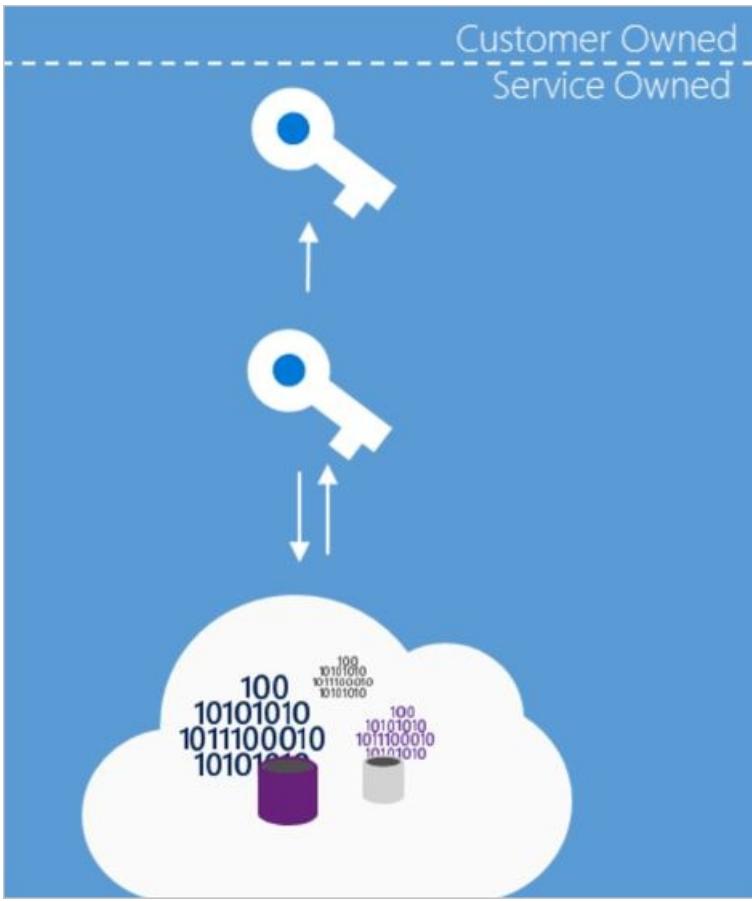


Server-side encryption key management models

Each of the server-side encryption at rest models implies distinctive characteristics of key management. This includes where and how encryption keys are created, and stored as well as the access models and the key rotation procedures.

Server-side encryption using service managed keys

For many customers, the essential requirement is to ensure that the data is encrypted whenever it is at rest. Server-side encryption using Service Managed Keys enables this model by allowing customers to mark the specific resource (Storage Account, SQL DB, etc.) for encryption and leaving all key management aspects such as key issuance, rotation and backup to Microsoft. Most Azure Services that support encryption at rest typically support this model of offloading the management of the encryption keys to Azure. The Azure resource provider creates the keys, places them in secure storage, and retrieves them when needed. This means that the service has full access to the keys and the service has full control over the credential lifecycle management.



Server-side encryption using service managed keys therefore quickly addresses the need to have encryption at rest with low overhead to the customer. When available a customer typically opens the Azure portal for the target subscription and resource provider and checks a box indicating they would like the data to be encrypted. In some Resource Managers server-side encryption with service managed keys is on by default.

Server-side encryption with Microsoft managed keys does imply the service has full access to store and manage the keys. While some customers may want to manage the keys because they feel they can ensure greater security, the cost and risk associated with a custom key storage solution should be considered when evaluating this model. In many cases an organization may determine that resource constraints or risks of an on-premises solution may greater than the risk of cloud management of the encryption at rest keys. However, this model might not be sufficient for organizations that have requirements to control the creation or lifecycle of the encryption keys or to have different personnel manage a service's encryption keys than those managing the service (i.e., segregation of key management from the overall management model for the service).

Key access

When Server-side encryption with Service Managed keys is used, the key creation, storage and service access are all managed by the service. Typically, the foundational Azure resource providers will store the Data Encryption Keys in a store that is close to the data and quickly available and accessible while the Key Encryption Keys are stored in a secure internal store.

Advantages

- Simple setup
- Microsoft manages key rotation, backup and redundancy
- Customer does not have the cost associated with implementation or the risk of a custom key management scheme.

Disadvantages

- No customer control over the encryption keys (key specification, lifecycle, revocation, etc.)
- No ability to segregate key management from overall management model for the service

Server-side encryption using customer managed keys in Azure Key Vault

For scenarios where the requirement is to encrypt the data at rest and control the encryption keys customers can use server-side Encryption using Customer Managed Keys in Key Vault. Some services may store only the root Key Encryption Key in Azure Key Vault and store the encrypted Data Encryption Key in an internal location closer to the data. In that scenario customers can bring their own keys to Key Vault (BYOK – Bring Your Own Key), or generate new ones, and use them to encrypt the desired resources. While the Resource Provider performs the encryption and decryption operations it uses the configured key as the root key for all encryption operations.

Key Access

The server-side encryption model with customer managed keys in Azure Key Vault involves the service accessing the keys to encrypt and decrypt as needed. Encryption at rest keys are made accessible to a service through an access control policy granting that service identity access to receive the key. An Azure service running on behalf of an associated subscription can be configured with an identity for that service within that subscription. The service can perform Azure Active Directory authentication and receive an authentication token identifying itself as that service acting on behalf of the subscription. That token can then be presented to Key Vault to obtain a key it has been given access to.

For operations using encryption keys, a service identity can be granted access to any of the following operations: decrypt, encrypt, unwrapKey, wrapKey, verify, sign, get, list, update, create, import, delete, backup, and restore.

To obtain a key for use in encrypting or decrypting data at rest the service identity that the Resource Manager service instance will run as must have UnwrapKey (to get the key for decryption) and WrapKey (to insert a key into key vault when creating a new key).

NOTE

For more detail on Key Vault authorization see the secure your key vault page in the [Azure Key Vault documentation](#).

Advantages

- Full control over the keys used – encryption keys are managed in the customer's Key Vault under the customer's control.
- Ability to encrypt multiple services to one master
- Can segregate key management from overall management model for the service
- Can define service and key location across regions

Disadvantages

- Customer has full responsibility for key access management
- Customer has full responsibility for key lifecycle management
- Additional Setup & configuration overhead

Server-side encryption using service managed keys in customer controlled hardware

For scenarios where the requirement is to encrypt the data at rest and manage the keys in a proprietary repository outside of Microsoft's control, some Azure services enable the Host Your Own Key (HYOK) key management model. In this model, the service must retrieve the key from an external site and therefore performance and availability guarantees are impacted, and configuration is more complex. Additionally, since the service does have access to the DEK during the encryption and decryption operations the overall security guarantees of this model are similar to when the keys are customer managed in Azure Key Vault. As a result, this model is not appropriate for most organizations unless they have specific key management requirements necessitating it. Due to these limitations, most Azure Services do not support server-side encryption using server-managed keys in customer controlled hardware.

Key Access

When server-side encryption using service managed keys in customer controlled hardware is used the keys are

maintained on a system configured by the customer. Azure services that support this model provide a means of establishing a secure connection to a customer supplied key store.

Advantages

- Full control over the root key used – encryption keys are managed by a customer provided store
- Ability to encrypt multiple services to one master
- Can segregate key management from overall management model for the service
- Can define service and key location across regions

Disadvantages

- Full responsibility for key storage, security, performance and availability
- Full responsibility for key access management
- Full responsibility for key lifecycle management
- Significant setup, configuration and ongoing maintenance costs
- Increased dependency on network availability between the customer datacenter and Azure datacenters.

Encryption at rest in Microsoft cloud services

Microsoft Cloud services are used in all three cloud models: IaaS, PaaS, SaaS. Below you have examples of how they fit on each model:

- Software services, referred to as Software as a Server or SaaS, which have application provided by the cloud such as Office 365.
- Platform services which customers leverage the cloud in their applications, using the cloud for things like storage, analytics and service bus functionality.
- Infrastructure services, or Infrastructure as a Service (IaaS) in which customer deploy operating systems and applications that are hosted in the cloud and possibly leveraging other cloud services.

Encryption at rest for SaaS customers

Software as a Service (SaaS) customers typically have encryption at rest enabled or available in each service. Office 365 services has several options for customers to verify or enable encryption at rest. For information about Office 365 services see Data Encryption Technologies for Office 365.

Encryption at rest for PaaS customers

Platform as a Service (PaaS) customer's data typically resides in an application execution environment and any Azure Resource Providers used to store customer data. To see the encryption at rest options available to you examine the table below for the storage and application platforms that you use. Where supported, links to instructions on enabling Encryption at Rest are provided for each resource provider.

Encryption at rest for IaaS customers

Infrastructure as a Service (IaaS) customers can have a variety of services and applications in use. IaaS services can enable encryption at rest in their Azure hosted virtual machines and VHDs using Azure Disk Encryption.

Encrypted storage

Like PaaS, IaaS solutions can leverage other Azure services that store data encrypted at rest. In these cases, you can enable the Encryption at Rest support as provided by each consumed Azure service. The below table enumerates the major storage, services and application platforms and the model of Encryption at Rest supported. Where supported, links are provided to instructions on enabling Encryption at Rest.

Encrypted compute

A complete Encryption at Rest solution requires that the data is never persisted in unencrypted form. While in use, on a server loading the data in memory, data can be persisted locally in various ways including the Windows page file, a crash dump, and any logging the application may perform. To ensure this data is encrypted at rest IaaS

applications can use Azure Disk Encryption on an Azure IaaS virtual machine (Windows or Linux) and virtual disk.

Custom encryption at rest

It is recommended that whenever possible, IaaS applications leverage Azure Disk Encryption and Encryption at Rest options provided by any consumed Azure services. In some cases, such as irregular encryption requirements or non-Azure based storage, a developer of an IaaS application may need to implement encryption at rest themselves. Developers of IaaS solutions can better integrate with Azure management and customer expectations by leveraging certain Azure components. Specifically, developers should use the Azure Key Vault service to provide secure key storage as well as provide their customers with consistent key management options with that of most Azure platform services. Additionally, custom solutions should use Azure Managed Service Identities to enable service accounts to access encryption keys. For developer information on Azure Key Vault and Managed Service Identities see their respective SDKs.

Azure resource providers encryption model support

Microsoft Azure Services each support one or more of the encryption at rest models. For some services, however, one or more of the encryption models may not be applicable. Additionally, services may release support for these scenarios at different schedules. This section describes the encryption at rest support at the time of this writing for each of the major Azure data storage services.

Azure disk encryption

Any customer using Azure Infrastructure as a Service (IaaS) features can achieve encryption at rest for their IaaS VMs and disks through Azure Disk Encryption. For more information on Azure Disk encryption see the [Azure Disk Encryption documentation](#).

Azure storage

Azure Blob, and File supports encryption at rest for server-side encrypted scenarios as well as customer encrypted data (client-side encryption).

- Server-side: customers using Azure blob storage can enable encryption at rest on each Azure storage resource account. Once enabled server-side encryption is done transparently to the application. See [Azure Storage Service Encryption for Data at Rest](#) for more information.
- Client-side: client-side encryption of Azure Blobs is supported. When using client-side encryption customers encrypt the data and upload the data as an encrypted blob. Key management is done by the customer. See [Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage](#) for more information.

SQL Azure

SQL Azure currently supports encryption at rest for Microsoft managed service side and client-side encryption scenarios.

Support for sever encryption is currently provided through the SQL feature called Transparent Data Encryption. Once a SQL Azure customer enables TDE key are automatically created and managed for them. Encryption at rest can be enabled at the database and server levels. As of June 2017, [Transparent Data Encryption \(TDE\)](#) will be enabled by default on newly created databases.

Client-side encryption of SQL Azure data is supported through the [Always Encrypted](#) feature. Always Encrypted uses a key that created and stored by the client. Customers can store the master key in a Windows certificate store, Azure Key Vault, or a local Hardware Security Module. Using SQL Server Management Studio, SQL users choose what key they'd like to use to encrypt which column.

			ENCRYPTION MODEL		
					Client

			ENCRYPTION MODEL		
	Key Management	Service Managed Key	Customer Managed in Key Vault	Customer Managed On-premises	
Storage and Databases					
Disk (IaaS)		-	Yes	Yes*	-
SQL Server (IaaS)		Yes	Yes	Yes	Yes
SQL Azure (PaaS)		Yes	Preview	-	Yes
Azure Storage (Block/Page Blobs)		Yes	Preview	-	Yes
Azure Storage (Files)		Yes	-	-	-
Azure Storage (Tables, Queues)		-	-	-	Yes
Cosmos DB (Document DB)		Yes	-	-	-
StorSimple		Yes	-	-	Yes
Backup		-	-	-	Yes
Intelligence and Analytics					
Azure Data Factory		Yes	-	-	-
Azure Machine Learning		-	Preview	-	-
Azure Stream Analytics		Yes	-	-	-
HDInsights (Azure Blob Storage)		Yes	-	-	-
HDInsights (Data Lake Storage)		Yes	-	-	-
Azure Data Lake Store		Yes	Yes	-	-

			ENCRYPTION MODEL		
Azure Data Catalog		Yes	-	-	-
Power BI		Yes	-	-	-
IoT Services					
IoT Hub		-	-	-	Yes
Service Bus		-	-	-	Yes
Event Hubs		-	-	-	-

Conclusion

Protection of customer data stored within Azure Services is of paramount importance to Microsoft. All Azure hosted services are committed to providing Encryption at Rest options. Foundational services such as Azure Storage, SQL Azure and key analytics and intelligence services already provide Encryption at Rest options. Some of these services support either customer controlled keys and client-side encryption as well as service managed keys and encryption. Microsoft Azure services are broadly enhancing Encryption at Rest availability and new options are planned for preview and general availability in the upcoming months.

Getting started with Microsoft Azure security

6/27/2017 • 16 min to read • [Edit Online](#)

When you build or migrate IT assets to a cloud provider, you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security needs. In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your deployments.

In this overview article on Azure security, we'll look at:

- Azure services and features you can use to help secure your services and data within Azure.
- How Microsoft secures the Azure infrastructure to help protect your data and applications.

Identity and access management

Controlling access to IT infrastructure, data, and applications is critical. Microsoft Azure delivers these capabilities by services such as Azure Active Directory (Azure AD), Azure Storage, and support for numerous standards and APIs.

[Azure AD](#) is an identity repository and engine that provides authentication, authorization, and access control for an organization's users, groups, and objects. Azure AD also offers developers an effective way to integrate identity management in their applications. Industry-standard protocols such as [SAML 2.0](#), [WS-Federation](#), and [OpenID Connect](#) make sign-in possible on platforms such as .NET, Java, Node.js, and PHP.

The REST-based Graph API enables developers to read and write to the directory from any platform. Through support for [OAuth 2.0](#), developers can build mobile and web applications that integrate with Microsoft and third-party web APIs, and build their own secure web APIs. Open-source client libraries are available for .Net, Windows Store, iOS, and Android, with additional libraries under development.

How Azure enables identity and access management

Azure AD can be used as a standalone cloud directory for your organization or as an integrated solution with your existing on-premises Active Directory. Some integration features include directory sync and single sign-on (SSO). These extend the reach of your existing on-premises identities into the cloud and improve the admin and user experience.

Some other capabilities for identity and access management include:

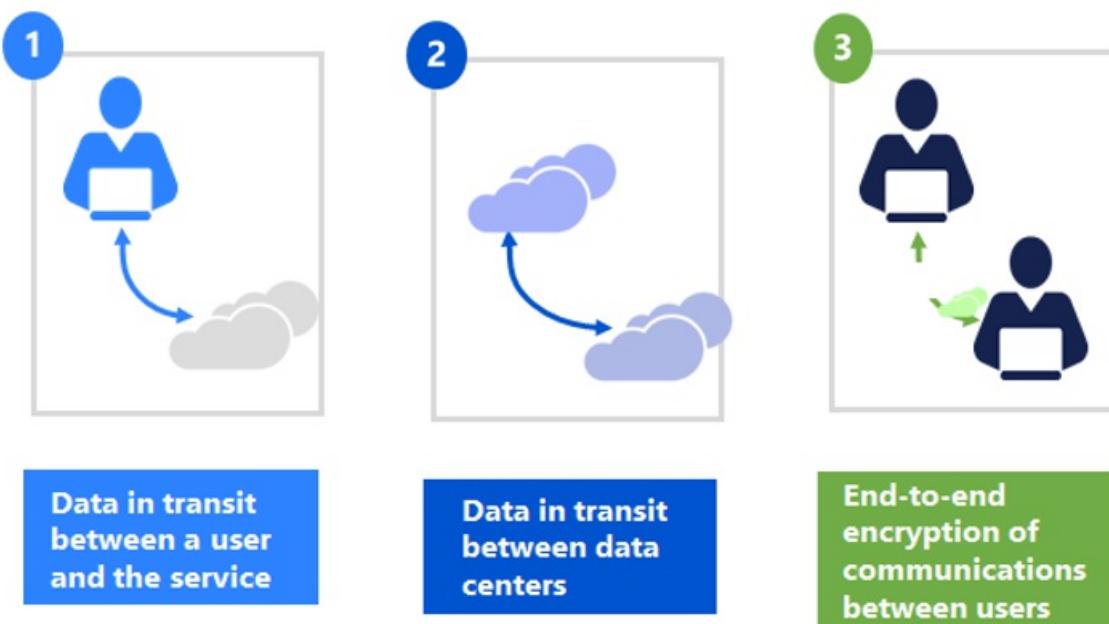
- Azure AD enables [SSO](#) to SaaS applications, regardless of where they are hosted. Some applications are federated with Azure AD, and others use password SSO. Federated applications can also support user provisioning and password vaulting.
- Access to data in [Azure Storage](#) is controlled via authentication. Each storage account has a primary key ([storage account key](#), or SAK) and a secondary secret key (the shared access signature, or SAS).
- Azure AD provides Identity as a Service through federation by using [Active Directory Federation Services](#), synchronization, and replication with on-premises directories.
- [Azure Multi-Factor Authentication](#) is the multi-factor authentication service that requires users to verify sign-ins by using a mobile app, phone call, or text message. It can be used with Azure AD to help secure on-premises resources with the Azure Multi-Factor Authentication server, and also with custom applications and directories using the SDK.

- [Azure AD Domain Services](#) lets you join Azure virtual machines to a domain without deploying domain controllers. You can sign in to these virtual machines with your corporate Active Directory credentials and administer domain-joined virtual machines by using Group Policy to enforce security baselines on all your Azure virtual machines.
- [Azure Active Directory B2C](#) provides a highly available global-identity management service for consumer-facing applications that scales to hundreds of millions of identities. It can be integrated across mobile and web platforms. Your consumers can sign in to all your applications through customizable experiences by using their existing social accounts or by creating new credentials.

Data access control and encryption

Microsoft employs the principles of Separation of Duties and [Least Privilege](#) throughout Azure operations. Access to data by Azure support personnel requires your explicit permission and is granted on a “just-in-time” basis that is logged and audited, then revoked after completion of the engagement.

Azure also provides multiple capabilities for protecting data in transit and at rest. This includes encryption for data, files, applications, services, communications, and drives. You can encrypt information before placing it in Azure, and also store keys in your on-premises datacenters.



Azure encryption technologies

You can gather details on administrative access to your subscription environment by using [Azure AD Reporting](#). You can configure [BitLocker Drive Encryption](#) on VHDs containing sensitive information in Azure.

Other capabilities in Azure that will assist you to keep your data secure include:

- Application developers can build encryption into the applications they deploy in Azure by using the Windows [CryptoAPI](#) and .NET Framework.
- Completely control the keys with client-side encryption for Azure Blob storage. The storage service never sees the keys and is incapable of decrypting the data.
- [Azure Rights Management \(Azure RMS\)](#) (with the [RMS SDK](#)) provides file and data-level encryption and data-leak prevention through policy-based access management.
- Azure supports [table-level and column-level encryption \(TDE/CLE\)](#) in SQL Server virtual machines, and it supports third-party on-premises key management servers in datacenters.
- Storage Account Keys, Shared Access Signatures, management certificates, and other keys are unique to each Azure tenant.
- Azure [StorSimple](#) hybrid storage encrypts data via a 128-bit public/private key pair before uploading it to Azure

Storage.

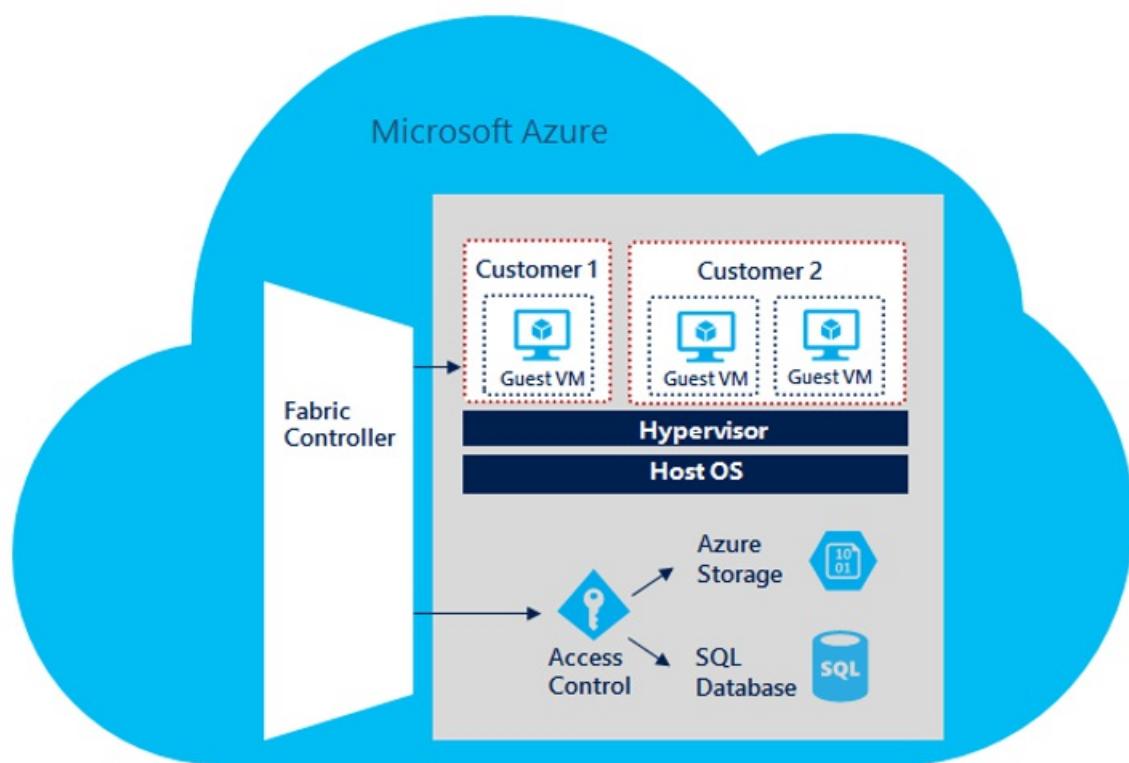
- Azure supports and uses numerous encryption mechanisms, including SSL/TLS, IPsec, and AES, depending on the data types, containers, and transports.

Virtualization

The Azure platform uses a virtualized environment. User instances operate as standalone virtual machines that do not have access to a physical host server, and this isolation is enforced by using [physical processor \(ring-0/ring-3\) privilege levels](#).

Ring 0 is the most privileged and 3 is the least. The guest OS runs in a lesser-privileged Ring 1, and applications run in the least privileged Ring 3. This virtualization of physical resources leads to a clear separation between guest OS and hypervisor, resulting in additional security separation between the two.

The Azure hypervisor acts like a micro-kernel and passes all hardware access requests from guest virtual machines to the host for processing by using a shared-memory interface called VMBus. This prevents users from obtaining raw read/write/execute access to the system and mitigates the risk of sharing system resources.



How Azure implements virtualization

Azure uses a hypervisor firewall (packet filter) that is implemented in the hypervisor and configured by a fabric controller agent. This helps protect tenants from unauthorized access. By default, all traffic is blocked when a virtual machine is created, and then the fabric controller agent configures the packet filter to add *rules and exceptions* to allow authorized traffic.

There are two categories of rules that are programmed here:

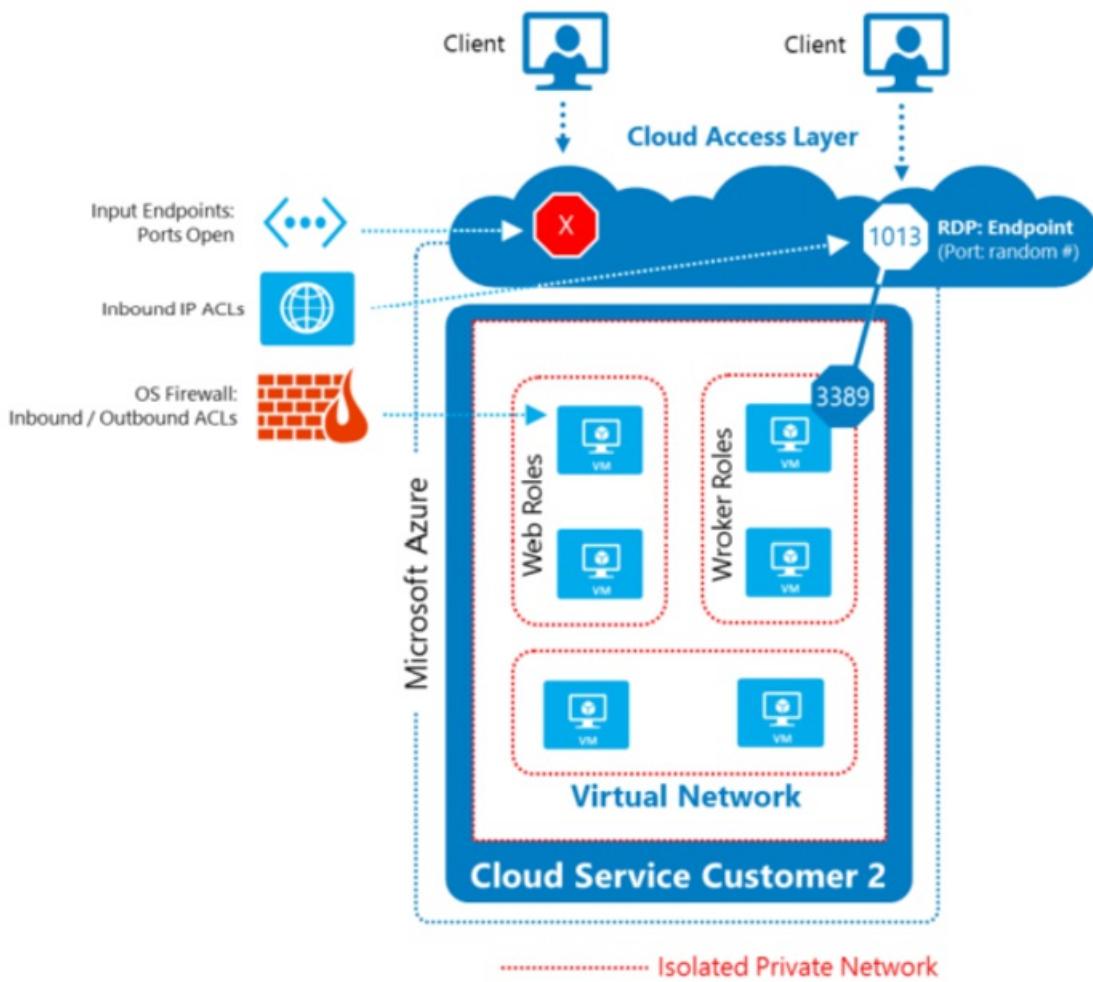
- **Machine configuration or infrastructure rules:** By default, all communication is blocked. There are exceptions to allow a virtual machine to send and receive DHCP and DNS traffic. Virtual machines can also send traffic to the “public” internet and send traffic to other virtual machines within the cluster and the OS activation server. The virtual machines’ list of allowed outgoing destinations does not include Azure router subnets, Azure management back end, and other Microsoft properties.
- **Role configuration file:** This defines the inbound Access Control Lists (ACLs) based on the tenant’s service model. For example, if a tenant has a Web front end on port 80 on a certain virtual machine, then Azure opens TCP port 80 to all IPs if you’re configuring an endpoint in the [Azure classic deployment model](#). If the virtual

machine has a back end or worker role running, then it opens the worker role only to the virtual machine within the same tenant.

Isolation

Another important cloud security requirement is separation to prevent unauthorized and unintentional transfer of information between deployments in a shared multi-tenant architecture.

Azure implements [network access control](#) and segregation through VLAN isolation, ACLs, load balancers, and IP filters. It restricts external traffic inbound to ports and protocols on your virtual machines that you define. Azure implements network filtering to prevent spoofed traffic and restrict incoming and outgoing traffic to trusted platform components. Traffic flow policies are implemented on boundary protection devices that deny traffic by default.



Network Address Translation (NAT) is used to separate internal network traffic from external traffic. Internal traffic is not externally routable. [Virtual IP addresses](#) that are externally routable are translated into [internal Dynamic IP](#) addresses that are only routable within Azure.

External traffic to Azure virtual machines is firewalled via ACLs on routers, load balancers, and Layer 3 switches. Only specific known protocols are permitted. ACLs are in place to limit traffic originating from guest virtual machines to other VLANs used for management. In addition, traffic filtered via IP filters on the host OS further limits the traffic on both data link and network layers.

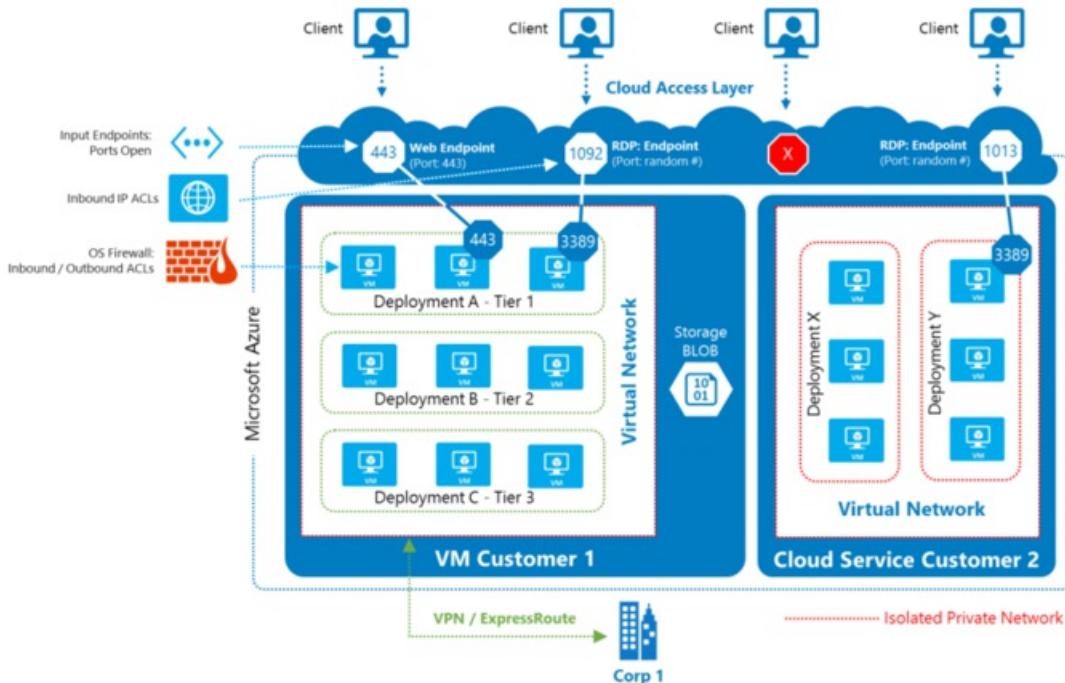
How Azure implements isolation

The Azure Fabric Controller is responsible for allocating infrastructure resources to tenant workloads, and it manages unidirectional communications from the host to virtual machines. The Azure hypervisor enforces memory and process separation between virtual machines, and it securely routes network traffic to guest OS tenants. Azure also implements isolation for tenants, storage, and virtual networks.

- Each Azure AD tenant is logically isolated by using security boundaries.
- Azure storage accounts are unique to each subscription, and access must be authenticated by using a storage account key.
- Virtual networks are logically isolated through a combination of unique private IP addresses, firewalls, and IP ACLs. Load balancers route traffic to the appropriate tenants based on endpoint definitions.

Virtual networks and firewalls

The [distributed and virtual networks](#) in Azure help ensure that your private network traffic is logically isolated from traffic on other Azure virtual networks.



Your subscription can contain multiple isolated private networks (and include firewall, load balancing, and network address translation).

Azure provides three primary levels of network segregation in each Azure cluster to logically segregate traffic. [Virtual local area networks](#) (VLANs) are used to separate customer traffic from the rest of the Azure network. Access to the Azure network from outside the cluster is restricted through load balancers.

Network traffic to and from virtual machines must pass through the hypervisor virtual switch. The IP filter component in the root OS isolates the root virtual machine from the guest virtual machines and the guest virtual machines from one another. It performs filtering of traffic to restrict communication between a tenant's nodes and the public Internet (based on the customer's service configuration), segregating them from other tenants.

The IP filter helps prevent guest virtual machines from:

- Generating spoofed traffic.
- Receiving traffic not addressed to them.
- Directing traffic to protected infrastructure endpoints.
- Sending or receiving inappropriate broadcast traffic.

You can place your virtual machines onto [Azure virtual networks](#). These virtual networks are similar to the networks you configure in on-premises environments, where they are typically associated with a virtual switch. Virtual machines connected to the same virtual network can communicate with one another without additional configuration. You can also configure different subnets within your virtual network.

You can use the following Azure Virtual Network technologies to help secure communications on your virtual

network:

- **Network Security Groups (NSGs).** You can use an NSG to control traffic to one or more virtual machine instances in your virtual network. An NSG contains access control rules that allow or deny traffic based on traffic direction, protocol, source address and port, and destination address and port.
- **User-defined routing.** You can control the routing of packets through a virtual appliance by creating user-defined routes that specify the next hop for packets flowing to a specific subnet to go to a virtual network security appliance.
- **IP forwarding.** A virtual network security appliance must be able to receive incoming traffic that is not addressed to itself. To allow a virtual machine to receive traffic addressed to other destinations, you enable IP forwarding for the virtual machine.
- **Forced tunneling.** Forced tunneling lets you redirect or "force" all Internet-bound traffic generated by your virtual machines in a virtual network back to your on-premises location via a site-to-site VPN tunnel for inspection and auditing
- **Endpoint ACLs.** You can control which machines are allowed inbound connections from the Internet to a virtual machine on your virtual network by defining endpoint ACLs.
- **Partner network security solutions.** There are a number of partner network security solutions that you can access from the Azure Marketplace.

How Azure implements virtual networks and firewalls

Azure implements packet-filtering firewalls on all host and guest virtual machines by default. Windows OS images from the Azure Marketplace also have Windows Firewall enabled by default. Load balancers at the perimeter of Azure public-facing networks control communications based on IP ACLs managed by customer administrators.

If Azure moves a customer's data as part of normal operations or during a disaster, it does so over private, encrypted communications channels. Other capabilities employed by Azure to use in virtual networks and firewalls are:

- **Native host firewall:** Azure Service Fabric and Azure Storage run on a native OS that has no hypervisor. Hence the windows firewall is configured with the previous two sets of rules. Storage runs native to optimize performance.
- **Host firewall:** The host firewall is to protect the host operating system that runs the hypervisor. The rules are programmed to allow only the Service Fabric controller and jump boxes to talk to the host OS on a specific port. The other exceptions are to allow DHCP response and DNS Replies. Azure uses a machine configuration file that has the template of firewall rules for the host OS. The host itself is protected from external attack by a Windows firewall configured to permit communication only from known, authenticated sources.
- **Guest firewall:** Replicates the rules in the virtual machine Switch packet filter but programmed in different software (such as the Windows Firewall piece of the guest OS). The guest virtual machine firewall can be configured to restrict communications to or from the guest virtual machine, even if the communication is permitted by configurations at the host IP Filter. For example, you may choose to use the guest virtual machine firewall to restrict communication between two of your VNets that have been configured to connect to one another.
- **Storage firewall (FW):** The firewall on the storage front end filters traffic to be only on ports 80/443 and other necessary utility ports. The firewall on the storage back end restricts communications to come only from storage front-end servers.
- **Virtual Network Gateway:** The [Azure Virtual Network Gateway](#) serves as the cross-premises gateway connecting your workloads in Azure Virtual Network to your on-premises sites. It is required to connect to on-premises sites through [IPsec site-to-site VPN tunnels](#), or through [ExpressRoute](#) circuits. For IPsec/IKE VPN tunnels, the gateways perform IKE handshakes and establish the IPsec S2S VPN tunnels between the virtual networks and on-premises sites. Virtual network gateways also terminate [point-to-site VPNs](#).

Secure remote access

Data stored in the cloud must have sufficient safeguards enabled to prevent exploits and maintain confidentiality and integrity while in-transit. This includes network controls that tie in with an organization's policy-based, auditable identity and access management mechanisms.

Built-in cryptographic technology enables you to encrypt communications within and between deployments, between Azure regions, and from Azure to on-premises datacenters. Administrator access to virtual machines through [remote desktop sessions](#), [remote Windows PowerShell](#), and the Azure portal is always encrypted.

To securely extend your on-premises datacenter to the cloud, Azure provides both [site-to-site VPN](#) and [point-to-site VPN](#), plus dedicated links with [ExpressRoute](#) (connections to Azure Virtual Networks over VPN are encrypted).

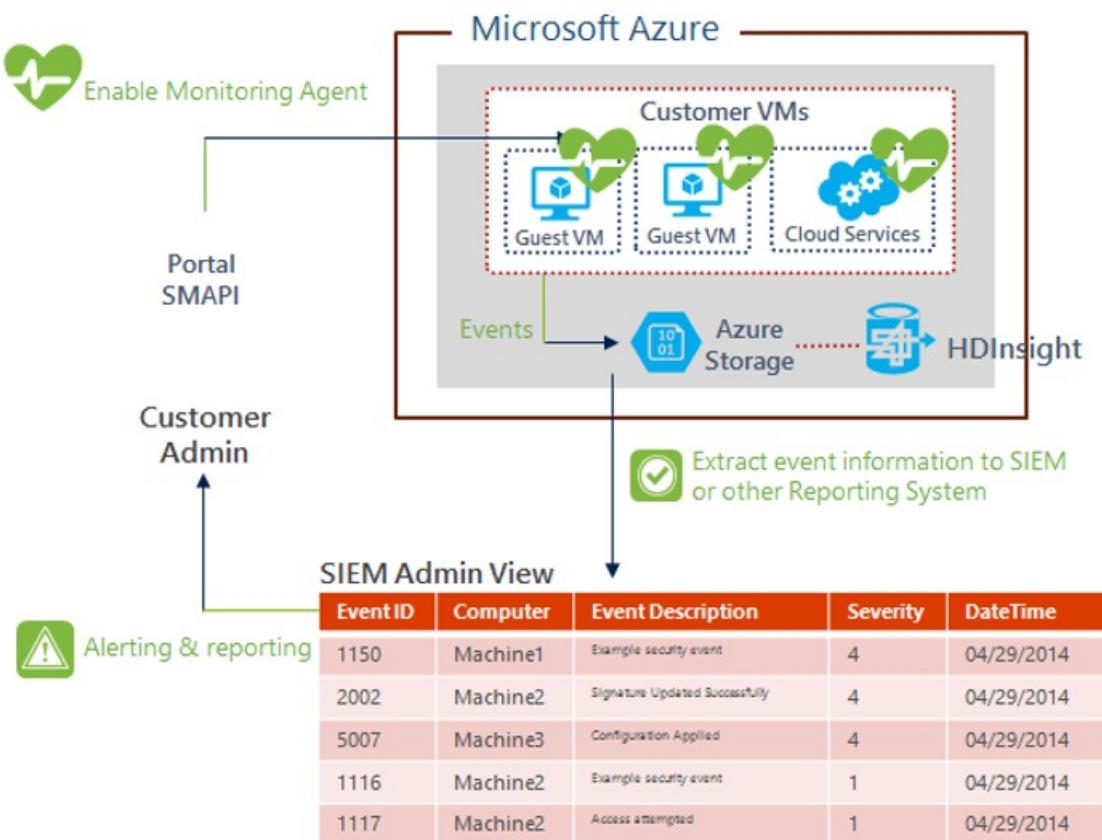
How Azure implements secure remote access

Connections to the Azure portal must always be authenticated, and they require SSL/TLS. You can configure management certificates to enable secure management. Industry-standard security protocols such as [SSTP](#) and [IPsec](#) are fully supported.

[Azure ExpressRoute](#) lets you create private connections between Azure datacenters and infrastructure that's on your premises or in a co-location environment. ExpressRoute connections do not go over the public Internet. They offer more reliability, faster speeds, lower latencies, and higher security than typical Internet-based links. In some cases, transferring data between on-premises locations and Azure by using ExpressRoute connections can also yield significant cost benefits.

Logging and monitoring

Azure provides authenticated logging of security-relevant events that generate an audit trail, and it is engineered to be resistant to tampering. This includes system information, such as security event logs in Azure infrastructure virtual machines and Azure AD. Security event monitoring includes collecting events such as changes in DHCP or DNS server IP addresses; attempted access to ports, protocols, or IP addresses that are blocked by design; changes in security policy or firewall settings; account or group creation; and unexpected processes or driver installation.



Audit logs recording privileged user access and activities, authorized and unauthorized access attempts, system exceptions, and information security events are retained for a set period of time. The retention of your logs is at

your discretion because you configure log collection and retention to your own requirements.

How Azure implements logging and monitoring

Azure deploys Management Agents (MA) and Azure Security Monitor (ASM) agents to each compute, storage, or fabric node under management whether they are native or virtual. Each Management Agent is configured to authenticate to a service team storage account with a certificate obtained from the Azure certificate store and forward pre-configured diagnostic and event data to the storage account. These agents are not deployed to customers' virtual machines.

Azure administrators access logs through a web portal for authenticated and controlled access to the logs. An administrator can parse, filter, correlate, and analyze logs. The Azure service team storage accounts for logs are protected from direct administrator access to help prevent against log tampering.

Microsoft collects logs from network devices using the Syslog protocol, and from host servers using Microsoft Audit Collection Services (ACS). These logs are placed into a log database from which alerts for suspicious events are generated. The administrator can access and analyze these logs.

[Azure Diagnostics](#) is a feature of Azure that enables you to collect diagnostic data from an application running in Azure. This is diagnostic data for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis, capacity planning, and auditing. After the diagnostic data is collected, it can be transferred to an Azure storage account for persistence. Transfers can either be scheduled or on demand.

Threat mitigation

In addition to isolation, encryption, and filtering, Azure employs a number of threat mitigation mechanisms and processes to protect infrastructure and services. These include internal controls and technologies used to detect and remediate advanced threats such as DDoS, privilege escalation, and the [OWASP Top-10](#).

The security controls and risk management processes Microsoft has in place to secure its cloud infrastructure reduce the risk of security incidents. In the event an incident occurs, the Security Incident Management (SIM) team within the Microsoft Online Security Services and Compliance (OSSC) team is ready to respond at any time.

How Azure implements threat mitigation

Azure has security controls in place to implement threat mitigation and also to help customers mitigate potential threats in their environments. The following list summarizes the threat mitigation capabilities offered by Azure:

- [Azure Antimalware](#) is enabled by default on all infrastructure servers. You can optionally enable it within your own virtual machines.
- Microsoft maintains continuous monitoring across servers, networks, and applications to detect threats and prevent exploits. Automated alerts notify administrators of anomalous behaviors, allowing them to take corrective action on both internal and external threats.
- You can deploy third-party security solutions within your subscriptions, such as web application firewalls from [Barracuda](#).
- Microsoft's approach to penetration testing includes "[Red-Teaming](#)," which involves Microsoft security professionals attacking (non-customer) live production systems in Azure to test defenses against real-world, advanced, persistent threats.
- Integrated deployment systems manage the distribution and installation of security patches across the Azure platform.

Next steps

[Azure Trust Center](#)

[Azure Security Team Blog](#)

Microsoft Security Response Center

Active Directory Blog

Azure security best practices and patterns

6/27/2017 • 1 min to read • [Edit Online](#)

We currently have the following Azure security best practices and patterns articles. Make sure to visit this site periodically to see updates to our growing list of Azure security best practices and patterns:

- [Azure network security best practices](#)
- [Azure data security and encryption best practices](#)
- [Identity management and access control security best practices](#)
- [Internet of Things security best practices](#)
- [Azure IaaS Security Best Practices](#)
- [Azure boundary security best practices](#)
- [Implementing a secure hybrid network architecture in Azure](#)
- [Azure PaaS Best Practices](#)

Azure provides a secure platform on which you can build your solutions. We also provide services and technologies to make your solutions on Azure more secure. Because of the many options available to you, many of you have voiced an interest in what Microsoft recommends as best practices and patterns for improving security.

We understand your interest and have created a collection of documents that describe things you can do, given the right context, to improve the security of Azure deployments.

In these best practices and patterns articles, we discuss a collection of best practices and useful patterns for specific topics. These best practices and patterns are derived from our experiences with these technologies and the experiences of customers like yourself.

For each best practice we strive to explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

We look forward to including many more articles on Azure security architecture and best practices. If there are topics that you'd like us to include, let us know in the discussion area at the bottom of this page.

Azure Security Services and Technologies

8/21/2017 • 1 min to read • [Edit Online](#)

In our discussions with current and future Azure customers, we're often asked "do you have a list of all the security related services and technologies that Azure has to offer?"

We understand that when you're evaluating your cloud service provider technical options, it's helpful to have such a list available that you can use to dig down deeper when the time is right for you.

The following is our initial effort at providing a list. Over time, this list will change and grow, just as Azure does. The list is categorized, and the list of categories will also grow over time. Make sure to check this page on a regular basis to stay up-to-date on our security-related services and technologies.

Azure Security - General

- [Azure Security Center](#)
- [Azure Key Vault](#)
- [Azure Disk Encryption](#)
- [Log Analytics](#)
- [Azure Dev/Test Labs](#)

Azure Storage Security

- [Azure Storage Service Encryption](#)
- [StorSimple Encrypted Hybrid Storage](#)
- [Azure Client-Side Encryption](#)
- [Azure Storage Shared Access Signatures](#)
- [Azure Storage Account Keys](#)
- [Azure File shares with SMB 3.0 Encryption](#)
- [Azure Storage Analytics](#)

Azure Database Security

- [Azure SQL Firewall](#)
- [Azure SQL Cell Level Encryption](#)
- [Azure SQL Connection Encryption](#)
- [Azure SQL Authentication](#)
- [Azure SQL Always Encryption](#)
- [Azure SQL Column Level Encryption](#)
- [Azure SQL Transparent Data Encryption](#)
- [Azure SQL Database Auditing](#)

Azure Identity and Access Management

- [Azure Role Based Access Control](#)
- [Azure Active Directory](#)
- [Azure Active Directory B2C](#)
- [Azure Active Directory Domain Services](#)

- [Azure Multi-Factor Authentication](#)

Backup and Disaster Recovery

- [Azure Backup](#)
- [Azure Site Recovery](#)

Azure Networking

- [Network Security Groups](#)
- [Azure VPN Gateway](#)
- [Azure Application Gateway](#)
- [Azure Load Balancer](#)
- [Azure ExpressRoute](#)
- [Azure Traffic Manager](#)
- [Azure Application Proxy](#)

Azure Network Security Best Practices

6/27/2017 • 17 min to read • [Edit Online](#)

Microsoft Azure enables you to connect virtual machines and appliances to other networked devices by placing them on Azure Virtual Networks. An Azure Virtual Network is a virtual network construct that allows you to connect virtual network interface cards to a virtual network to allow TCP/IP-based communications between network enabled devices. Azure Virtual Machines connected to an Azure Virtual Network are able to connect to devices on the same Azure Virtual Network, different Azure Virtual Networks, on the Internet or even on your own on-premises networks.

In this article we will discuss a collection of Azure network security best practices. These best practices are derived from our experience with Azure networking and the experiences of customers like yourself.

For each best practice, we'll explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

This Azure Network Security Best Practices article is based on a consensus opinion, and Azure platform capabilities and feature sets, as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure Network security best practices discussed in this article include:

- Logically segment subnets
- Control routing behavior
- Enable Forced Tunneling
- Use Virtual network appliances
- Deploy DMZs for security zoning
- Avoid exposure to the Internet with dedicated WAN links
- Optimize uptime and performance
- Use global load balancing
- Disable RDP Access to Azure Virtual Machines
- Enable Azure Security Center
- Extend your datacenter into Azure

Logically segment subnets

[Azure Virtual Networks](#) are similar to a LAN on your on-premises network. The idea behind an Azure Virtual Network is that you create a single private IP address space-based network on which you can place all your [Azure Virtual Machines](#). The private IP address spaces available are in the Class A (10.0.0.0/8), Class B (172.16.0.0/12) and Class C (192.168.0.0/16) ranges.

Similar to what you do on-premises, you'll want to segment the larger address space into subnets. You can use [CIDR](#) based subnetting principles to create your subnets.

Routing between subnets will happen automatically and you do not need to manually configure routing tables.

However, the default setting is that there are no network access controls between the subnets you create on the Azure Virtual Network. In order to create network access controls between subnets, you'll need to put something between the subnets.

One of the things you can use to accomplish this task is a [Network Security Group](#) (NSG). NSGs are simple stateful packet inspection devices that use the 5-tuple (the source IP, source port, destination IP, destination port, and layer 4 protocol) approach to create allow/deny rules for network traffic. You can allow or deny traffic to and from single IP address, to and from multiple IP addresses or even to and from entire subnets.

Using NSGs for network access control between subnets enables you to put resources that belong to the same security zone or role in their own subnets. For example, think of a simple 3-tier application that has a web tier, an application logic tier and a database tier. You put virtual machines that belong to each of these tiers into their own subnets. Then you use NSGs to control traffic between the subnets:

- Web tier virtual machines can only initiate connections to the application logic machines and can only accept connections from the Internet
- Application logic virtual machines can only initiate connections with database tier and can only accept connections from the web tier
- Database tier virtual machines cannot initiate connection with anything outside of their own subnet and can only accept connections from the application logic tier

To learn more about Network Security Groups and how you can use them to logically segment your Azure Virtual Networks, please read the article [What is a Network Security Group \(NSG\)](#).

Control routing behavior

When you put a virtual machine on an Azure Virtual Network, you'll notice that the virtual machine can connect to any other virtual machine on the same Azure Virtual Network, even if the other virtual machines are on different subnets. The reason why this is possible is that there is a collection of system routes that are enabled by default that allow this type of communication. These default routes allow virtual machines on the same Azure Virtual Network to initiate connections with each other, and with the Internet (for outbound communications to the Internet only).

While the default system routes are useful for many deployment scenarios, there are times when you want to customize the routing configuration for your deployments. These customizations will allow you to configure the next hop address to reach specific destinations.

We recommend that you configure User Defined Routes when you deploy a virtual network security appliance, which we'll talk about in a later best practice.

NOTE

User Defined Routes are not required and the default system routes will work in most instances.

You can learn more about User Defined Routes and how to configure them by reading the article [What are User Defined Routes and IP Forwarding](#).

Enable Forced Tunneling

To better understand forced tunneling, it's useful to understand what "split tunneling" is. The most common example of split tunneling is seen with VPN connections. Imagine that you establish a VPN connection from your hotel room to your corporate network. This connection allows you to connect to resources on your corporate network and all communications to resources on your corporate network go through the VPN tunnel.

What happens when you want to connect to resources on the Internet? When split tunneling is enabled, those

connections go directly to the Internet and not through the VPN tunnel. Some security experts consider this to be a potential risk and therefore recommend that split tunneling be disabled and all connections, those destined for the Internet and those destined for corporate resources, go through the VPN tunnel. The advantage of doing this is that connections to the Internet are then forced through the corporate network security devices, which wouldn't be the case if the VPN client connected to the Internet outside of the VPN tunnel.

Now let's bring this back to virtual machines on an Azure Virtual Network. The default routes for an Azure Virtual Network allow virtual machines to initiate traffic to the Internet. This too can represent a security risk, as these outbound connections could increase the attack surface of a virtual machine and be leveraged by attackers. For this reason, we recommend that you enable forced tunneling on your virtual machines when you have cross-premises connectivity between your Azure Virtual Network and your on-premises network. We will talk about cross-premises connectivity later in this Azure networking best practices document.

If you do not have a cross premises connection, make sure you take advantage of Network Security Groups (discussed earlier) or Azure virtual network security appliances (discussed next) to prevent outbound connections to the Internet from your Azure Virtual Machines.

To learn more about forced tunneling and how to enable it, please read the article [Configure Forced Tunneling using PowerShell and Azure Resource Manager](#).

Use virtual network appliances

While Network Security Groups and User Defined Routing can provide a certain measure of network security at the network and transport layers of the [OSI model](#), there are going to be situations where you'll want or need to enable security at high levels of the stack. In such situations, we recommend that you deploy virtual network security appliances provided by Azure partners.

Azure network security appliances can deliver significantly enhanced levels of security over what is provided by network level controls. Some of the network security capabilities provided by virtual network security appliances include:

- Firewalling
- Intrusion detection/Intrusion Prevention
- Vulnerability management
- Application control
- Network-based anomaly detection
- Web filtering
- Antivirus
- Botnet protection

If you require a higher level of network security than you can obtain with network level access controls, then we recommend that you investigate and deploy Azure virtual network security appliances.

To learn about what Azure virtual network security appliances are available, and about their capabilities, please visit the [Azure Marketplace](#) and search for "security" and "network security".

Deploy DMZs for security zoning

A DMZ or "perimeter network" is a physical or logical network segment that is designed to provide an additional layer of security between your assets and the Internet. The intent of the DMZ is to place specialized network access control devices on the edge of the DMZ network so that only desired traffic is allowed past the network security device and into your Azure Virtual Network.

DMZs are useful because you can focus your network access control management, monitoring, logging and reporting on the devices at the edge of your Azure Virtual Network. Here you would typically enable DDoS

prevention, Intrusion Detection/Intrusion Prevention systems (IDS/IPS), firewall rules and policies, web filtering, network antimalware and more. The network security devices sit between the Internet and your Azure Virtual Network and have an interface on both networks.

While this is the basic design of a DMZ, there are many different DMZ designs, such as back-to-back, tri-homed, multi-homed, and others.

We recommend for all high security deployments that you consider deploying a DMZ to enhance the level of network security for your Azure resources.

To learn more about DMZs and how to deploy them in Azure, please read the article [Microsoft Cloud Services and Network Security](#).

Avoid exposure to the Internet with dedicated WAN links

Many organizations have chosen the Hybrid IT route. In hybrid IT, some of the company's information assets are in Azure, while others remain on-premises. In many cases some components of a service will be running in Azure while other components remain on-premises.

In the hybrid IT scenario, there is usually some type of cross-premises connectivity. This cross-premises connectivity allows the company to connect their on-premises networks to Azure Virtual Networks. There are two cross-premises connectivity solutions available:

- Site-to-site VPN
- ExpressRoute

[Site-to-site VPN](#) represents a virtual private connection between your on-premises network and an Azure Virtual Network. This connection takes place over the Internet and allows you to "tunnel" information inside an encrypted link between your network and Azure. Site-to-site VPN is a secure, mature technology that has been deployed by enterprises of all sizes for decades. Tunnel encryption is performed using [IPsec tunnel mode](#).

While site-to-site VPN is a trusted, reliable, and established technology, traffic within the tunnel does traverse the Internet. In addition, bandwidth is relatively constrained to a maximum of about 200Mbps.

If you require an exceptional level of security or performance for your cross-premises connections, we recommend that you use Azure ExpressRoute for your cross-premises connectivity. ExpressRoute is a dedicated WAN link between your on-premises location or an Exchange hosting provider. Because this is a telco connection, your data doesn't travel over the Internet and therefore is not exposed to the potential risks inherent in Internet communications.

To learn more about how Azure ExpressRoute works and how to deploy, please read the article [ExpressRoute Technical Overview](#).

Optimize uptime and performance

Confidentiality, integrity and availability (CIA) comprise the triad of today's most influential security model. Confidentiality is about encryption and privacy, integrity is about making sure that data is not changed by unauthorized personnel, and availability is about making sure that authorized individuals are able to access the information they are authorized to access. Failure in any one of these areas represents a potential breach in security.

Availability can be thought of as being about uptime and performance. If a service is down, information can't be accessed. If performance is so poor as to make the data unusable, then we can consider the data to be inaccessible. Therefore, from a security perspective, we need to do whatever we can to make sure our services have optimal uptime and performance. A popular and effective method used to enhance availability and performance is to use load balancing. Load balancing is a method of distributing network traffic across servers that are part of a service. For example, if you have front-end web servers as part of your service, you can use load balancing to distribute the

traffic across your multiple front-end web servers.

This distribution of traffic increases availability because if one of the web servers becomes unavailable, the load balancer will stop sending traffic to that server and redirect traffic to the servers that are still online. Load balancing also helps performance, because the processor, network and memory overhead for serving requests is distributed across all the load balanced servers.

We recommend that you employ load balancing whenever you can, and as appropriate for your services. We'll address appropriateness in the following sections. At the Azure Virtual Network level, Azure provides you with three primary load balancing options:

- HTTP-based load balancing
- External load balancing
- Internal load balancing

HTTP-based Load Balancing

HTTP-based load balancing bases decisions about what server to send connections using characteristics of the HTTP protocol. Azure has an HTTP load balancer that goes by the name of Application Gateway.

We recommend that you us Azure Application Gateway when:

- Applications that require requests from the same user/client session to reach the same back-end virtual machine. Examples of this would be shopping cart apps and web mail servers.
- Applications that want to free web server farms from SSL termination overhead by taking advantage of Application Gateway's [SSL offload](#) feature.
- Applications, such as a content delivery network, that require multiple HTTP requests on the same long-running TCP connection to be routed or load balanced to different back-end servers.

To learn more about how Azure Application Gateway works and how you can use it in your deployments, please read the article [Application Gateway Overview](#).

External Load Balancing

External load balancing takes place when incoming connections from the Internet are load balanced among your servers located in an Azure Virtual Network. The Azure External Load balancer can provide you this capability and we recommend that you use it when you don't require the sticky sessions or SSL offload.

In contrast to HTTP-based load balancing, the External Load Balancer uses information at the network and transport layers of the OSI networking model to make decisions on what server to load balance connection to.

We recommend that you use External Load Balancing whenever you have [stateless applications](#) accepting incoming requests from the Internet.

To learn more about how the Azure External Load Balancer works and how you can deploy it, please read the article [Get Started Creating an Internet Facing Load Balancer in Resource Manager using PowerShell](#).

Internal Load Balancing

Internal load balancing is similar to external load balancing and uses the same mechanism to load balance connections to the servers behind them. The only difference is that the load balancer in this case is accepting connections from virtual machines that are not on the Internet. In most cases, the connections that are accepted for load balancing are initiated by devices on an Azure Virtual Network.

We recommend that you use internal load balancing for scenarios that will benefit from this capability, such as when you need to load balance connections to SQL Servers or internal web servers.

To learn more about how Azure Internal Load Balancing works and how you can deploy it, please read the article [Get Started Creating an Internal Load Balancer using PowerShell](#).

Use global load balancing

Public cloud computing makes it possible to deploy globally distributed applications that have components located in datacenters all over the world. This is possible on Microsoft Azure due to Azure's global datacenter presence. In contrast to the load balancing technologies mentioned earlier, global load balancing makes it possible to make services available even when entire datacenters might become unavailable.

You can get this type of global load balancing in Azure by taking advantage of [Azure Traffic Manager](#). Traffic Manager makes it possible to load balance connections to your services based on the location of the user.

For example, if the user is making a request to your service from the EU, the connection is directed to your services located in an EU datacenter. This part of Traffic Manager global load balancing helps to improve performance because connecting to the nearest datacenter is faster than connecting to datacenters that are far away.

On the availability side, global load balancing makes sure that your service is available even if an entire datacenter should become unavailable.

For example, if an Azure datacenter should become unavailable due to environmental reasons or due to outages (such as regional network failures), connections to your service would be rerouted to the nearest online datacenter. This global load balancing is accomplished by taking advantage of DNS policies that you can create in Traffic Manager.

We recommend that you use Traffic Manager for any cloud solution you develop that has a widely distributed scope across multiple regions and requires the highest level of uptime possible.

To learn more about Azure Traffic Manager and how to deploy it, please read the article [What is Traffic Manager](#).

Disable RDP/SSH Access to Azure Virtual Machines

It is possible to reach Azure Virtual Machines using the [Remote Desktop Protocol](#) (RDP) and the [Secure Shell](#) (SSH) protocols. These protocols make it possible to manage virtual machines from remote locations and are standard in datacenter computing.

The potential security problem with using these protocols over the Internet is that attackers can use various [brute force](#) techniques to gain access to Azure Virtual Machines. Once the attackers gain access, they can use your virtual machine as a launch point for compromising other machines on your Azure Virtual Network or even attack networked devices outside of Azure.

Because of this, we recommend that you disable direct RDP and SSH access to your Azure Virtual Machines from the Internet. After direct RDP and SSH access from the Internet is disabled, you have other options you can use to access these virtual machines for remote management:

- Point-to-site VPN
- Site-to-site VPN
- ExpressRoute

[Point-to-site VPN](#) is another term for a remote access VPN client/server connection. A point-to-site VPN enables a single user to connect to an Azure Virtual Network over the Internet. After the point-to-site connection is established, the user will be able to use RDP or SSH to connect to any virtual machines located on the Azure Virtual Network that the user connected to via point-to-site VPN. This assumes that the user is authorized to reach those virtual machines.

Point-to-site VPN is more secure than direct RDP or SSH connections because the user has to authenticate twice before connecting to a virtual machine. First, the user needs to authenticate (and be authorized) to establish the

point-to-site VPN connection; second, the user needs to authenticate (and be authorized) to establish the RDP or SSH session.

A [site-to-site VPN](#) connects an entire network to another network over the Internet. You can use a site-to-site VPN to connect your on-premises network to an Azure Virtual Network. If you deploy a site-to-site VPN, users on your on-premises network will be able to connect to virtual machines on your Azure Virtual Network by using the RDP or SSH protocol over the site-to-site VPN connection and does not require you to allow direct RDP or SSH access over the Internet.

You can also use a dedicated WAN link to provide functionality similar to the site-to-site VPN. The main differences are 1. the dedicated WAN link doesn't traverse the Internet, and 2. dedicated WAN links are typically more stable and performant. Azure provides you a dedicated WAN link solution in the form of [ExpressRoute](#).

Enable Azure Security Center

Azure Security Center helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Azure Security Center helps you optimize and monitor network security by:

- Providing network security recommendations
- Monitoring the state of your network security configuration
- Alerting you to network based threats both at the endpoint and network levels

We highly recommend that you enable Azure Security Center for all of your Azure deployments.

To learn more about Azure Security Center and how to enable it for your deployments, please read the article [Introduction to Azure Security Center](#).

Securely extend your datacenter into Azure

Many enterprise IT organizations are looking to expand into the cloud instead of growing their on-premises datacenters. This expansion represents an extension of existing IT infrastructure into the public cloud. By taking advantage of cross-premises connectivity options it's possible to treat your Azure Virtual Networks as just another subnet on your on-premises network infrastructure.

However, there is a lot of planning and design issues that need to be addressed first. This is especially important in the area of network security. One of the best ways to understand how you approach such a design is to see an example.

Microsoft has created the [Datacenter Extension Reference Architecture Diagram](#) and supporting collateral to help you understand what such a datacenter extension would look like. This provides an example reference implementation that you can use to plan and design a secure enterprise datacenter extension to the cloud. We recommend that you review this document to get an idea of the key components of a secure solution.

To learn more about how to securely extend your datacenter into Azure, please view the video [Extending Your Datacenter to Microsoft Azure](#).

Azure network security

8/15/2017 • 34 min to read • [Edit Online](#)

We know that security is job one in the cloud and how important it is that you find accurate and timely information about Azure security. One of the best reasons to use Azure for your applications and services is to take advantage of Azure's wide array of security tools and capabilities. These tools and capabilities help make it possible to create secure solutions on the Azure platform.

Microsoft Azure provides confidentiality, integrity, and availability of customer data, while also enabling transparent accountability. To help you better understand the collection of network security controls implemented within Microsoft Azure from the customer's perspective, this article, "Azure Network Security", is written to provide a comprehensive look at the network security controls available with Microsoft Azure.

This paper is intended to inform you about the wide range of network controls that you can configure to enhance the security of the solutions you deploy in Azure. If you are interested in what Microsoft does to secure the network fabric of the Azure platform itself, see the Azure security section in the [Microsoft Trust Center](#).

Azure platform

Azure is a public cloud service platform that supports a broad selection of operating systems, programming languages, frameworks, tools, databases, and devices. It can run Linux containers with Docker integration; build apps with JavaScript, Python, .NET, PHP, Java, and Node.js; build back-ends for iOS, Android, and Windows devices. Azure cloud services support the same technologies millions of developers and IT professionals already rely on and trust.

When you build on, or migrate IT assets to, a public cloud service provider, you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Azure's infrastructure is designed from the facility to applications for hosting millions of customers simultaneously, and it provides a trustworthy foundation upon which businesses can meet their security requirements. In addition, Azure provides you with an extensive collection of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of your organization's deployments.

Abstract

Microsoft public cloud services deliver hyper-scale services and infrastructure, enterprise-grade capabilities, and many choices for hybrid connectivity. You can choose to access these services either via the Internet or with Azure ExpressRoute, which provides private network connectivity. The Microsoft Azure platform allows you to seamlessly extend your infrastructure into the cloud and build multi-tier architectures. Additionally, third parties can enable enhanced capabilities by offering security services and virtual appliances.

Azure's network services maximize flexibility, availability, resiliency, security, and integrity by design. This white paper provides details on the networking functions of Azure and information on how customers can use Azure's native security features to help protect their information assets.

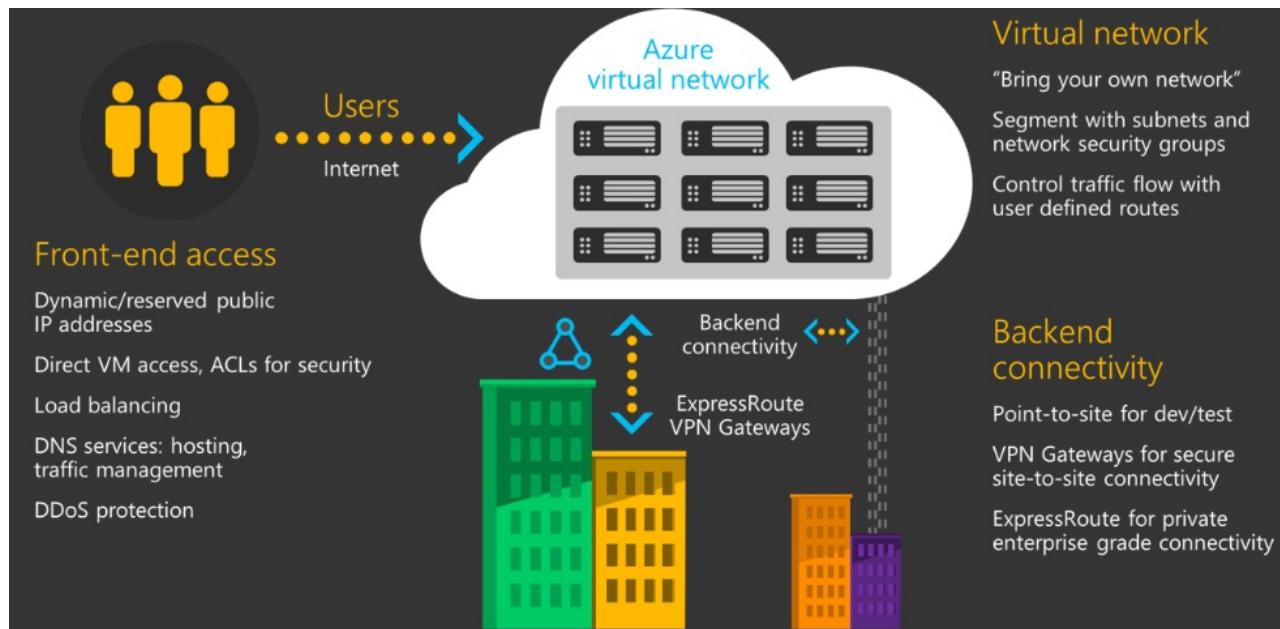
The intended audiences for this whitepaper include:

- Technical managers, network administrators, and developers who are looking for security solutions available and supported in Azure.
- SMEs or business process executives who want to get a high-level overview the Azure networking

technologies and services that are relevant in discussions around network security in the Azure public cloud.

Azure networking big picture

Microsoft Azure includes a robust networking infrastructure to support your application and service connectivity requirements. Network connectivity is possible between resources located in Azure, between on-premises and Azure hosted resources, and to and from the Internet and Azure.



The [Azure network infrastructure](#) enables you to securely connect Azure resources to each other with virtual networks (VNets). A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure cloud network dedicated to your subscription. You can connect VNets to your on-premises networks.

Azure supports dedicated WAN link connectivity to your on-premises network and an Azure Virtual Network with [ExpressRoute](#). The link between Azure and your site uses a dedicated connection that does not go over the public Internet. If your Azure application is running in multiple datacenters, you can use [Azure Traffic Manager](#) to route requests from users intelligently across instances of the application. You can also route traffic to services not running in Azure if they are accessible from the Internet.

Enterprise view of Azure networking components

Azure has many networking components that are relevant to network security discussions. we describe these networking components and focus on the security issues related to them.

NOTE

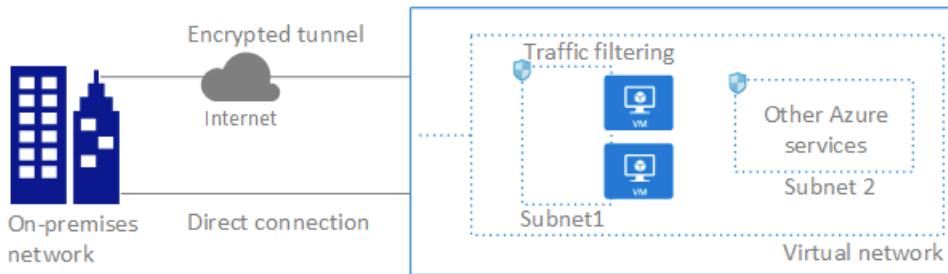
Not all aspects of Azure networking are described – we discuss only those considered to be pivotal in planning and designing a secure network infrastructure around your services and applications you deploy in Azure.

In this paper, will be cover the following Azure networking enterprise capabilities:

- Basic network connectivity
- Hybrid Connectivity
- Security Controls
- Network Validation

Basic network connectivity

The [Azure Virtual Network](#) service enables you to securely connect Azure resources to each other with virtual networks (VNet). A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure network infrastructure dedicated to your subscription. You can also connect VNets to each other and to your on-premises networks using site-to-site VPNs and dedicated [WAN links](#).



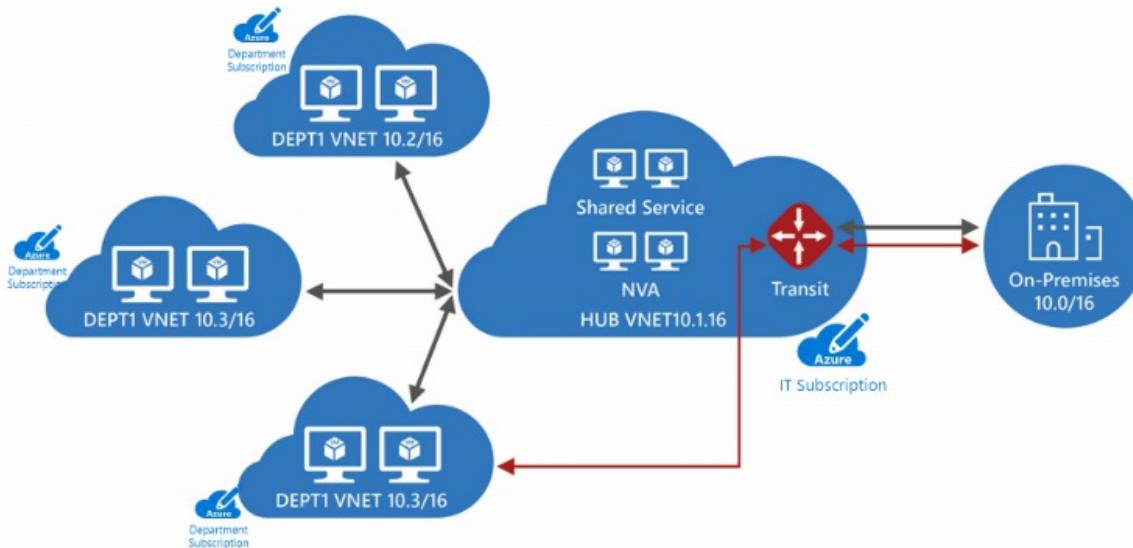
With the understanding that you use VMs to host servers in Azure, the question is how those VMs connect to a network. The answer is that VMs connect to an [Azure Virtual Network](#).

Azure Virtual Networks are like the virtual networks you use on-premises with your own virtualization platform solutions, such as Microsoft Hyper-V or VMware.

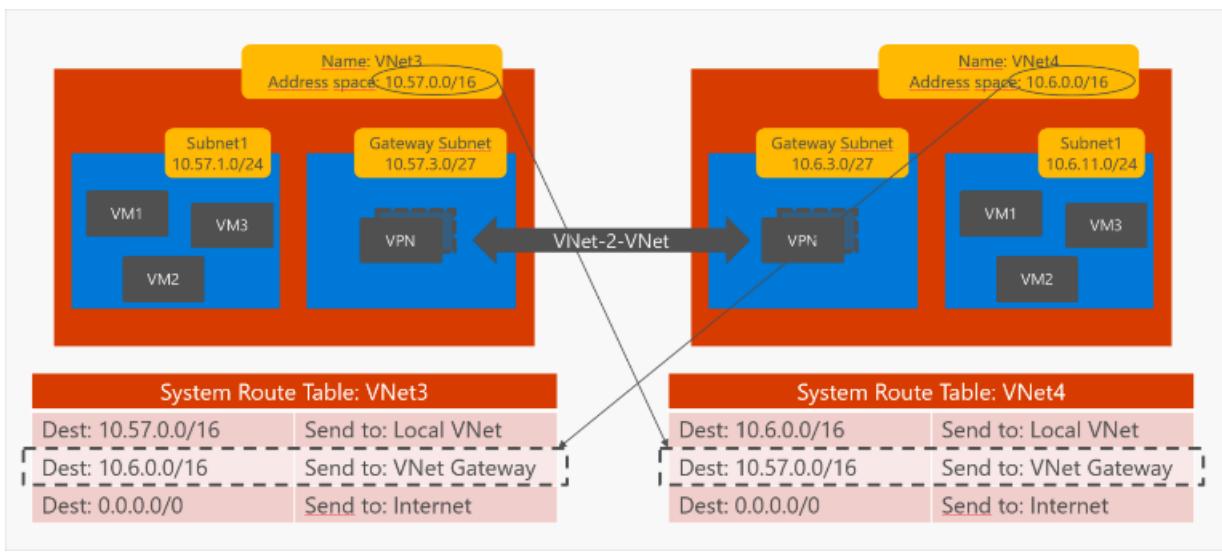
Intra-VNet connectivity

You can connect VNets to each other, enabling resources connected to either VNet to communicate with each other across VNets. You can use either or both of the following options to connect VNets to each other:

- **Peering:** Enables resources connected to different Azure VNets within the same Azure location to communicate with each other. The bandwidth and latency across the VNet is the same as if the resources were connected to the same VNet. To learn more about peering, read [Virtual network peering](#).



- **VNet-to-VNet connection:** Enables resources connected to different Azure VNet within the same, or different Azure locations. Unlike peering, bandwidth is limited between VNets because traffic must flow through an Azure VPN Gateway.



To learn more about connecting VNets with a VNet-to-VNet connection, read the [Configure a VNet-to-VNet connection article](#).

Azure virtual network capabilities:

As you can see, an Azure Virtual Network provides virtual machines to connect to the network so that they can connect to other network resources in a secure fashion. However, basic connectivity is just the beginning. The following capabilities of the Azure Virtual Network service expose security characteristics of the Azure Virtual Network:

- Isolation
- Internet connectivity
- Azure resource connectivity
- VNet connectivity
- On-premises connectivity
- Traffic filtering
- Routing

Isolation

VNets are [isolated](#) from one another. You can create separate VNets for development, testing, and production that use the same [CIDR](#) address blocks. Conversely, you can create multiple VNets that use different CIDR address blocks and connect networks together. You can segment a VNet into multiple subnets.

Azure provides internal name resolution for VMs and [Cloud Services](#) role instances connected to a VNet. You can optionally configure a VNet to use your own DNS servers, instead of using Azure internal name resolution.

You can implement multiple VNets within each Azure [subscription](#) and Azure [region](#). Each VNet is isolated from other VNets. For each VNet you can:

- Specify a custom private IP address space using public and private (RFC 1918) addresses. Azure assigns resources connected to the VNet a private IP address from the address space, you assign.
- Segment the VNet into one or more subnets and allocate a portion of the VNet address space to each subnet.
- Use Azure-provided name resolution or specify your own DNS server for use by resources connected to a VNet. To learn more about name resolution in VNets, read the [Name resolution for VMs and Cloud Services](#).

Internet connectivity

All [Azure Virtual Machines \(VM\)](#) and Cloud Services role instances connected to a VNet have access to the Internet, by default. You can also enable inbound access to specific resources, as needed.(VM) and Cloud Services role instances connected to a VNet have access to the Internet, by default. You can also enable inbound access to specific resources, as needed.

All resources connected to a VNet have outbound connectivity to the Internet by default. The private IP address of the resource is source network address translated (SNAT) to a public IP address by the Azure infrastructure. You can change the default connectivity by implementing custom routing and traffic filtering. To learn more about outbound Internet connectivity, read the [Understanding outbound connections in Azure](#).

To communicate inbound to Azure resources from the Internet, or to communicate outbound to the Internet without SNAT, a resource must be assigned a public IP address. To learn more about public IP addresses, read the [Public IP addresses](#).

Azure resource connectivity

[Azure resources](#) such as Cloud Services and VMs can be connected to the same VNet. The resources can connect to each other using private IP addresses, even if they are in different subnets. Azure provides default routing between subnets, VNets, and on-premises networks, so you don't have to configure and manage routes.

You can connect several Azure resources to a VNet, such as Virtual Machines (VM), Cloud Services, App Service Environments, and Virtual Machine Scale Sets. VMs connect to a subnet within a VNet through a network interface (NIC). To learn more about NICs, read the [Network interfaces](#).

VNet connectivity

[VNets](#) can be connected to each other, enabling resources connected to any VNet to communicate with any resource on any other VNet.

You can connect VNets to each other, enabling resources connected to either VNet to communicate with each other across VNets. You can use either or both of the following options to connect VNets to each other:

- **Peering:** Enables resources connected to different Azure VNets within the same Azure location to communicate with each other. The bandwidth and latency across the VNets is the same as if the resources were connected to the same VNet. To learn more about peering, read the [Virtual network peering](#).
- **VNet-to-VNet connection:** Enables resources connected to different Azure VNet within the same, or different Azure locations. Unlike peering, bandwidth is limited between VNets because traffic must flow through an Azure VPN Gateway. To learn more about connecting VNets with a VNet-to-VNet connection. To learn more, read the [Configure a VNet-to-VNet connection](#).

On-premises connectivity

VNets can be connected to [on-premises](#) networks through private network connections between your network and Azure, or through a site-to-site VPN connection over the Internet.

You can connect your on-premises network to a VNet using any combination of the following options:

- **Point-to-site virtual private network (VPN):** Established between a single PC connected to your network and the VNet. This connection type is great if you're just getting started with Azure, or for developers, because it requires little or no changes to your existing network. The connection uses the SSTP protocol to provide encrypted communication over the Internet between the PC and the VNet. The latency for a point-to-site VPN is unpredictable since the traffic traverses the Internet.
- **Site-to-site VPN:** Established between your VPN device and an Azure VPN Gateway. This connection type enables any on-premises resource you authorize to access a VNet. The connection is an IPsec/IKE VPN that provides encrypted communication over the Internet between your on-premises device and the Azure VPN gateway. The latency for a site-to-site connection is unpredictable since the traffic traverses the Internet.

- **Azure ExpressRoute:** Established between your network and Azure, through an ExpressRoute partner. This connection is private. Traffic does not traverse the Internet. The latency for an ExpressRoute connection is predictable since traffic doesn't traverse the Internet. To learn more about all the previous connection options, read the [Connection topology diagrams](#).

Traffic filtering

VM and Cloud Services role instances [network traffic](#) can be filtered inbound and outbound by source IP address and port, destination IP address and port, and protocol.

You can filter network traffic between subnets using either or both of the following options:

- **Network security groups (NSG):** Each NSG can contain multiple inbound and outbound security rules that enable you to filter traffic by source and destination IP address, port, and protocol. You can apply an NSG to each NIC in a VM. You can also apply an NSG to the subnet a NIC, or other Azure resource, is connected to. To learn more about NSGs, read the [Network security groups](#).
- **Virtual Network Appliances:** A virtual network appliance is a VM running software that performs a network function, such as a firewall. View a list of available NVAs in the Azure Marketplace. NVAs are also available that provide WAN optimization and other network traffic functions. NVAs are typically used with user-defined or BGP routes. You can also use an NVA to filter traffic between VNets.

Routing

You can optionally override Azure's default routing by configuring your own routes, or using BGP routes through a network gateway.

Azure creates route tables that enable resources connected to any subnet in any VNet to communicate with each other, by default. You can implement either or both of the following options to override the default routes Azure creates:

- **User-defined routes:** You can create custom route tables with routes that control where traffic is routed to for each subnet. To learn more about user-defined routes, read the [User-defined routes](#).
- **BGP routes:** If you connect your VNet to your on-premises network using an Azure VPN Gateway or ExpressRoute connection, you can propagate BGP routes to your VNets.

Hybrid internet connectivity: Connect to an on-premises network

You can connect your on-premises network to a VNet using any combination of the following options:

- Internet connectivity
- Point-to-site VPN (P2S VPN)
- Site-to-Site VPN (S2S VPN)
- ExpressRoute

Internet Connectivity

As its name suggests, Internet connectivity makes your workloads accessible from the Internet, by having you expose different public endpoints to workloads that live inside the virtual network. These workloads could be exposed using [Internet-facing Load Balancer](#) or simply assigning a public IP address to the VM. This way, it becomes possible for anything on the Internet to be able to reach that virtual machine, provided a host firewall, [network security groups \(NSG\)](#), and [User-Defined Routes](#) allow that to happen.

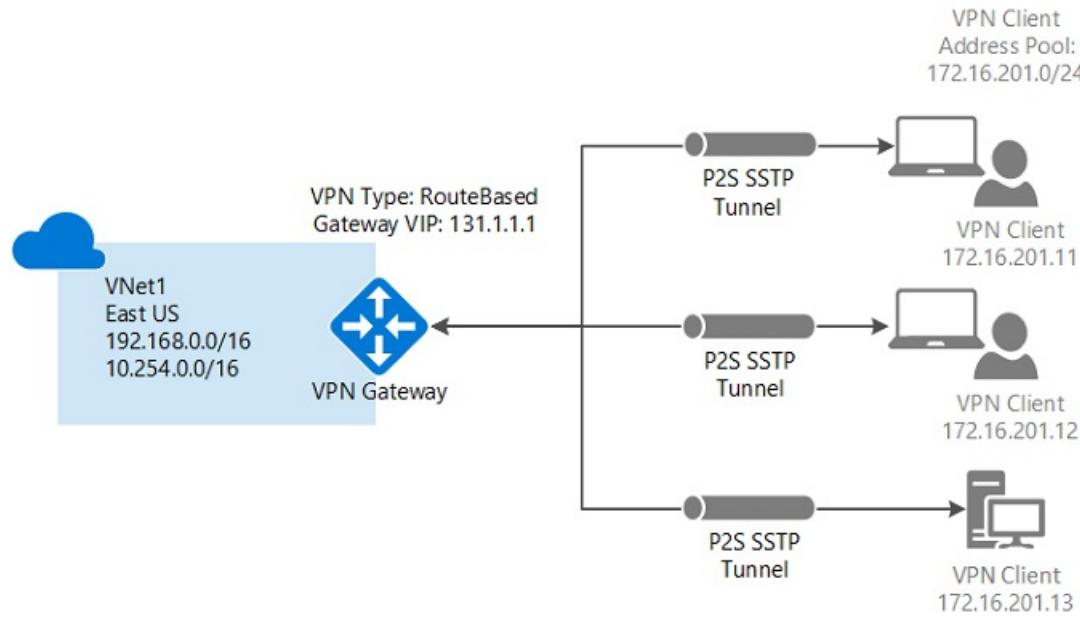
In this scenario, you could expose an application that needs to be public to the Internet and be able to connect to it from anywhere, or from specific locations depending on the configuration of your workloads.

Point-to-Site VPN or Site-to-Site VPN

These two falls into the same category. They both need your VNet to have a VPN Gateway and you can connect to it

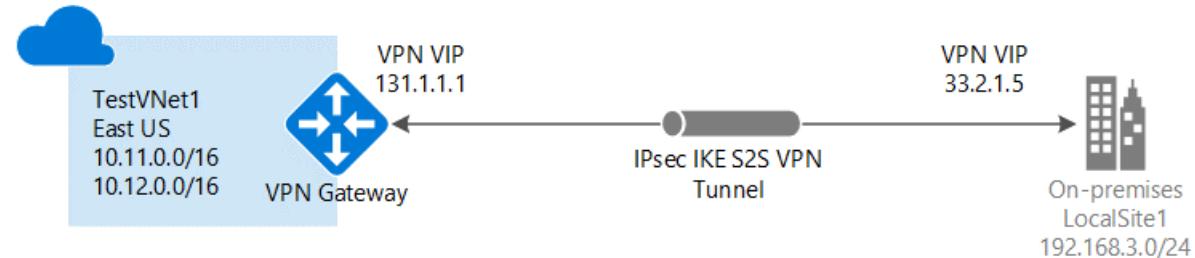
using either a VPN Client for your workstation as part of the [Point-to-Site configuration](#) or you can configure your on-premises [VPN device](#) to be able to terminate a site-to-site VPN. This way, on-premises devices can connect to resources within the VNet.

A Point-to-Site (P2S) configuration lets you create a secure connection from an individual client computer to a virtual network. P2S is a VPN connection over SSTP (Secure Socket Tunneling Protocol).



Point-to-Site connections are useful when you want to connect to your VNet from a remote location, such as from home or a conference center, or when you only have a few clients that need to connect to a virtual network.

P2S connections do not require a VPN device or a public-facing IP address. You establish the VPN connection from the client computer. Therefore, P2S is not recommended way to connect to Azure in case you need a persistent connection from many on-premises devices and computers to your Azure network.



NOTE

For more information about Point-to-Site connections, see the [Point-to-Site FA v Q](#).

A Site-to-Site VPN gateway connection is used to connect your on-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel.

This type of connection requires a VPN device located on-premises that has an externally facing public IP address assigned to it. This connection takes place over the Internet and allows you to "tunnel" information inside an encrypted link between your network and Azure. Site-to-site VPN is a secure, mature technology that has been deployed by enterprises of all sizes for decades. Tunnel encryption is performed using [IPsec tunnel mode](#).

While site-to-site VPN is a trusted, reliable, and established technology, traffic within the tunnel does traverse the Internet. In addition, bandwidth is relatively constrained to a maximum of about 200 Mbps.

If you require an exceptional level of security or performance for your cross-premises connections, we recommend

that you use Azure ExpressRoute for your cross-premises connectivity. ExpressRoute is a dedicated WAN link between your on-premises location or an Exchange hosting provider. Because this is a telco connection, your data doesn't travel over the Internet and therefore is not exposed to the potential risks inherent in Internet communications.

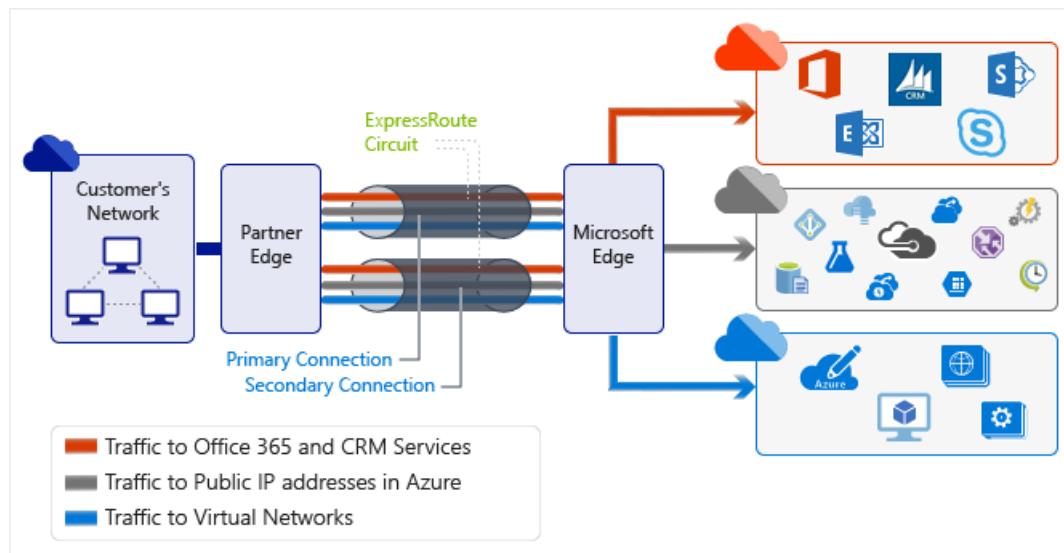
NOTE

For more information about VPN gateways, see [About VPN gateway](#).

Dedicated WAN link

Microsoft Azure ExpressRoute lets you extend your on-premises networks into the Azure over a dedicated private connection facilitated by a connectivity provider.

ExpressRoute connections do not go over the public Internet. This allows ExpressRoute connections to offer more reliability, faster speeds, lower latencies, and higher security than typical connections over the Internet.



NOTE

For information on how to connect your network to Microsoft using ExpressRoute, see [ExpressRoute connectivity models](#) and [ExpressRoute technical overview](#).

As with the site-to-site VPN options, ExpressRoute also allows you to connect to resources that are not necessarily in only one VNet. In fact, depending on the SKU, you can connect to 10 VNets. If you have the [premium add-on](#), connections to up to 100 VNets are possible, depending on bandwidth. To learn more about what these types of connections look like, read [Connection topology diagrams](#).

Security controls

An Azure Virtual Network provides a secure, logical network that is isolated from other virtual networks and supports many security controls that you use on your on-premises networks. Customers create their own structure by using: subnets—they use their own private IP address range, configure route tables, network security groups, access control lists (ACLs), gateways, and virtual appliances to run their workloads in the cloud.

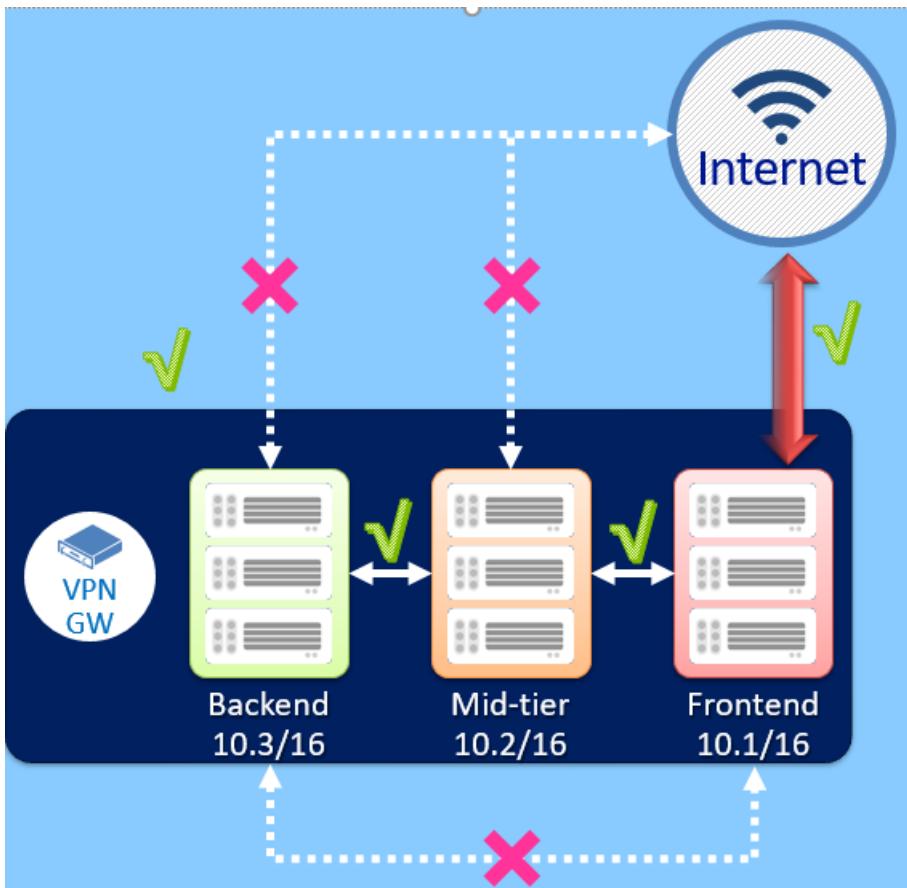
The following are security controls you can use on your Azure Virtual Networks:

- Network Access Controls
- User-Defined Routes
- Network Security Appliance

- Application Gateway
- Azure Web Application Firewall
- Network Availability Control

Network access controls

While the Azure Virtual Network (VNet) is the cornerstone of Azure networking model and provides isolation and protection, the [Network Security Groups \(NSG\)](#) are the main tool you use to enforce and control network traffic rules at the network level.



You can control access by permitting or denying communication between the workloads within a virtual network, from systems on customer's networks via cross-premises connectivity, or direct Internet communication.

In the diagram, both VNets and NSGs reside in a specific layer in the Azure overall security stack, where NSGs, UDR, and network virtual appliances can be used to create security boundaries to protect the application deployments in the protected network.

NSGs use a 5-tuple to evaluate traffic (and are used in the rules you configure for the NSG):

- Source and destination IP address
- Source and destination port
- Protocol: [Transmission Control Protocol \(TCP\)](#) or [User Datagram Protocol \(UDP\)](#)

This means you can control access between a single VM and a group of VMs, or a single VM to another single VM, or between entire subnets. Again, keep in mind that this is simple stateful packet filtering, not full packet inspection. There is no protocol validation or network level IDS or IPS capability in a Network Security Group.

An NSG comes with some built-in rules that you should be aware of. These are:

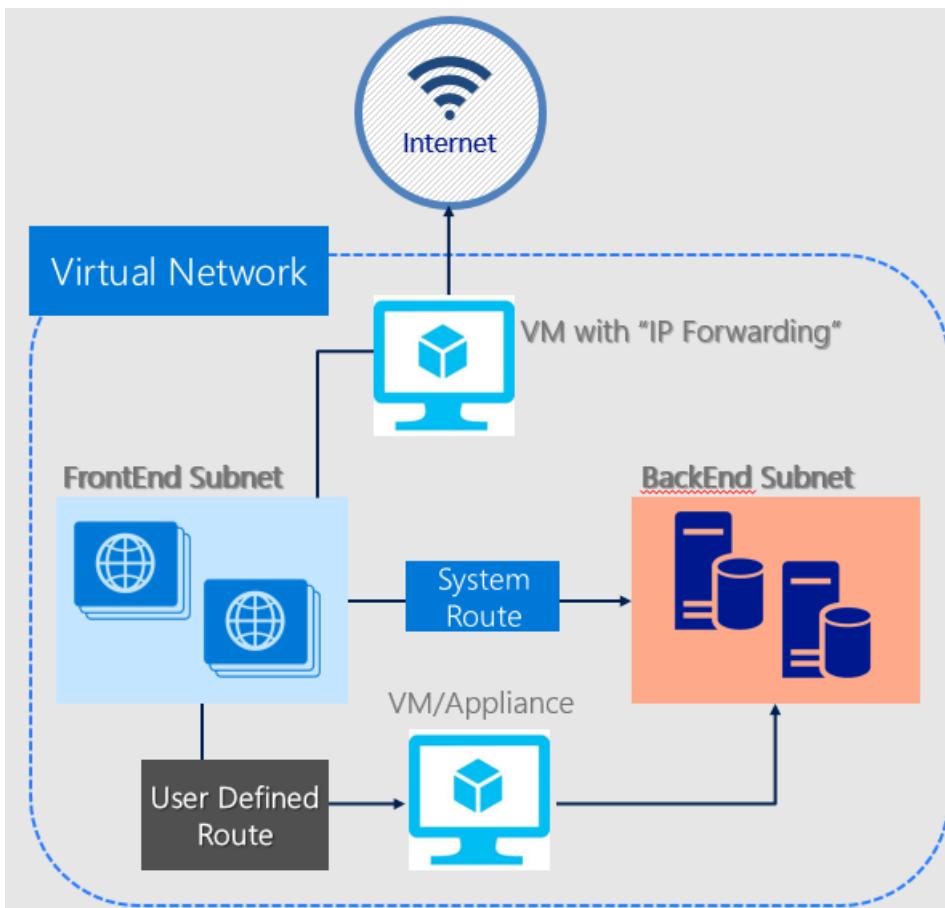
- **Allow all traffic within a specific virtual network:** All VMs on the same Azure Virtual Network can communicate with each other.

- **Allow Azure load balancing to inbound:** This rule allows traffic from any source address to any destination address for the Azure load balancer.
- **Deny all inbound:** This rule blocks all traffic sourcing from the Internet that you have explicitly allowed.
- **Allow all traffic outbound to the Internet:** This rule allows VMs to initiate connections to the Internet. If you do not want these connections initiated, you need to create a rule to block those connections or enforce forced tunneling.

System routes and user-defined routes

When you add virtual machines (VMs) to a virtual network (VNet) in Azure, you notice that the VMs are able to communicate with each other over the network, automatically. You do not need to specify a gateway, even though the VMs are in different subnets.

The same is true for communication from the VMs to the public Internet, and even to your on-premises network when a hybrid connection from Azure to your own datacenter is present.



This flow of communication is possible because Azure uses a series of system routes to define how IP traffic flows. System routes control the flow of communication in the following scenarios:

- From within the same subnet.
- From a subnet to another within a VNet.
- From VMs to the Internet.
- From a VNet to another VNet through a VPN gateway.
- From a VNet to another VNet through VNet Peering ([Service Chaining](#)).
- From a VNet to your on-premises network through a VPN gateway.

Many enterprises have strict security and compliance requirements that require on-premises inspection of all network packets to enforce specific policies. Azure provides a mechanism called [forced tunneling](#) that routes traffic

from the VMs to on-premises by creating a custom route or by [Border Gateway Protocol \(BGP\)](#) advertisements through ExpressRoute or VPN.

Forced tunneling in Azure is configured via virtual network user-defined routes (UDR). Redirecting traffic to an on-premises site is expressed as a Default Route to the Azure VPN gateway.

The following section lists the current limitation of the routing table and routes for an Azure Virtual Network:

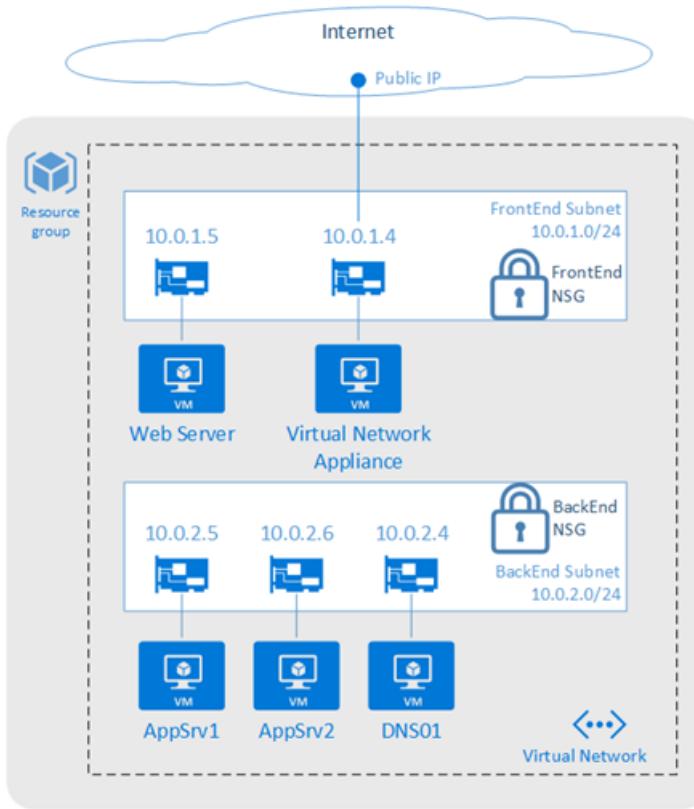
- Each virtual network subnet has a built-in, system routing table. The system routing table has the following three groups of routes:
 - **Local VNet routes:** Directly to the destination VMs in the same virtual network
 - **On premises routes:** To the Azure VPN gateway
 - **Default route:** Directly to the Internet. Packets destined to the private IP addresses not covered by the previous two routes are dropped.
- With the release of user-defined routes, you can create a routing table to add a default route, and then associate the routing table to your VNet subnet to enable forced tunneling on those subnets.
- You need to set a "default site" among the cross-premises local sites connected to the virtual network.
- Forced tunneling must be associated with a VNet that has a dynamic routing VPN gateway (not a static gateway).
- ExpressRoute forced tunneling is not configured via this mechanism, but instead, is enabled by advertising a default route via the ExpressRoute BGP peering sessions.

NOTE

For more information, see the [ExpressRoute Documentation](#) for more information.

Network security appliances

While Network Security Groups and User-Defined Routes can provide a certain measure of network security at the network and transport layers of the [OSI model](#), there are going to be situations where you want or need to enable security at higher levels of the networking stack. In such situations, we recommend that you deploy virtual network security appliances provided by Azure partners.

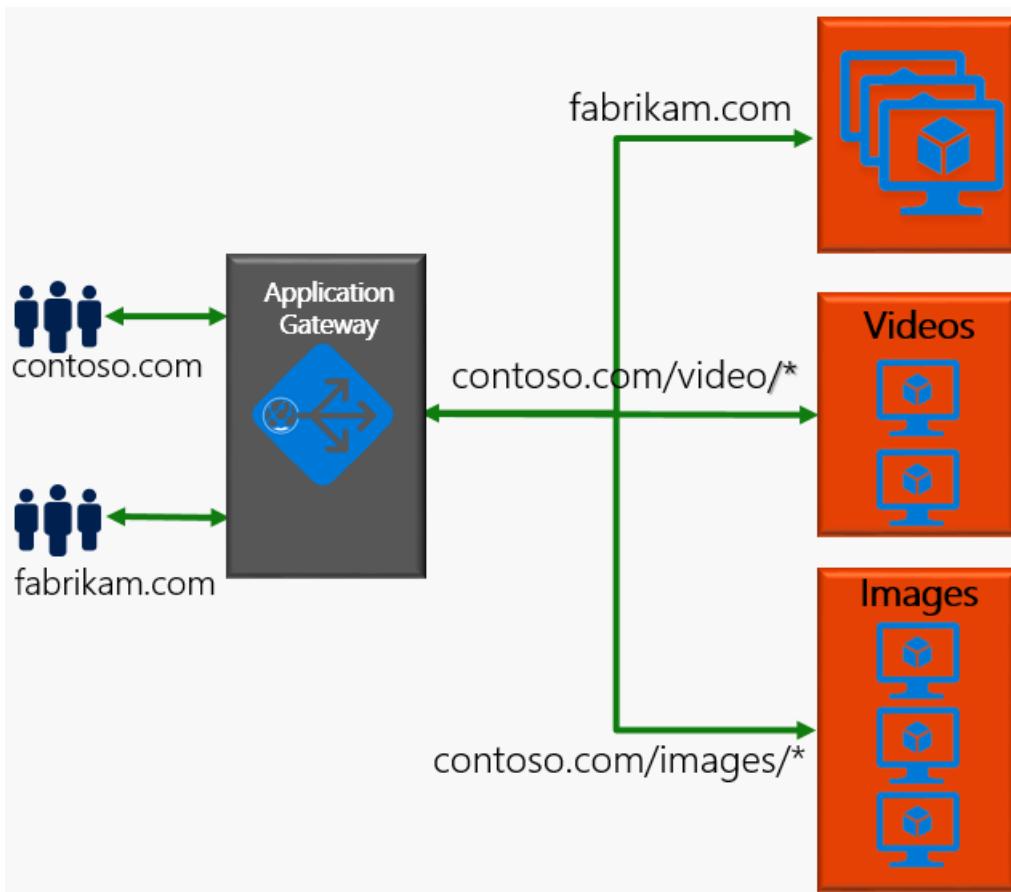


Azure network security appliances improve VNet security and network functions, and they're available from numerous vendors via the [Azure Marketplace](#). These virtual security appliances can be deployed to provide:

- Highly available firewalls
- Intrusion prevention
- Intrusion detection
- Web application firewalls (WAFs)
- WAN optimization
- Routing
- Load balancing
- VPN
- Certificate management
- Active Directory
- Multifactor authentication

Application gateway

[Microsoft Azure Application Gateway](#) is a dedicated virtual appliance that provides an application delivery controller (ADC) as a service.

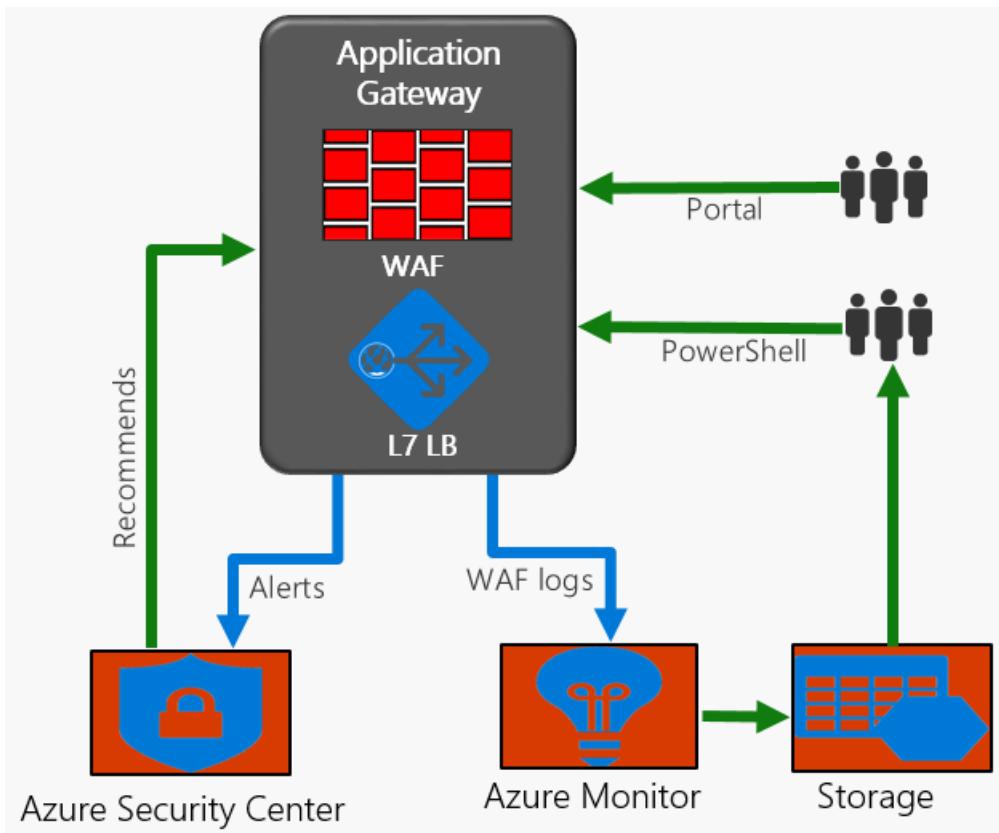


Application Gateway enables you to optimize web farm performance and availability by offloading CPU intensive SSL termination to the application gateway (SSL-offloading). It also provides other layer 7 routing capabilities including:

- Round-robin distribution of incoming traffic
- Cookie-based session affinity
- URL path-based routing
- Ability to host multiple websites behind a single Application Gateway

A [web application firewall \(WAF\)](#) is also provided as part of the application gateway. This provides protection to web applications from common web vulnerabilities and exploits. Application Gateway can be configured as an Internet facing gateway, internal only gateway, or a combination of both.

Application Gateway WAF can be run in detection or prevention mode. A common use case is for administrators to run in detection mode to observe traffic for malicious patterns. Once potential exploits are detected, turning to prevention mode blocks suspicious incoming traffic.



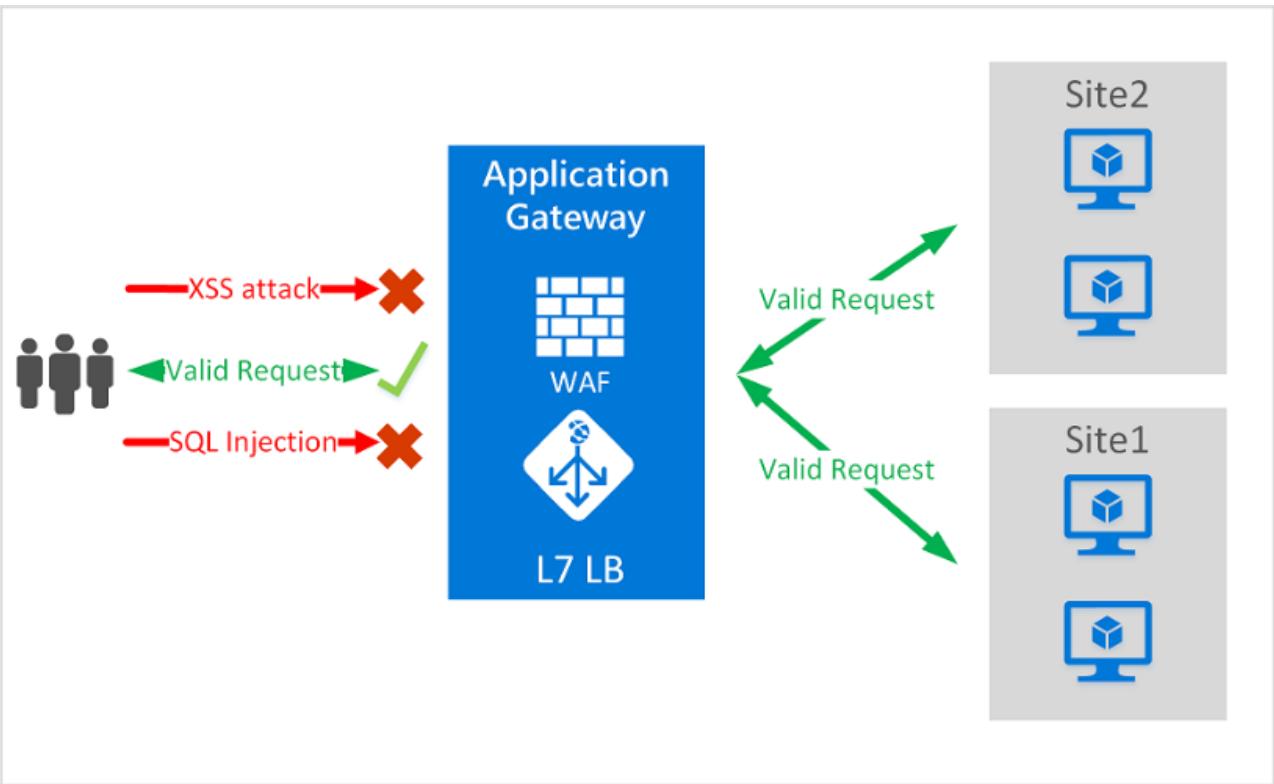
In addition, Application Gateway WAF helps you monitor web applications against attacks using a real-time WAF log that is integrated with [Azure Monitor](#) and [Azure Security Center](#) to track WAF alerts and easily monitor trends.

The JSON formatted log goes directly to the customer's storage account. You have full control over these logs and can apply your own retention policies.

You can also ingest these logs into your own analytics system using [Azure Log Integration](#). WAF logs are also integrated with [Operations Management Suite \(OMS\)](#) so you can use OMS log analytics to execute sophisticated fine-grained queries.

Azure web application firewall (WAF)

Web applications are increasingly targets of malicious attacks that exploit common known vulnerabilities, such as SQL injection, cross site scripting attacks, and other attacks that appear in the [OWASP top 10](#). Preventing such exploits in the application requires rigorous maintenance, patching, and monitoring at multiple layers of the application topology.



A centralized [web application firewall \(WAF\)](#) can protect against web attacks and simplifies security management without requiring any application changes.

A WAF solution can also react to a security threat faster by patching a known vulnerability at a central location versus securing each of individual web applications. Existing application gateways can be converted to a web application firewall enabled application gateway easily.

Network availability controls

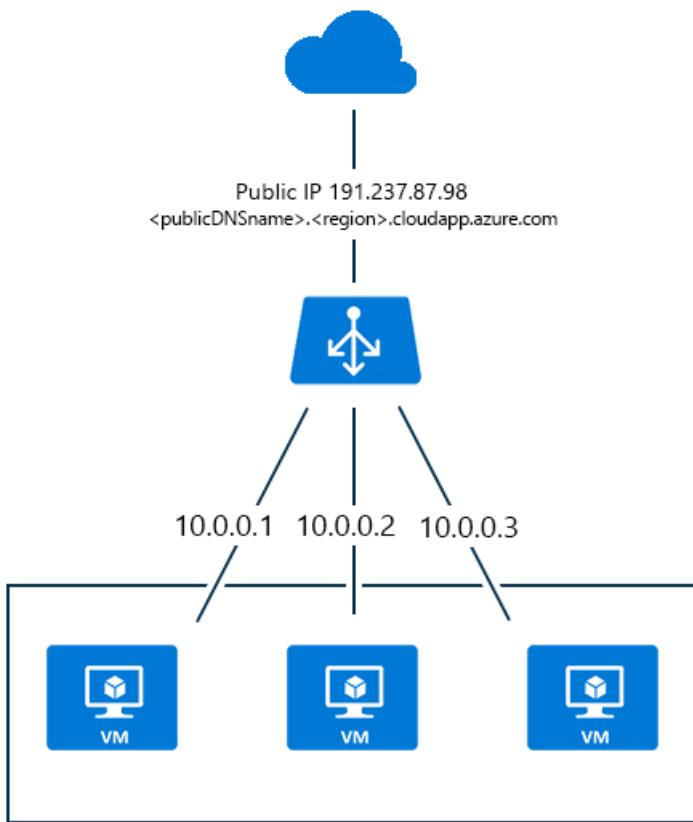
There are different options to distribute network traffic using Microsoft Azure. These options work differently from each other, having a different feature set and support different scenarios. They can each be used in isolation, or combining them.

Following are the Network availability controls:

- Azure Load Balancer
- Application Gateway
- Traffic Manager

Azure Load balancer

Delivers high availability and network performance to your applications. It is a Layer 4 (TCP, UDP) load balancer that distributes incoming traffic among healthy instances of services defined in a load-balanced set.



Azure Load Balancer can be configured to:

- Load balance incoming Internet traffic to virtual machines. This configuration is known as [Internet-facing load balancing](#).
- Load balance traffic between virtual machines in a virtual network, between virtual machines in cloud services, or between on-premises computers and virtual machines in a cross-premises virtual network. This configuration is known as [internal load balancing](#).
- Forward external traffic to a specific virtual machine.

All resources in the cloud need a public IP address to be reachable from the Internet. The cloud infrastructure in Azure uses non-routable IP addresses for its resources. Azure uses network address translation (NAT) with public IP addresses to communicate to the Internet.

Application gateway

Application Gateway works at the application layer (Layer 7 in the OSI network reference stack). It acts as a reverse-proxy service, terminating the client connection and forwarding requests to back-end endpoints.

Traffic manager

Microsoft Azure Traffic Manager allows you to control the distribution of user traffic for service endpoints in different datacenters. Service endpoints supported by Traffic Manager include Azure VMs, Web Apps, and cloud services. You can also use Traffic Manager with external, non-Azure endpoints.

Traffic Manager uses the Domain Name System (DNS) to direct client requests to the most appropriate endpoint based on a [traffic-routing method](#) and the health of the endpoints. Traffic Manager provides a range of traffic-routing methods to suit different application needs, endpoint health [monitoring](#), and automatic failover. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

Azure Traffic Manager enables you to control the distribution of traffic across your application endpoints. An

endpoint is any Internet-facing service hosted inside or outside of Azure.

Traffic Manager provides two key benefits:

- Distribution of traffic according to one of several [traffic-routing methods](#).
- [Continuous monitoring of endpoint health](#) and automatic failover when endpoints fail.

When a client attempts to connect to a service, it must first resolve the DNS name of the service to an IP address. The client then connects to that IP address to access the service. Traffic Manager uses DNS to direct clients to specific service endpoints based on the rules of the traffic-routing method. Clients connect to the selected endpoint directly. Traffic Manager is not a proxy or a gateway. Traffic Manager does not see the traffic passing between the client and the service.

Azure network validation

Azure network validation is to ensure that the Azure network is operating as it is configured and validation can be done using the services and features available to monitor the network. With Azure Network Watcher, you can access a plethora of logging and diagnostic capabilities that empower you with insights to understand your network performance and health. These capabilities are accessible via Portal, Power Shell, CLI, Rest API and SDK.

Azure Operational Security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure. Azure Operational Security is built on a framework that incorporates the knowledge gained through a variety of capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Centre program, and deep awareness of the cyber security threat landscape.

- [Azure Operations Management Suite](#)
- [Azure Security Center](#)
- [Azure Monitor](#)
- [Azure Network Watcher](#)
- [Azure Storage Analytics](#)
- [Azure Resource Manager](#)

Azure resource manager

The people and processes that operate Microsoft Azure are perhaps the most important security feature of the platform. This section describes features of Microsoft's global datacenter infrastructure that help enhance and maintain security, continuity, and privacy.

The infrastructure for your application is typically made up of many components – maybe a virtual machine, storage account, and virtual network, or a web app, database, database server, and third-party services. You do not see these components as separate entities, instead you see them as related and interdependent parts of a single entity. You want to deploy, manage, and monitor them as a group. Azure Resource Manager enables you to work with the resources in your solution as a group.

You can deploy, update, or delete all the resources for your solution in a single, coordinated operation. You use a template for deployment and that template can work for different environments such as testing, staging, and production. Resource Manager provides security, auditing, and tagging features to help you manage your resources after deployment.

The benefits of using Resource Manager

Resource Manager provides several benefits:

- You can deploy, manage, and monitor all the resources for your solution as a group, rather than handling these resources individually.

- You can repeatedly deploy your solution throughout the development lifecycle and have confidence your resources are deployed in a consistent state.
- You can manage your infrastructure through declarative templates rather than scripts.
- You can define the dependencies between resources, so they are deployed in the correct order.
- You can apply access control to all services in your resource group because Role-Based Access Control (RBAC) is natively integrated into the management platform.
- You can apply tags to resources to logically organize all the resources in your subscription.
- You can clarify your organization's billing by viewing costs for a group of resources sharing tag.

NOTE

Resource Manager provides a new way to deploy and manage your solutions. If you used the earlier deployment model and want to learn about the changes, see [Understanding Resource Manager deployment and classic deployment](#).

Azure network logging and monitoring

Azure offers many tools to monitor, prevent, detect, and respond to network security events. Some of the most powerful tools available to you in this area include:

- Network Watcher
- Network Resource Level Monitoring
- Log Analytics

Network watcher

[Network Watcher](#) - Scenario-based monitoring is provided with the features in Network Watcher. This service includes packet capture, next hop, IP flow verify, security group view, NSG flow logs. Scenario level monitoring provides an end to end view of network resources in contrast to individual network resource monitoring.

NAME	RESOURCE TYPE	RESOURCE GROUP	STATUS
netanalyticsNsg	Network security group	netanalytics	Enabled
netanalyticsNsg2	Network security group	netanalytics	Enabled
netanalyticsVm-nsg	Network security group	netanalytics	Disabled
nsgT	Network security group	netanalytics	Disabled
nsgTest	Network security group	netanalytics	Disabled
netAllhandsNSG	Network security group	NwNetAllHandsDemo	Disabled
AppNSG	Network security group	NwTestRgWestCentralUS	Enabled
BackendNSG	Network security group	NwTestRgWestCentralUS	Enabled
FrontendNSG	Network security group	NwTestRgWestCentralUS	Disabled

Network Watcher is a regional service that enables you to monitor and diagnose conditions at a network scenario level in, to, and from Azure. Network diagnostic and visualization tools available with Network Watcher help you understand, diagnose, and gain insights to your network in Azure.

Network Watcher currently has the following capabilities:

Topology

[Topology](#) returns a graph of network resources in a virtual network. The graph depicts the interconnection between the resources to represent the end to end network connectivity. In the portal, Topology returns the resource objects on as per virtual network basis. The relationships are depicted by lines between the resources outside of the Network Watcher region, even if in the resource group will not be displayed. The resources returned in the portal view are a subset of the networking components that are graphed. To see the full list of networking resources, you can use [PowerShell](#) or [REST](#).

As resources are returned the connection between they are modeled under two relationships.

- **Containment** - Virtual Network contains a Subnet, which contains a NIC.
- **Associated** - A NIC is associated with a VM.

Variable packet capture

Network Watcher [variable packet capture](#) allows you to create packet capture sessions to track traffic to and from a virtual machine. Packet capture helps to diagnose network anomalies both reactively and proactively. Other uses include gathering network statistics, gaining information on network intrusions, to debug client-server communications and much more.

Packet capture is a virtual machine extension that is remotely started through Network Watcher. This capability eases the burden of running a packet capture manually on the desired virtual machine, which saves valuable time. Packet capture can be triggered through the portal, PowerShell, CLI, or REST API. One example of how packet capture can be triggered is with Virtual Machine alerts.

IP flow verify

[IP flows verify](#) checks if a packet is allowed or denied to or from a virtual machine based on 5-tuple information. This information consists of direction, protocol, local IP, remote IP, local port, and remote port. If the packet is denied by a security group, the name of the rule that denied the packet is returned. While any source or destination IP can be chosen, this feature helps administrators quickly diagnose connectivity issues from or to the internet and from or to the on-premises environment.

IP flows verify targets a network interface of a virtual machine. Traffic flow is then verified based on the configured settings to or from that network interface. This capability is useful in confirming if a rule in a Network Security Group is blocking ingress or egress traffic to or from a virtual machine.

Next hop

Determines the [next hop](#) for packets being routed in the Azure Network Fabric, enabling you to diagnose any misconfigured user-defined routes. Traffic from a VM is sent to a destination based on the effective routes associated with a NIC. Next hop gets the next hop type and IP address of a packet from a specific virtual machine and NIC. This helps to determine if the packet is being directed to the destination or is the traffic being black holed.

Next hop also returns the route table associated with the next hop. When querying a next hop if the route is defined as a user-defined route, that route will be returned. Otherwise Next hop returns "System Route".

Security group view

Gets the effective and applied security rules that are applied on a VM. Network Security groups are associated at a subnet level or at a NIC level. When associated at a subnet level, it applies to all the VM instances in the subnet. Network [Security Group view](#) returns all the configured NSGs and rules that are associated at a NIC and subnet level for a virtual machine providing insight into the configuration. In addition, the effective security rules are returned for each of the NICs in a VM. Using Network Security Group view, you can assess a VM for network vulnerabilities such as open ports. You can also validate if your Network Security Group is working as expected

based on a [comparison between the configured and the effective security rules](#).

NSG Flow logging

Flow logs for Network Security Groups enable you to capture logs related to traffic that are allowed or denied by the security rules in the group. The flow is defined by a 5-tuple information – Source IP, Destination IP, Source Port, Destination Port, and Protocol.

[Network Security Group flow logs](#) are a feature of Network Watcher that allows you to view information about ingress and egress IP traffic through a Network Security Group.

Virtual network gateway and connection troubleshooting

Network Watcher provides many capabilities as it relates to understanding your network resources in Azure. One of these capabilities is resource troubleshooting. [Resource troubleshooting](#) can be called by PowerShell, CLI, or REST API. When called, Network Watcher inspects the health of a Virtual Network gateway or a Connection and returns its findings.

This section takes you through the different management tasks that are currently available for resource troubleshooting.

- [Troubleshoot a Virtual Network gateway](#)
- [Troubleshoot a Connection](#)

Network subscription limits

[Network subscription limits](#) provide you with details of the usage of each of the network resource in a subscription in a region against the maximum number of resources available.

Configuring diagnostics Log

Network Watcher provides a [diagnostic logs](#) view. This view contains all networking resources that support diagnostic logging. From this view, you can enable and disable networking resources conveniently and quickly.

Network resource level monitoring

The following features are available for resource level monitoring:

Audit log

Operations performed as part of the configuration of networks are logged. These audit logs are essential to establish various compliances. These logs can be viewed in the Azure portal or retrieved using Microsoft tools such as Power BI or third-party tools. Audit logs are available through the portal, PowerShell, CLI, and Rest API.

NOTE

For more information on Audit logs, see [Audit operations with Resource Manager](#). Audit logs are available for operations done on all network resources.

Metrics

Metrics are performance measurements and counters collected over a period. Metrics are currently available for Application Gateway. Metrics can be used to trigger alerts based on threshold. Azure Application Gateway by default monitors the health of all resources in its back-end pool and automatically removes any resource considered unhealthy from the pool. Application Gateway continues to monitor the unhealthy instances and adds them back to the healthy back-end pool once they become available and respond to health probes. Application gateway sends the health probes with the same port that is defined in the back-end HTTP settings. This configuration ensures that the probe is testing the same port that customers would be using to connect to the backend.

NOTE

See [Application Gateway Diagnostics](#) to view how metrics can be used to create alerts.

Diagnostic logs

Periodic and spontaneous events are created by network resources and logged in storage accounts, sent to an Event Hub, or Log Analytics. These logs provide insights into the health of a resource. These logs can be viewed in tools such as Power BI and Log Analytics. To learn how to view diagnostic logs, visit [Log Analytics](#).

Diagnostic logs are available for [Load Balancer](#), [Network Security Groups](#), Routes, and [Application Gateway](#).

Network Watcher provides a diagnostic logs view. This view contains all networking resources that support diagnostic logging. From this view, you can enable and disable networking resources conveniently and quickly.

Log analytics

[Log Analytics](#) is a service in [Operations Management Suite \(OMS\)](#) that monitors your cloud and on-premises environments to maintain their availability and performance. It collects data generated by resources in your cloud and on-premises environments and from other monitoring tools to provide analysis across multiple sources.

Log Analytics offers the following solutions for monitoring your networks:

- Network Performance Monitor (NPM)
- Azure Application Gateway analytics
- Azure Network Security Group analytics

Network performance monitor (NPM)

The [Network Performance Monitor](#) management solution is a network monitoring solution that monitors the health, availability, and reachability of networks.

It is used to monitor connectivity between:

- public cloud and on-premises
- data centers and user locations (branch offices)
- subnets hosting various tiers of a multi-tiered application.

Azure application gateway analytics in log analytics

The following logs are supported for Application Gateways:

- ApplicationGatewayAccessLog
- ApplicationGatewayPerformanceLog
- ApplicationGatewayFirewallLog

The following metrics are supported for Application Gateways:

- 5-minute throughput

Azure network security group analytics in log analytics

The following logs are supported for [network security groups](#):

- **NetworkSecurityGroupEvent:** Contains entries for which NSG rules are applied to VMs and instance roles based on MAC address. The status for these rules is collected every 60 seconds.
- **NetworkSecurityGroupRuleCounter:** Contains entries for how many times each NSG rule is applied to deny or allow traffic.

Next steps

Find out more about security by reading some of our in-depth security topics:

- [Log Analytics for Network Security Groups \(NSGs\)](#)
- [Networking innovations that drive the cloud disruption](#)
- [SONiC: The networking switch software that powers the Microsoft Global Cloud](#)
- [How Microsoft builds its fast and reliable global network](#)
- [Lighting up network innovation](#)

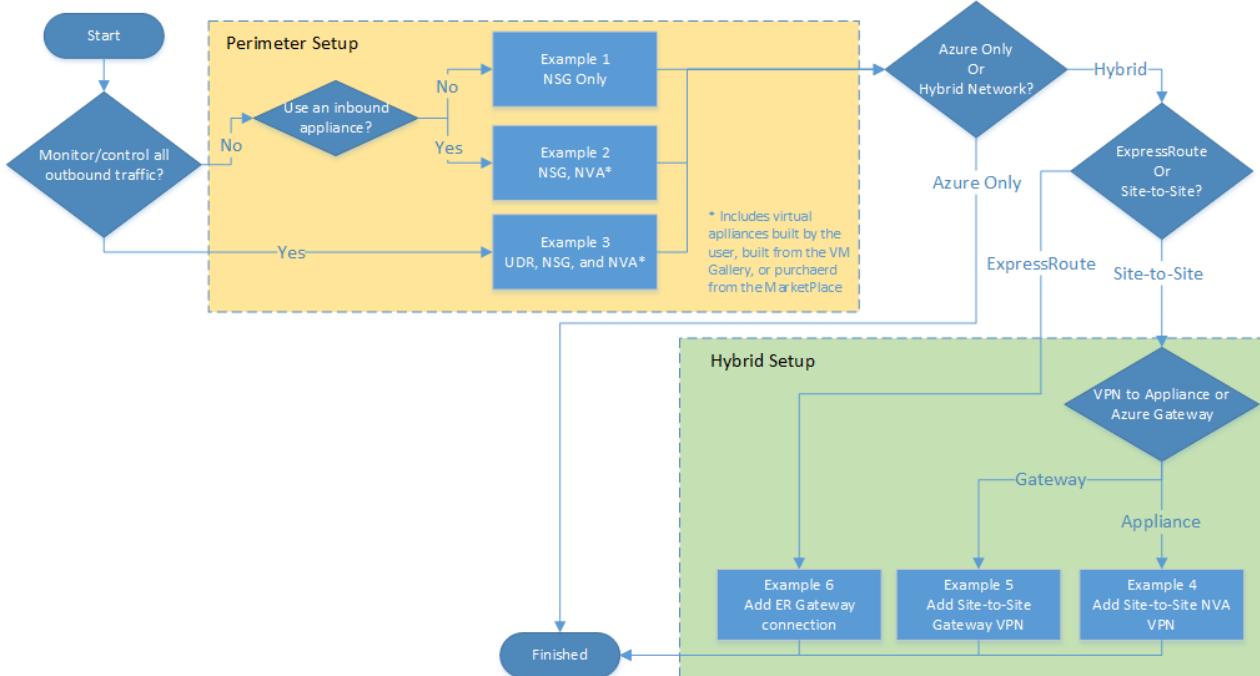
Microsoft cloud services and network security

6/27/2017 • 37 min to read • [Edit Online](#)

Microsoft cloud services deliver hyper-scale services and infrastructure, enterprise-grade capabilities, and many choices for hybrid connectivity. Customers can choose to access these services either via the Internet or with Azure ExpressRoute, which provides private network connectivity. The Microsoft Azure platform allows customers to seamlessly extend their infrastructure into the cloud and build multi-tier architectures. Additionally, third parties can enable enhanced capabilities by offering security services and virtual appliances. This white paper provides an overview of security and architectural issues that customers should consider when using Microsoft cloud services accessed via ExpressRoute. It also covers creating more secure services in Azure virtual networks.

Fast start

The following logic chart can direct you to a specific example of the many security techniques available with the Azure platform. For quick reference, find the example that best fits your case. For expanded explanations, continue reading through the paper.



Example 1: Build a perimeter network (also known as DMZ, demilitarized zone, or screened subnet) to help protect applications with network security groups (NSGs).

Example 2: Build a perimeter network to help protect applications with a firewall and NSGs.

Example 3: Build a perimeter network to help protect networks with a firewall, user-defined route (UDR), and NSG.

Example 4: Add a hybrid connection with a site-to-site, virtual appliance virtual private network (VPN).

Example 5: Add a hybrid connection with a site-to-site, Azure VPN gateway.

Example 6: Add a hybrid connection with ExpressRoute.

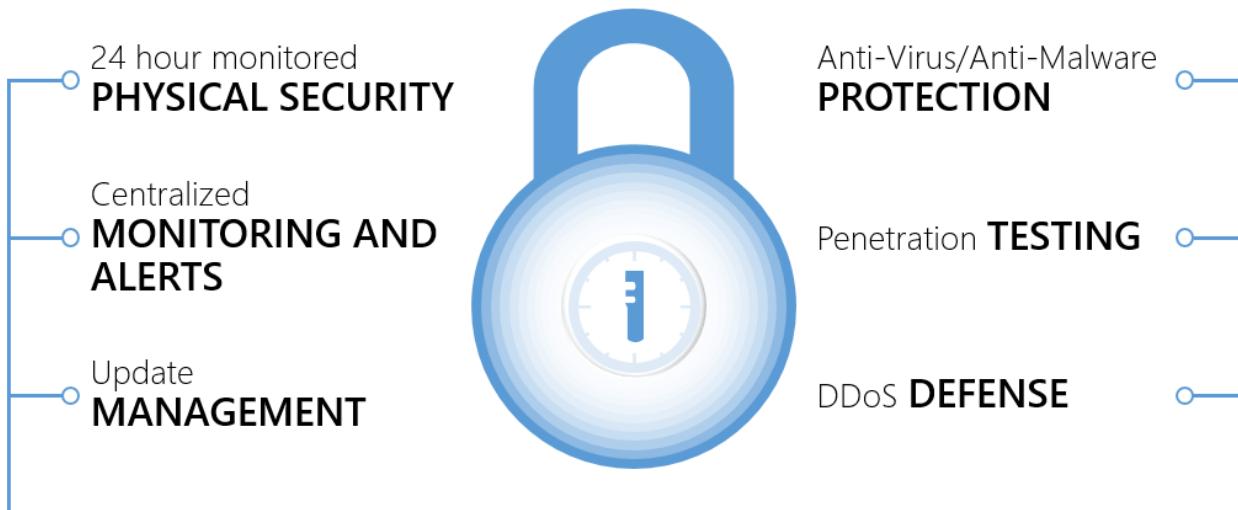
Examples for adding connections between virtual networks, high availability, and service chaining will be added to this document over the next few months.

Microsoft compliance and infrastructure protection

To help organizations comply with national, regional, and industry-specific requirements governing the collection and use of individuals' data, Microsoft offers over 40 certifications and attestations. The most comprehensive set of any cloud service provider.

For more information, see the compliance information on the [Microsoft Trust Center](#).

Microsoft has a comprehensive approach to protect cloud infrastructure needed to run hyper-scale global services. Microsoft cloud infrastructure includes hardware, software, networks, and administrative and operations staff, in addition to the physical data centers.

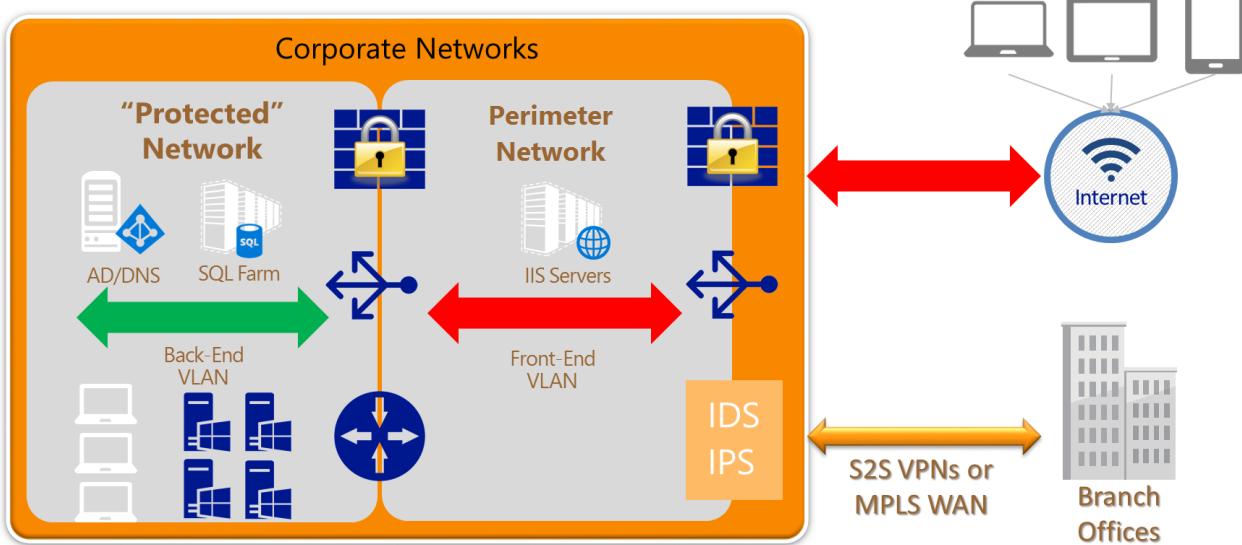


This approach provides a more secure foundation for customers to deploy their services in the Microsoft cloud. The next step is for customers to design and create a security architecture to protect these services.

Traditional security architectures and perimeter networks

Although Microsoft invests heavily in protecting the cloud infrastructure, customers must also protect their cloud services and resource groups. A multilayered approach to security provides the best defense. A perimeter network security zone protects internal network resources from an untrusted network. A perimeter network refers to the edges or parts of the network that sit between the Internet and the protected enterprise IT infrastructure.

In typical enterprise networks, the core infrastructure is heavily fortified at the perimeters, with multiple layers of security devices. The boundary of each layer consists of devices and policy enforcement points. Each layer can include a combination of the following network security devices: firewalls, Denial of Service (DoS) prevention, Intrusion Detection or Protection Systems (IDS/IPS), and VPN devices. Policy enforcement can take the form of firewall policies, access control lists (ACLs), or specific routing. The first line of defense in the network, directly accepting incoming traffic from the Internet, is a combination of these mechanisms to block attacks and harmful traffic while allowing legitimate requests further into the network. This traffic routes directly to resources in the perimeter network. That resource may then "talk" to resources deeper in the network, transiting the next boundary for validation first. The outermost layer is called the perimeter network because this part of the network is exposed to the Internet, usually with some form of protection on both sides. The following figure shows an example of a single subnet perimeter network in a corporate network, with two security boundaries.

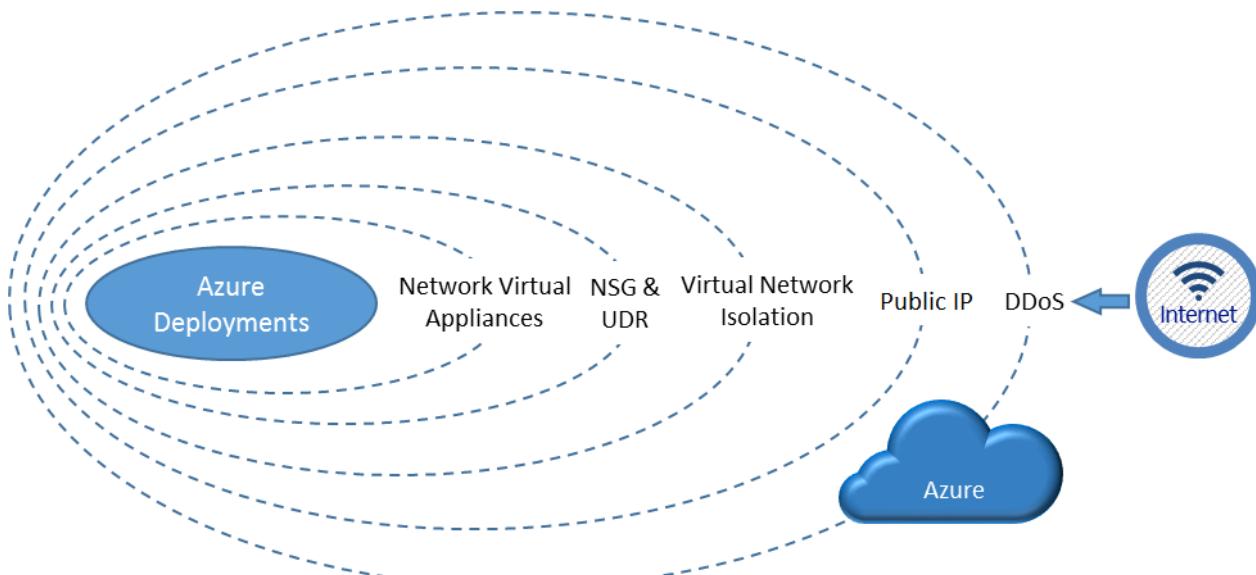


There are many architectures used to implement a perimeter network. These architectures can range from a simple load balancer to a multiple-subnet perimeter network with varied mechanisms at each boundary to block traffic and protect the deeper layers of the corporate network. How the perimeter network is built depends on the specific needs of the organization and its overall risk tolerance.

As customers move their workloads to public clouds, it is critical to support similar capabilities for perimeter network architecture in Azure to meet compliance and security requirements. This document provides guidelines on how customers can build a secure network environment in Azure. It focuses on the perimeter network, but also includes a comprehensive discussion of many aspects of network security. The following questions inform this discussion:

- How can a perimeter network in Azure be built?
- What are some of the Azure features available to build the perimeter network?
- How can back-end workloads be protected?
- How are Internet communications controlled to the workloads in Azure?
- How can the on-premises networks be protected from deployments in Azure?
- When should native Azure security features be used versus third-party appliances or services?

The following diagram shows various layers of security Azure provides to customers. These layers are both native in the Azure platform itself and customer-defined features:



Inbound from the Internet, Azure DDoS helps protect against large-scale attacks against Azure. The next layer is

customer-defined public IP addresses (endpoints), which are used to determine which traffic can pass through the cloud service to the virtual network. Native Azure virtual network isolation ensures complete isolation from all other networks and that traffic only flows through user configured paths and methods. These paths and methods are the next layer, where NSGs, UDR, and network virtual appliances can be used to create security boundaries to protect the application deployments in the protected network.

The next section provides an overview of Azure virtual networks. These virtual networks are created by customers, and are what their deployed workloads are connected to. Virtual networks are the basis of all the network security features required to establish a perimeter network to protect customer deployments in Azure.

Overview of Azure virtual networks

Before Internet traffic can get to the Azure virtual networks, there are two layers of security inherent to the Azure platform:

1. **DDoS protection:** DDoS protection is a layer of the Azure physical network that protects the Azure platform itself from large-scale Internet-based attacks. These attacks use multiple "bot" nodes in an attempt to overwhelm an Internet service. Azure has a robust DDoS protection mesh on all inbound, outbound, and cross-Azure region connectivity. This DDoS protection layer has no user configurable attributes and is not accessible to the customer. The DDoS protection layer protects Azure as a platform from large-scale attacks, it also monitors out-bound traffic and cross-Azure region traffic. Using network virtual appliances on the VNet, additional layers of resilience can be configured by the customer against a smaller scale attack that doesn't trip the platform level protection. An example of DDoS in action; if an internet facing IP address was attacked by a large-scale DDoS attack, Azure would detect the sources of the attacks and scrub the offending traffic before it reached its intended destination. In almost all cases, the attacked endpoint isn't affected by the attack. In the rare cases that an endpoint is affected, no traffic is affected to other endpoints, only the attacked endpoint. Thus other customers and services would see no impact from that attack. It's critical to note that Azure DDoS is only looking for large-scale attacks. It is possible that your specific service could be overwhelmed before the platform level protection thresholds are exceeded. For example, a web site on a single A0 IIS server, could be taken offline by a DDoS attack before Azure platform level DDoS protection registered a threat.
2. **Public IP Addresses:** Public IP addresses (enabled via service endpoints, Public IP addresses, Application Gateway, and other Azure features that present a public IP address to the internet routed to your resource) allow cloud services or resource groups to have public Internet IP addresses and ports exposed. The endpoint uses Network Address Translation (NAT) to route traffic to the internal address and port on the Azure virtual network. This path is the primary way for external traffic to pass into the virtual network. The Public IP addresses are configurable to determine which traffic is passed in, and how and where it's translated on to the virtual network.

Once traffic reaches the virtual network, there are many features that come into play. Azure virtual networks are the foundation for customers to attach their workloads and where basic network-level security applies. It is a private network (a virtual network overlay) in Azure for customers with the following features and characteristics:

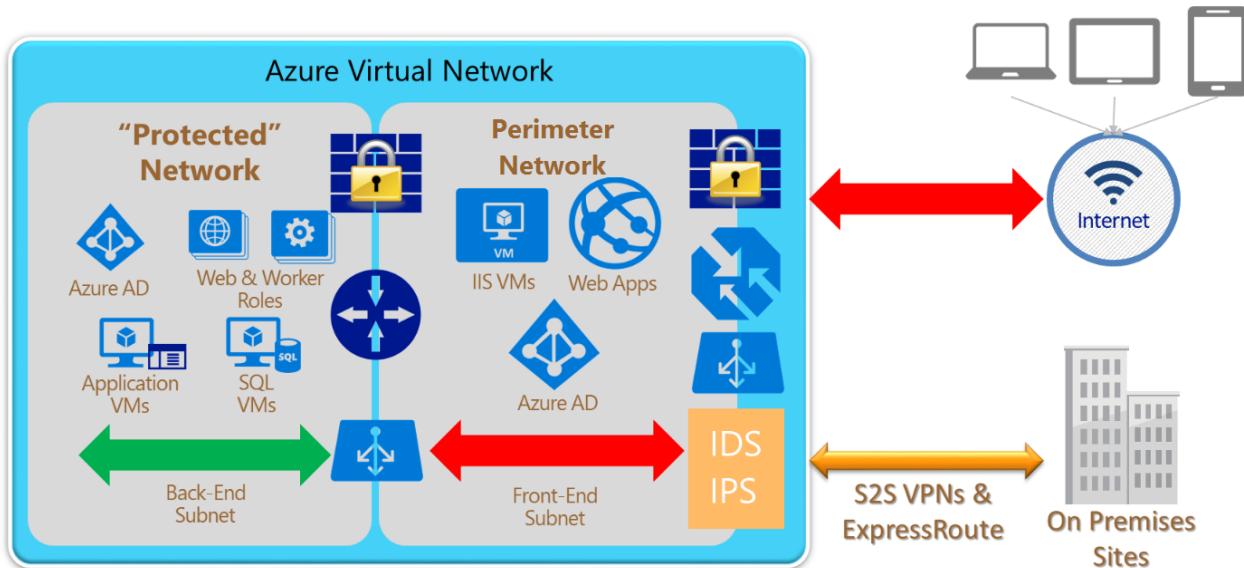
- **Traffic isolation:** A virtual network is the traffic isolation boundary on the Azure platform. Virtual machines (VMs) in one virtual network cannot communicate directly to VMs in a different virtual network, even if both virtual networks are created by the same customer. Isolation is a critical property that ensures customer VMs and communication remains private within a virtual network.

NOTE

Traffic isolation refers only to traffic *inbound* to the virtual network. By default outbound traffic from the VNet to the internet is allowed, but can be prevented if desired by NSGs.

- **Multi-tier topology:** Virtual networks allow customers to define multi-tier topology by allocating subnets and designating separate address spaces for different elements or “tiers” of their workloads. These logical groupings and topologies enable customers to define different access policy based on the workload types, and also control traffic flows between the tiers.
- **Cross-premises connectivity:** Customers can establish cross-premises connectivity between a virtual network and multiple on-premises sites or other virtual networks in Azure. To construct a connection, customers can use VNet Peering, Azure VPN Gateways, third-party network virtual appliances, or ExpressRoute. Azure supports site-to-site (S2S) VPNs using standard IPsec/IKE protocols and ExpressRoute private connectivity.
- **NSG** allows customers to create rules (ACLs) at the desired level of granularity: network interfaces, individual VMs, or virtual subnets. Customers can control access by permitting or denying communication between the workloads within a virtual network, from systems on customer’s networks via cross-premises connectivity, or direct Internet communication.
- **UDR and IP Forwarding** allow customers to define the communication paths between different tiers within a virtual network. Customers can deploy a firewall, IDS/IPS, and other virtual appliances, and route network traffic through these security appliances for security boundary policy enforcement, auditing, and inspection.
- **Network virtual appliances** in the Azure Marketplace: Security appliances such as firewalls, load balancers, and IDS/IPS are available in the Azure Marketplace and the VM Image Gallery. Customers can deploy these appliances into their virtual networks, and specifically, at their security boundaries (including the perimeter network subnets) to complete a multi-tiered secure network environment.

With these features and capabilities, one example of how a perimeter network architecture could be constructed in Azure is the following diagram:



Perimeter network characteristics and requirements

The perimeter network is the front end of the network, directly interfacing communication from the Internet. The incoming packets should flow through the security appliances, such as the firewall, IDS, and IPS, before reaching the back-end servers. Internet-bound packets from the workloads can also flow through the security appliances in the perimeter network for policy enforcement, inspection, and auditing purposes, before leaving the network. Additionally, the perimeter network can host cross-premises VPN gateways between customer virtual networks and on-premises networks.

Perimeter network characteristics

Referencing the previous figure, some of the characteristics of a good perimeter network are as follows:

- Internet-facing:
 - The perimeter network subnet itself is Internet-facing, directly communicating with the Internet.

- Public IP addresses, VIPs, and/or service endpoints pass Internet traffic to the front-end network and devices.
- Inbound traffic from the Internet passes through security devices before other resources on the front-end network.
- If outbound security is enabled, traffic passes through security devices, as the final step, before passing to the Internet.
- Protected network:
 - There is no direct path from the Internet to the core infrastructure.
 - Channels to the core infrastructure must traverse through security devices such as NSGs, firewalls, or VPN devices.
 - Other devices must not bridge Internet and the core infrastructure.
 - Security devices on both the Internet-facing and the protected network facing boundaries of the perimeter network (for example, the two firewall icons shown in the previous figure) may actually be a single virtual appliance with differentiated rules or interfaces for each boundary. For example, one physical device, logically separated, handling the load for both boundaries of the perimeter network.
- Other common practices and constraints:
 - Workloads must not store business critical information.
 - Access and updates to perimeter network configurations and deployments are limited to only authorized administrators.

Perimeter network requirements

To enable these characteristics, follow these guidelines on virtual network requirements to implement a successful perimeter network:

- **Subnet architecture:** Specify the virtual network such that an entire subnet is dedicated as the perimeter network, separated from other subnets in the same virtual network. This separation ensures the traffic between the perimeter network and other internal or private subnet tiers flows through a firewall or IDS/IPS virtual appliance. User-defined routes on the boundary subnets are required to forward this traffic to the virtual appliance.
- **NSG:** The perimeter network subnet itself should be open to allow communication with the Internet, but that does not mean customers should be bypassing NSGs. Follow common security practices to minimize the network surfaces exposed to the Internet. Lock down the remote address ranges allowed to access the deployments or the specific application protocols and ports that are open. There may be circumstances, however, in which a complete lock-down is not possible. For example, if customers have an external website in Azure, the perimeter network should allow the incoming web requests from any public IP addresses, but should only open the web application ports: TCP on port 80 and/or TCP on port 443.
- **Routing table:** The perimeter network subnet itself should be able to communicate to the Internet directly, but should not allow direct communication to and from the back end or on-premises networks without going through a firewall or security appliance.
- **Security appliance configuration:** To route and inspect packets between the perimeter network and the rest of the protected networks, the security appliances such as firewall, IDS, and IPS devices may be multi-homed. They may have separate NICs for the perimeter network and the back-end subnets. The NICs in the perimeter network communicate directly to and from the Internet, with the corresponding NSGs and the perimeter network routing table. The NICs connecting to the back-end subnets have more restricted NSGs and routing tables of the corresponding back-end subnets.
- **Security appliance functionality:** The security appliances deployed in the perimeter network typically perform the following functionality:
 - Firewall: Enforcing firewall rules or access control policies for the incoming requests.
 - Threat detection and prevention: Detecting and mitigating malicious attacks from the Internet.
 - Auditing and logging: Maintaining detailed logs for auditing and analysis.

- Reverse proxy: Redirecting the incoming requests to the corresponding back-end servers. This redirection involves mapping and translating the destination addresses on the front-end devices, typically firewalls, to the back-end server addresses.
- Forward proxy: Providing NAT and performing auditing for communication initiated from within the virtual network to the Internet.
- Router: Forwarding incoming and cross-subnet traffic inside the virtual network.
- VPN device: Acting as the cross-premises VPN gateways for cross-premises VPN connectivity between customer on-premises networks and Azure virtual networks.
- VPN server: Accepting VPN clients connecting to Azure virtual networks.

TIP

Keep the following two groups separate: the individuals authorized to access the perimeter network security gear and the individuals authorized as application development, deployment, or operations administrators. Keeping these groups separate allows for a segregation of duties and prevents a single person from bypassing both applications security and network security controls.

Questions to be asked when building network boundaries

In this section, unless specifically mentioned, the term "networks" refers to private Azure virtual networks created by a subscription administrator. The term doesn't refer to the underlying physical networks within Azure.

Also, Azure virtual networks are often used to extend traditional on-premises networks. It is possible to incorporate either site-to-site or ExpressRoute hybrid networking solutions with perimeter network architectures. This hybrid link is an important consideration in building network security boundaries.

The following three questions are critical to answer when you're building a network with a perimeter network and multiple security boundaries.

1) How many boundaries are needed?

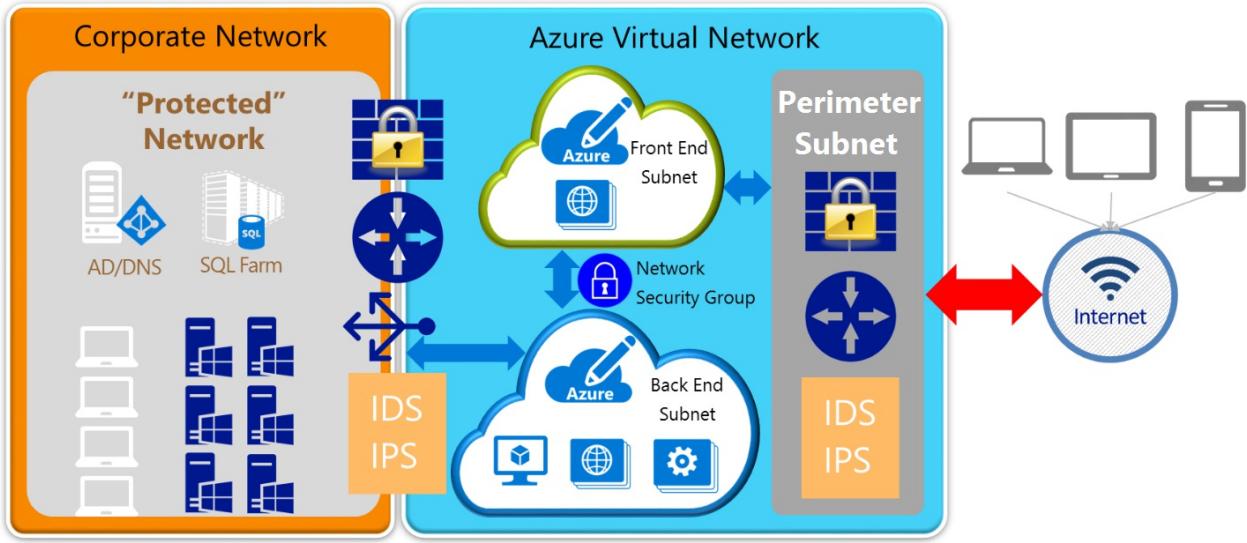
The first decision point is to decide how many security boundaries are needed in a given scenario:

- A single boundary: One on the front-end perimeter network, between the virtual network and the Internet.
- Two boundaries: One on the Internet side of the perimeter network, and another between the perimeter network subnet and the back-end subnets in the Azure virtual networks.
- Three boundaries: One on the Internet side of the perimeter network, one between the perimeter network and back-end subnets, and one between the back-end subnets and the on-premises network.
- N boundaries: A variable number. Depending on security requirements, there is no limit to the number of security boundaries that can be applied in a given network.

The number and type of boundaries needed vary based on a company's risk tolerance and the specific scenario being implemented. This decision is often made together with multiple groups within an organization, often including a risk and compliance team, a network and platform team, and an application development team. People with knowledge of security, the data involved, and the technologies being used should have a say in this decision to ensure the appropriate security stance for each implementation.

TIP

Use the smallest number of boundaries that satisfy the security requirements for a given situation. With more boundaries, operations and troubleshooting can be more difficult, as well as the management overhead involved with managing the multiple boundary policies over time. However, insufficient boundaries increase risk. Finding the balance is critical.



The preceding figure shows a high-level view of a three security boundary network. The boundaries are between the perimeter network and the Internet, the Azure front-end and back-end private subnets, and the Azure back-end subnet and the on-premises corporate network.

2) Where are the boundaries located?

Once the number of boundaries are decided, where to implement them is the next decision point. There are generally three choices:

- Using an Internet-based intermediary service (for example, a cloud-based Web application firewall, which is not discussed in this document)
- Using native features and/or network virtual appliances in Azure
- Using physical devices on the on-premises network

On purely Azure networks, the options are native Azure features (for example, Azure Load Balancers) or network virtual appliances from the rich partner ecosystem of Azure (for example, Check Point firewalls).

If a boundary is needed between Azure and an on-premises network, the security devices can reside on either side of the connection (or both sides). Thus a decision must be made on the location to place security gear.

In the previous figure, the Internet-to-perimeter network and the front-to-back-end boundaries are entirely contained within Azure, and must be either native Azure features or network virtual appliances. Security devices on the boundary between Azure (back-end subnet) and the corporate network could be either on the Azure side or the on-premises side, or even a combination of devices on both sides. There can be significant advantages and disadvantages to either option that must be seriously considered.

For example, using existing physical security gear on the on-premises network side has the advantage that no new gear is needed. It just needs reconfiguration. The disadvantage, however, is that all traffic must come back from Azure to the on-premises network to be seen by the security gear. Thus Azure-to-Azure traffic could incur significant latency, and affect application performance and user experience, if it was forced back to the on-premises network for security policy enforcement.

3) How are the boundaries implemented?

Each security boundary will likely have different capability requirements (for example, IDS and firewall rules on the Internet side of the perimeter network, but only ACLs between the perimeter network and back-end subnet).

Deciding on which device (or how many devices) to use depends on the scenario and security requirements. In the following section, examples 1, 2, and 3 discuss some options that could be used. Reviewing the Azure native network features and the devices available in Azure from the partner ecosystem shows the myriad options available to solve virtually any scenario.

Another key implementation decision point is how to connect the on-premises network with Azure. Should you use the Azure virtual gateway or a network virtual appliance? These options are discussed in greater detail in the

following section (examples 4, 5, and 6).

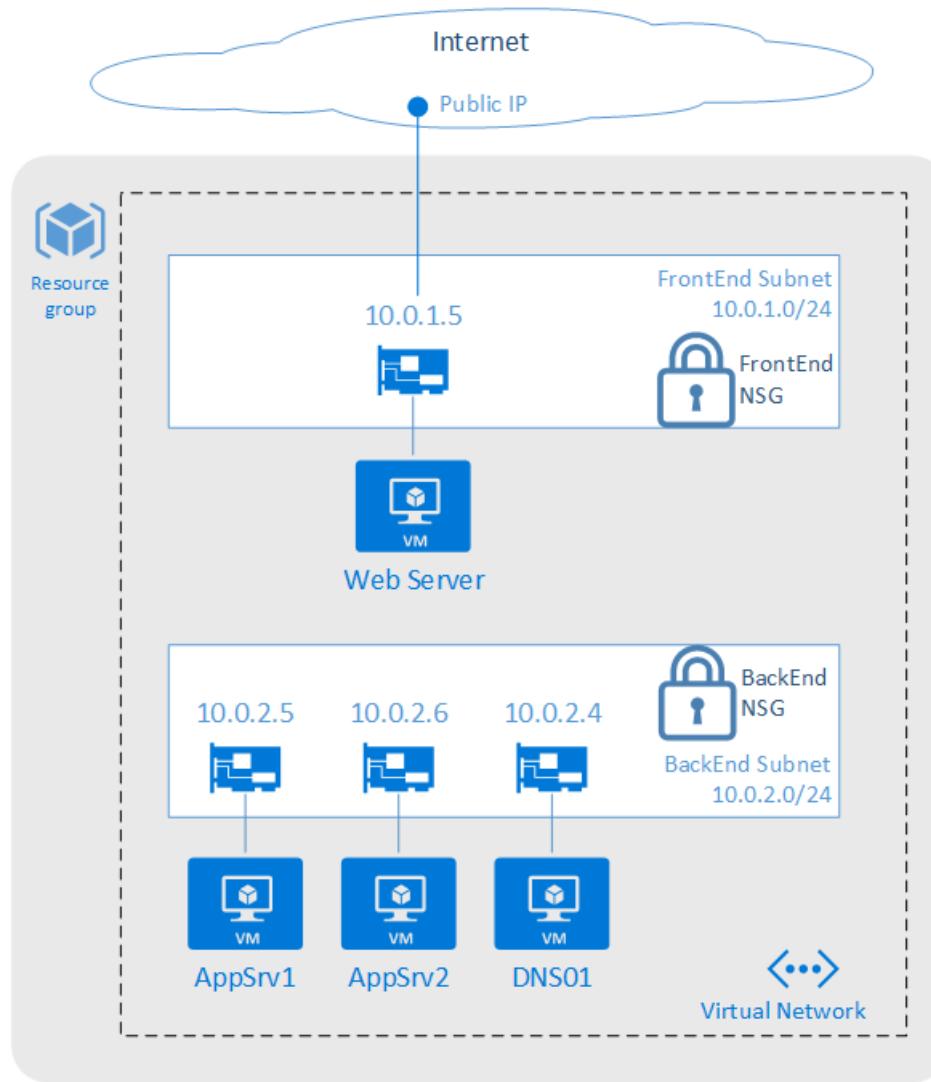
Additionally, traffic between virtual networks within Azure may be needed. These scenarios will be added in the future.

Once you know the answers to the previous questions, the [Fast Start](#) section can help identify which examples are most appropriate for a given scenario.

Examples: Building security boundaries with Azure virtual networks

Example 1 Build a perimeter network to help protect applications with NSGs

[Back to Fast start](#) | [Detailed build instructions for this example](#)



Environment description

In this example, there is a subscription that contains the following resources:

- A single resource group
- A virtual network with two subnets: "FrontEnd" and "BackEnd"
- A Network Security Group that is applied to both subnets
- A Windows server that represents an application web server ("IIS01")
- Two Windows servers that represent application back-end servers ("AppVM01", "AppVM02")
- A Windows server that represents a DNS server ("DNS01")
- A public IP associated with the application web server

For scripts and an Azure Resource Manager template, see the [detailed build instructions](#).

NSG description

In this example, an NSG group is built and then loaded with six rules.

TIP

Generally speaking, you should create your specific "Allow" rules first, followed by the more generic "Deny" rules. The given priority dictates which rules are evaluated first. Once traffic is found to apply to a specific rule, no further rules are evaluated. NSG rules can apply in either the inbound or outbound direction (from the perspective of the subnet).

Declaratively, the following rules are being built for inbound traffic:

1. Internal DNS traffic (port 53) is allowed.
2. RDP traffic (port 3389) from the Internet to any Virtual Machine is allowed.
3. HTTP traffic (port 80) from the Internet to web server (IIS01) is allowed.
4. Any traffic (all ports) from IIS01 to AppVM1 is allowed.
5. Any traffic (all ports) from the Internet to the entire virtual network (both subnets) is denied.
6. Any traffic (all ports) from the front-end subnet to the back-end subnet is denied.

With these rules bound to each subnet, if an HTTP request was inbound from the Internet to the web server, both rules 3 (allow) and 5 (deny) would apply. But because rule 3 has a higher priority, only it would apply, and rule 5 would not come into play. Thus the HTTP request would be allowed to the web server. If that same traffic was trying to reach the DNS01 server, rule 5 (deny) would be the first to apply, and the traffic would not be allowed to pass to the server. Rule 6 (deny) blocks the front-end subnet from talking to the back-end subnet (except for allowed traffic in rules 1 and 4). This rule-set protects the back-end network in case an attacker compromises the web application on the front end. The attacker would have limited access to the back-end "protected" network (only to resources exposed on the AppVM01 server).

There is a default outbound rule that allows traffic out to the Internet. For this example, we're allowing outbound traffic and not modifying any outbound rules. To lock down traffic in both directions, user-defined routing is required (see example 3).

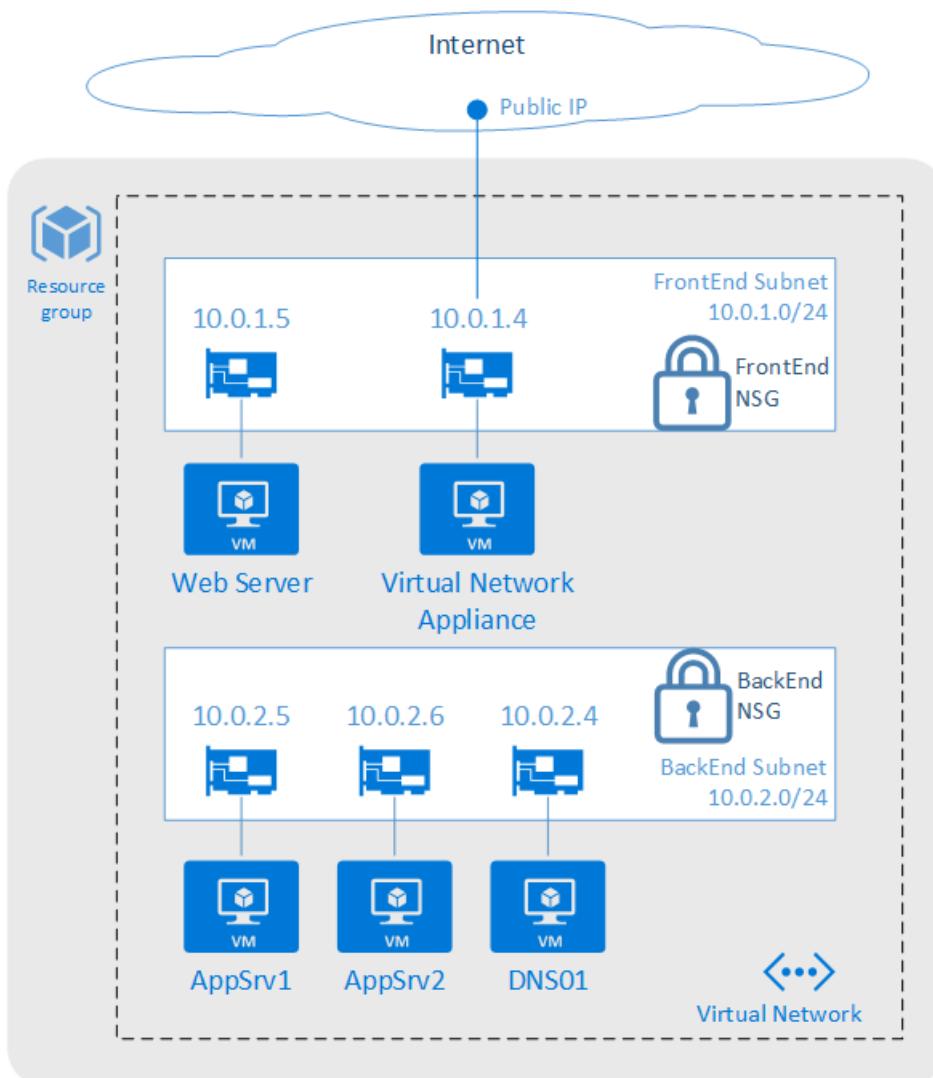
Conclusion

This example is a relatively simple and straightforward way of isolating the back-end subnet from inbound traffic. For more information, see the [detailed build instructions](#). These instructions include:

- How to build this perimeter network with classic PowerShell scripts.
- How to build this perimeter network with an Azure Resource Manager template.
- Detailed descriptions of each NSG command.
- Detailed traffic flow scenarios, showing how traffic is allowed or denied in each layer.

Example 2 Build a perimeter network to help protect applications with a firewall and NSGs

[Back to Fast start](#) | [Detailed build instructions for this example](#)



Environment description

In this example, there is a subscription that contains the following resources:

- A single resource group
- A virtual network with two subnets: "FrontEnd" and "BackEnd"
- A Network Security Group that is applied to both subnets
- A network virtual appliance, in this case a firewall, connected to the front-end subnet
- A Windows server that represents an application web server ("IIS01")
- Two Windows servers that represent application back-end servers ("AppVM01", "AppVM02")
- A Windows server that represents a DNS server ("DNS01")

For scripts and an Azure Resource Manager template, see the [detailed build instructions](#).

NSG description

In this example, an NSG group is built and then loaded with six rules.

TIP

Generally speaking, you should create your specific "Allow" rules first, followed by the more generic "Deny" rules. The given priority dictates which rules are evaluated first. Once traffic is found to apply to a specific rule, no further rules are evaluated. NSG rules can apply in either the inbound or outbound direction (from the perspective of the subnet).

Declaratively, the following rules are being built for inbound traffic:

1. Internal DNS traffic (port 53) is allowed.

2. RDP traffic (port 3389) from the Internet to any Virtual Machine is allowed.
3. Any Internet traffic (all ports) to the network virtual appliance (firewall) is allowed.
4. Any traffic (all ports) from IIS01 to AppVM1 is allowed.
5. Any traffic (all ports) from the Internet to the entire virtual network (both subnets) is denied.
6. Any traffic (all ports) from the front-end subnet to the back-end subnet is denied.

With these rules bound to each subnet, if an HTTP request was inbound from the Internet to the firewall, both rules 3 (allow) and 5 (deny) would apply. But because rule 3 has a higher priority, only it would apply, and rule 5 would not come into play. Thus the HTTP request would be allowed to the firewall. If that same traffic was trying to reach the IIS01 server, even though it's on the front-end subnet, rule 5 (deny) would apply, and the traffic would not be allowed to pass to the server. Rule 6 (deny) blocks the front-end subnet from talking to the back-end subnet (except for allowed traffic in rules 1 and 4). This rule-set protects the back-end network in case an attacker compromises the web application on the front end. The attacker would have limited access to the back-end "protected" network (only to resources exposed on the AppVM01 server).

There is a default outbound rule that allows traffic out to the Internet. For this example, we're allowing outbound traffic and not modifying any outbound rules. To lock down traffic in both directions, user-defined routing is required (see example 3).

Firewall rule description

On the firewall, forwarding rules should be created. Since this example only routes Internet traffic in-bound to the firewall and then to the web server, only one forwarding network address translation (NAT) rule is needed.

The forwarding rule accepts any inbound source address that hits the firewall trying to reach HTTP (port 80 or 443 for HTTPS). It's sent out of the firewall's local interface and redirected to the web server with the IP Address of 10.0.1.5.

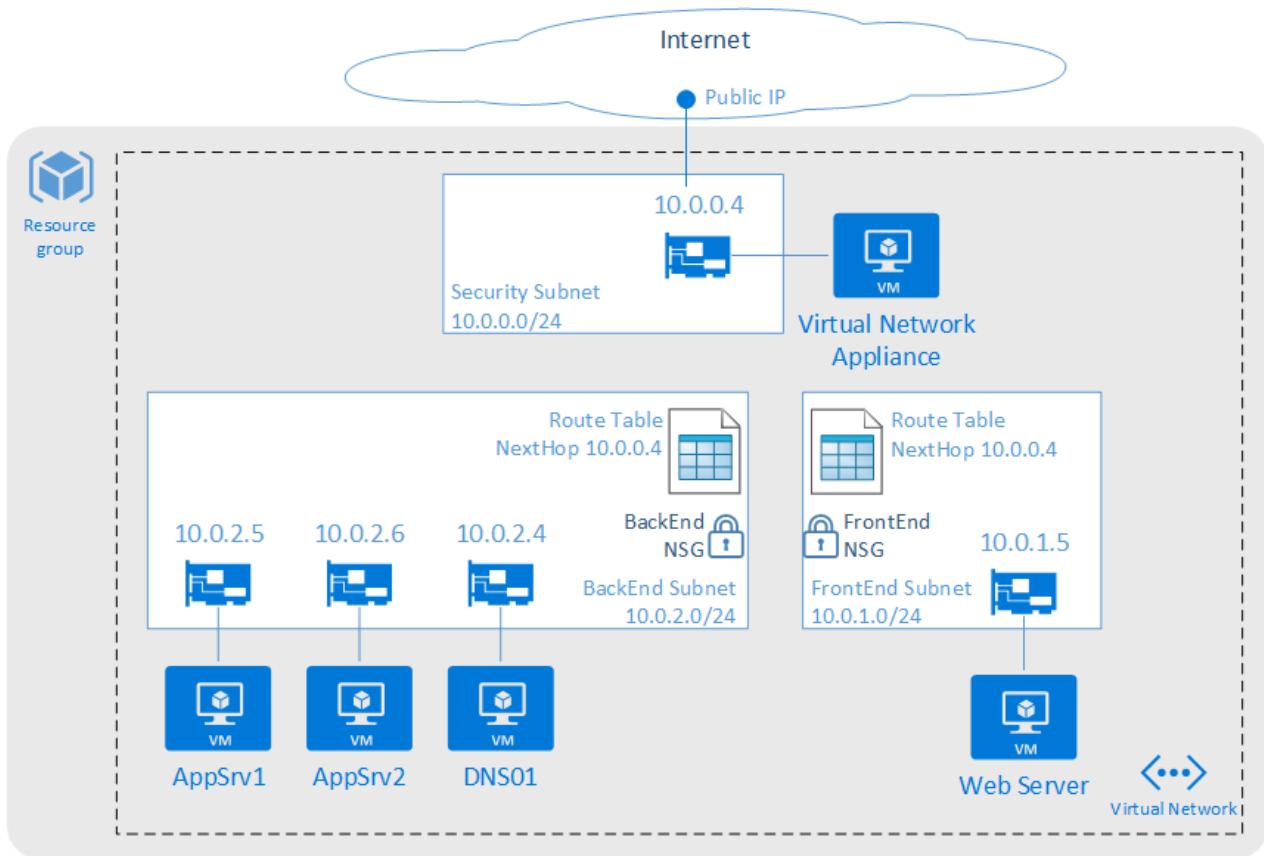
Conclusion

This example is a relatively straightforward way of protecting your application with a firewall and isolating the back-end subnet from inbound traffic. For more information, see the [detailed build instructions](#). These instructions include:

- How to build this perimeter network with classic PowerShell scripts.
- How to build this example with an Azure Resource Manager template.
- Detailed descriptions of each NSG command and firewall rule.
- Detailed traffic flow scenarios, showing how traffic is allowed or denied in each layer.

Example 3 Build a perimeter network to help protect networks with a firewall and UDR and NSG

[Back to Fast start](#) | [Detailed build instructions for this example](#)



Environment description

In this example, there is a subscription that contains the following resources:

- A single resource group
- A virtual network with three subnets: "SecNet", "FrontEnd", and "BackEnd"
- A network virtual appliance, in this case a firewall, connected to the SecNet subnet
- A Windows server that represents an application web server ("IIS01")
- Two Windows servers that represent application back-end servers ("AppVM01", "AppVM02")
- A Windows server that represents a DNS server ("DNS01")

For scripts and an Azure Resource Manager template, see the [detailed build instructions](#).

UDR description

By default, the following system routes are defined as:

Effective routes :					
Address	Prefix	Next hop type	Next hop IP address	Status	Source
{10.0.0.0/16}		VNETLocal		Active	Default
{0.0.0.0/0}		Internet		Active	Default
{10.0.0.0/8}		Null		Active	Default
{100.64.0.0/10}		Null		Active	Default
{172.16.0.0/12}		Null		Active	Default
{192.168.0.0/16}		Null		Active	Default

The VNETLocal is always one or more defined address prefixes that make up the virtual network for that specific network (that is, it changes from virtual network to virtual network, depending on how each specific virtual network is defined). The remaining system routes are static and default as indicated in the table.

In this example, two routing tables are created, one each for the front-end and back-end subnets. Each table is loaded with static routes appropriate for the given subnet. In this example, each table has three routes that direct all traffic (0.0.0.0/0) through the firewall (Next hop = Virtual Appliance IP address):

1. Local subnet traffic with no Next Hop defined to allow local subnet traffic to bypass the firewall.
2. Virtual network traffic with a Next Hop defined as firewall. This next hop overrides the default rule that allows local virtual network traffic to route directly.
3. All remaining traffic (0/0) with a Next Hop defined as the firewall.

TIP

Not having the local subnet entry in the UDR breaks local subnet communications.

- In our example, 10.0.1.0/24 pointing to VNETLocal is critical! Without it, packet leaving the Web Server (10.0.1.4) destined to another local server (for example) 10.0.1.25 will fail as they will be sent to the NVA. The NVA will send it to the subnet, and the subnet will resend it to the NVA in an infinite loop.
- The chances of a routing loop are typically higher on appliances with multiple NICs that are connected to separate subnets, which is often of traditional, on-premises appliances.

Once the routing tables are created, they must be bound to their subnets. The front-end subnet routing table, once created and bound to the subnet, would look like this output:

Effective routes :					
Address	Prefix	Next hop type	Next hop IP address	Status	Source
{10.0.1.0/24}		VNETLocal		Active	
{10.0.0.0/16}		VirtualAppliance	10.0.0.4	Active	
{0.0.0.0/0}		VirtualAppliance	10.0.0.4	Active	

NOTE

UDR can now be applied to the gateway subnet on which the ExpressRoute circuit is connected.

Examples of how to enable your perimeter network with ExpressRoute or site-to-site networking are shown in examples 3 and 4.

IP Forwarding description

IP Forwarding is a companion feature to UDR. IP Forwarding is a setting on a virtual appliance that allows it to receive traffic not specifically addressed to the appliance, and then forward that traffic to its ultimate destination.

For example, if AppVM01 makes a request to the DNS01 server, UDR would route this traffic to the firewall. With IP Forwarding enabled, the traffic for the DNS01 destination (10.0.2.4) is accepted by the appliance (10.0.0.4) and then forwarded to its ultimate destination (10.0.2.4). Without IP Forwarding enabled on the firewall, traffic would not be accepted by the appliance even though the route table has the firewall as the next hop. To use a virtual appliance, it's critical to remember to enable IP Forwarding along with UDR.

NSG description

In this example, an NSG group is built and then loaded with a single rule. This group is then bound only to the front-end and back-end subnets (not the SecNet). Declaratively the following rule is being built:

- Any traffic (all ports) from the Internet to the entire virtual network (all subnets) is denied.

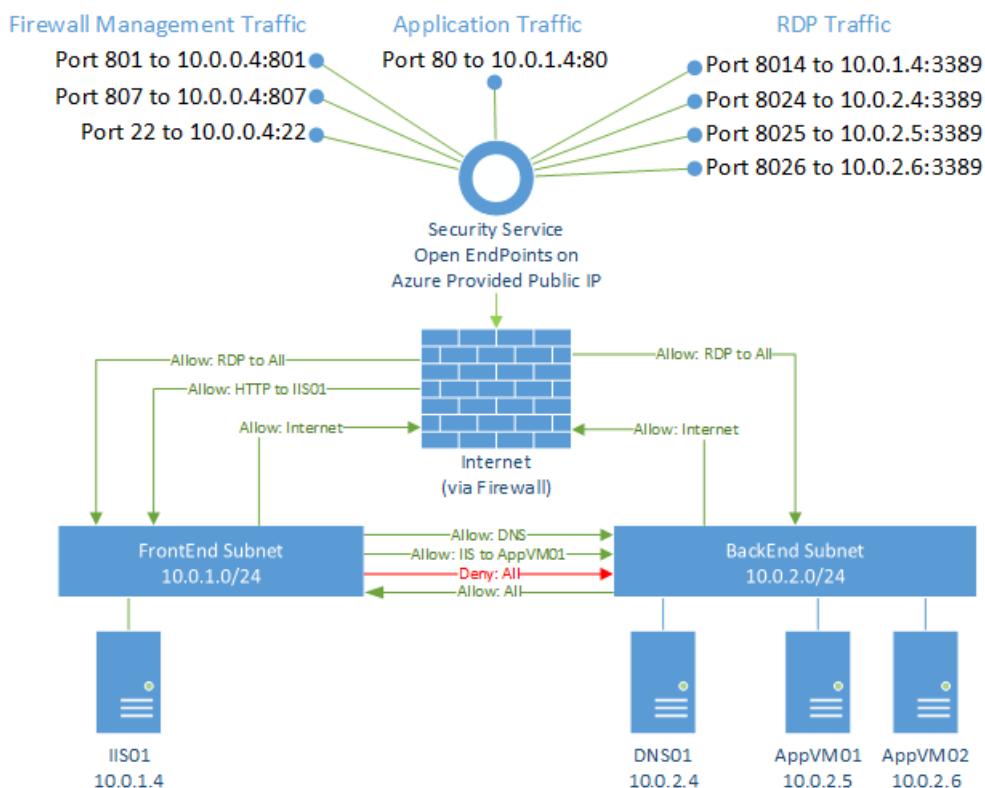
Although NSGs are used in this example, its main purpose is as a secondary layer of defense against manual misconfiguration. The goal is to block all inbound traffic from the Internet to either the front-end or back-end subnets. Traffic should only flow through the SecNet subnet to the firewall (and then, if appropriate, on to the front-end or back-end subnets). Plus, with the UDR rules in place, any traffic that did make it into the front-end or back-end subnets would be directed out to the firewall (thanks to UDR). The firewall would see this traffic as an asymmetric flow and would drop the outbound traffic. Thus there are three layers of security protecting the subnets:

- No Public IP addresses on any FrontEnd or BackEnd NICs.
- NSGs denying traffic from the Internet.
- The firewall dropping asymmetric traffic.

One interesting point regarding the NSG in this example is that it contains only one rule, which is to deny Internet traffic to the entire virtual network, including the Security subnet. However, since the NSG is only bound to the front-end and back-end subnets, the rule isn't processed on traffic inbound to the Security subnet. As a result, traffic flows to the Security subnet.

Firewall rules

On the firewall, forwarding rules should be created. Since the firewall is blocking or forwarding all inbound, outbound, and intra-virtual network traffic, many firewall rules are needed. Also, all inbound traffic hits the Security Service public IP address (on different ports), to be processed by the firewall. A best practice is to diagram the logical flows before setting up the subnets and firewall rules, to avoid rework later. The following figure is a logical view of the firewall rules for this example:



NOTE

Based on the Network Virtual Appliance used, the management ports vary. In this example, a Barracuda NextGen Firewall is referenced, which uses ports 22, 801, and 807. Consult the appliance vendor documentation to find the exact ports used for management of the device being used.

Firewall rules description

In the preceding logical diagram, the security subnet is not shown because the firewall is the only resource on that subnet. The diagram is showing the firewall rules and how they logically allow or deny traffic flows, not the actual routed path. Also, the external ports selected for the RDP traffic are higher ranged ports (8014 – 8026) and were selected to loosely align with the last two octets of the local IP address for easier readability (for example, local server address 10.0.1.4 is associated with external port 8014). Any higher non-conflicting ports, however, could be used.

For this example, we need seven types of rules:

- External rules (for inbound traffic):

1. Firewall management rule: This App Redirect rule allows traffic to pass to the management ports of the network virtual appliance.
 2. RDP rules (for each Windows server): These four rules (one for each server) allow management of the individual servers via RDP. The four RDP rules could also be collapsed into one rule, depending on the capabilities of the network virtual appliance being used.
 3. Application traffic rules: There are two of these rules, the first for the front-end web traffic, and the second for the back-end traffic (for example, web server to data tier). The configuration of these rules depends on the network architecture (where your servers are placed) and traffic flows (which direction the traffic flows, and which ports are used).
 - The first rule allows the actual application traffic to reach the application server. While the other rules allow for security and management, application traffic rules are what allow external users or services to access the applications. For this example, there is a single web server on port 80. Thus a single firewall application rule redirects inbound traffic to the external IP, to the web servers internal IP address. The redirected traffic session would be translated via NAT to the internal server.
 - The second rule is the back-end rule to allow the web server to talk to the AppVM01 server (but not AppVM02) via any port.
- Internal rules (for intra-virtual network traffic)
 1. Outbound to Internet rule: This rule allows traffic from any network to pass to the selected networks. This rule is usually a default rule already on the firewall, but in a disabled state. This rule should be enabled for this example.
 2. DNS rule: This rule allows only DNS (port 53) traffic to pass to the DNS server. For this environment, most traffic from the front end to the back end is blocked. This rule specifically allows DNS from any local subnet.
 3. Subnet to subnet rule: This rule is to allow any server on the back-end subnet to connect to any server on the front-end subnet (but not the reverse).
 - Fail-safe rule (for traffic that doesn't meet any of the previous):
 1. Deny all traffic rule: This deny rule should always be the final rule (in terms of priority), and as such if a traffic flow fails to match any of the preceding rules it is dropped by this rule. This rule is a default rule and usually in-place and active. No modifications are usually needed to this rule.

TIP

On the second application traffic rule, to simplify this example, any port is allowed. In a real scenario, the most specific port and address ranges should be used to reduce the attack surface of this rule.

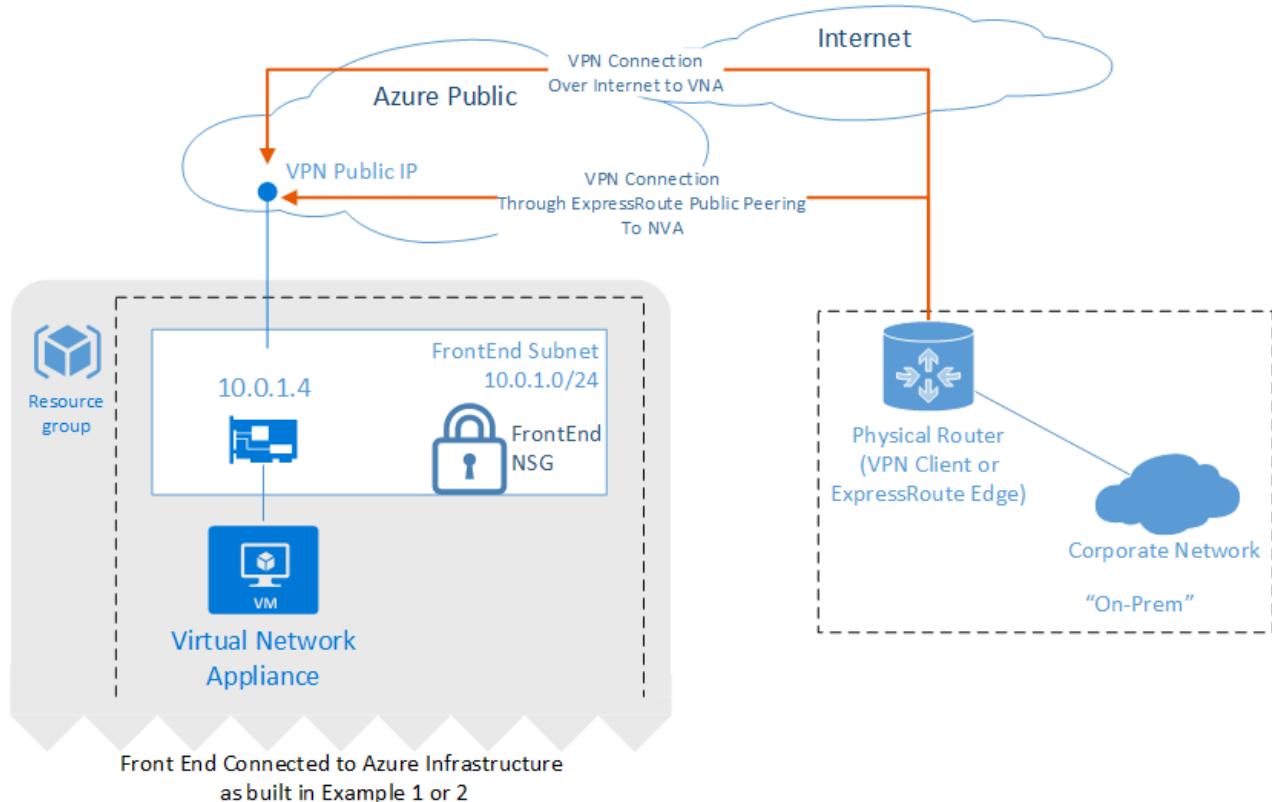
Once the previous rules are created, it's important to review the priority of each rule to ensure traffic is allowed or denied as desired. For this example, the rules are in priority order.

Conclusion

This example is a more complex but complete way of protecting and isolating the network than the previous examples. (Example 2 protects just the application, and Example 1 just isolates subnets). This design allows for monitoring traffic in both directions, and protects not just the inbound application server but enforces network security policy for all servers on this network. Also, depending on the appliance used, full traffic auditing and awareness can be achieved. For more information, see the [detailed build instructions](#). These instructions include:

- How to build this example perimeter network with classic PowerShell scripts.
- How to build this example with an Azure Resource Manager template.
- Detailed descriptions of each UDR, NSG command, and firewall rule.
- Detailed traffic flow scenarios, showing how traffic is allowed or denied in each layer.

Example 4 Add a hybrid connection with a site-to-site, virtual appliance VPN



Environment description

Hybrid networking using a network virtual appliance (NVA) can be added to any of the perimeter network types described in examples 1, 2, or 3.

As shown in the previous figure, a VPN connection over the Internet (site-to-site) is used to connect an on-premises network to an Azure virtual network via an NVA.

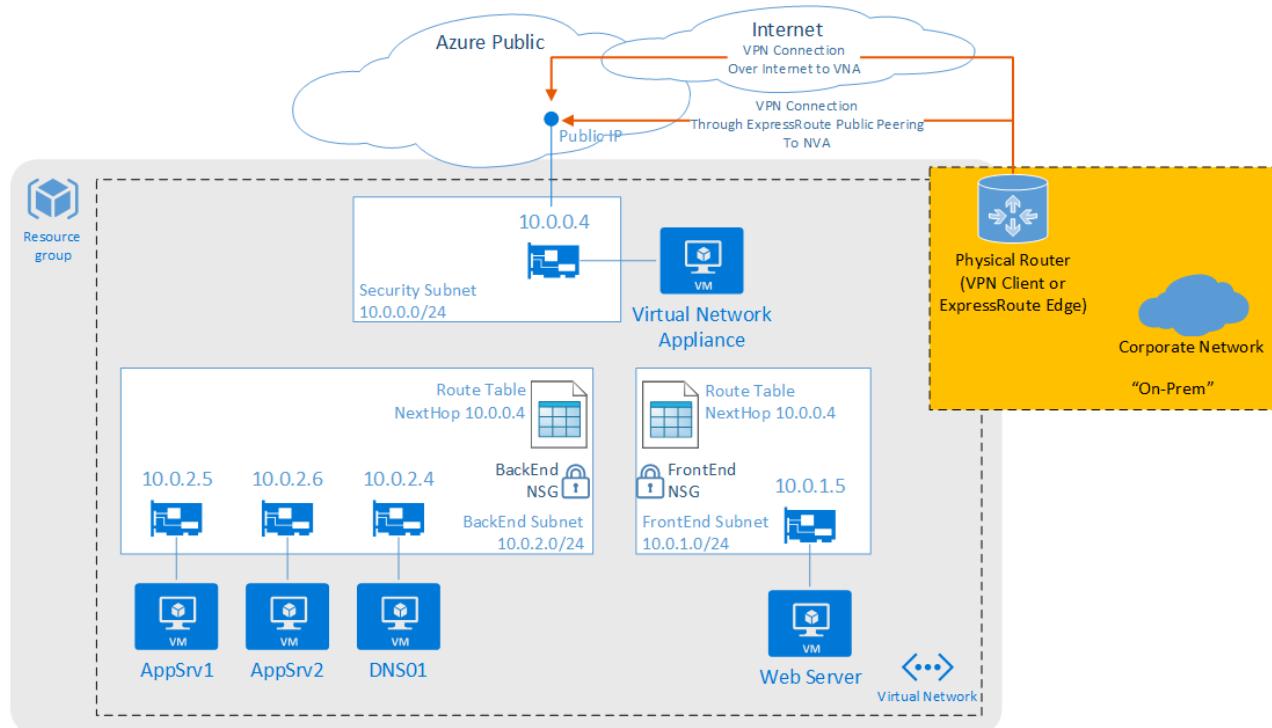
NOTE

If you use ExpressRoute with the Azure Public Peering option enabled, a static route should be created. This static route should route to the NVA VPN IP address out your corporate Internet and not via the ExpressRoute connection. The NAT required on the ExpressRoute Azure Public Peering option can break the VPN session.

Once the VPN is in place, the NVA becomes the central hub for all networks and subnets. The firewall forwarding rules determine which traffic flows are allowed, are translated via NAT, are redirected, or are dropped (even for traffic flows between the on-premises network and Azure).

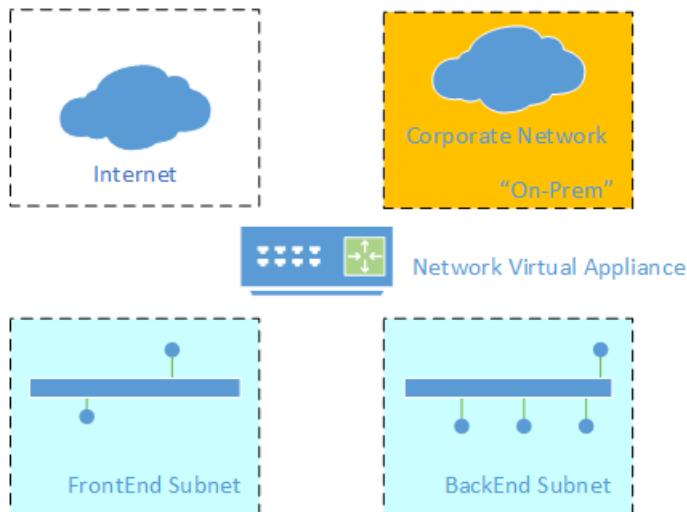
Traffic flows should be considered carefully, as they can be optimized or degraded by this design pattern, depending on the specific use case.

Using the environment built in example 3, and then adding a site-to-site VPN hybrid network connection, produces the following design:



The on-premises router, or any other network device that is compatible with your NVA for VPN, would be the VPN client. This physical device would be responsible for initiating and maintaining the VPN connection with your NVA.

Logically to the NVA, the network looks like four separate “security zones” with the rules on the NVA being the primary director of traffic between these zones:



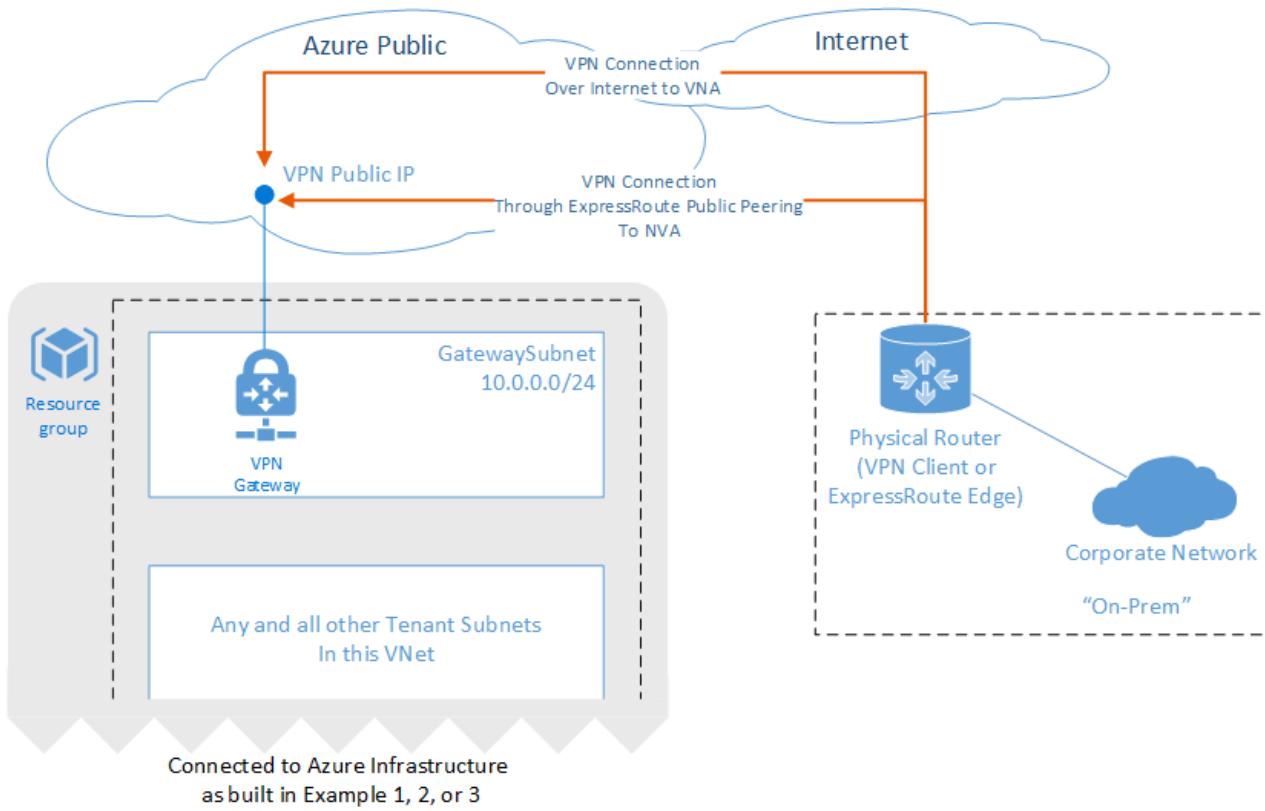
Conclusion

The addition of a site-to-site VPN hybrid network connection to an Azure virtual network can extend the on-premises network into Azure in a secure manner. In using a VPN connection, your traffic is encrypted and routes via the Internet. The NVA in this example provides a central location to enforce and manage the security policy. For more information, see the detailed build instructions (forthcoming). These instructions include:

- How to build this example perimeter network with PowerShell scripts.
- How to build this example with an Azure Resource Manager template.
- Detailed traffic flow scenarios, showing how traffic flows through this design.

Example 5 Add a hybrid connection with a site-to-site, Azure VPN gateway

[Back to Fast start](#) | Detailed build instructions available soon



Environment description

Hybrid networking using an Azure VPN gateway can be added to either perimeter network type described in examples 1 or 2.

As shown in the preceding figure, a VPN connection over the Internet (site-to-site) is used to connect an on-premises network to an Azure virtual network via an Azure VPN gateway.

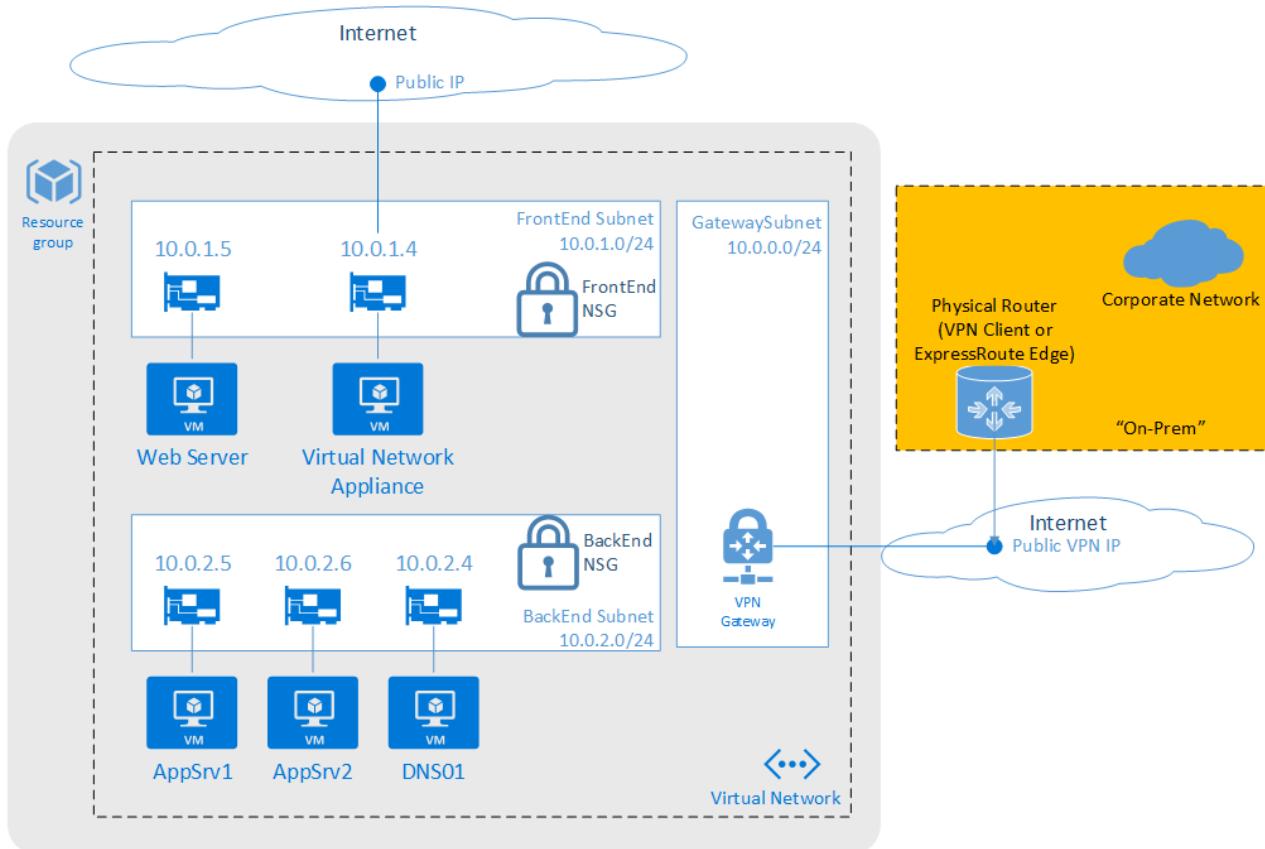
NOTE

If you use ExpressRoute with the Azure Public Peering option enabled, a static route should be created. This static route should route to the NVA VPN IP address out your corporate Internet and not via the ExpressRoute WAN. The NAT required on the ExpressRoute Azure Public Peering option can break the VPN session.

The following figure shows the two network edges in this example. On the first edge, the NVA and NSGs control traffic flows for intra-Azure networks and between Azure and the Internet. The second edge is the Azure VPN gateway, which is a separate and isolated network edge between on-premises and Azure.

Traffic flows should be considered carefully, as they can be optimized or degraded by this design pattern, depending on the specific use case.

Using the environment built in example 1, and then adding a site-to-site VPN hybrid network connection, produces the following design:



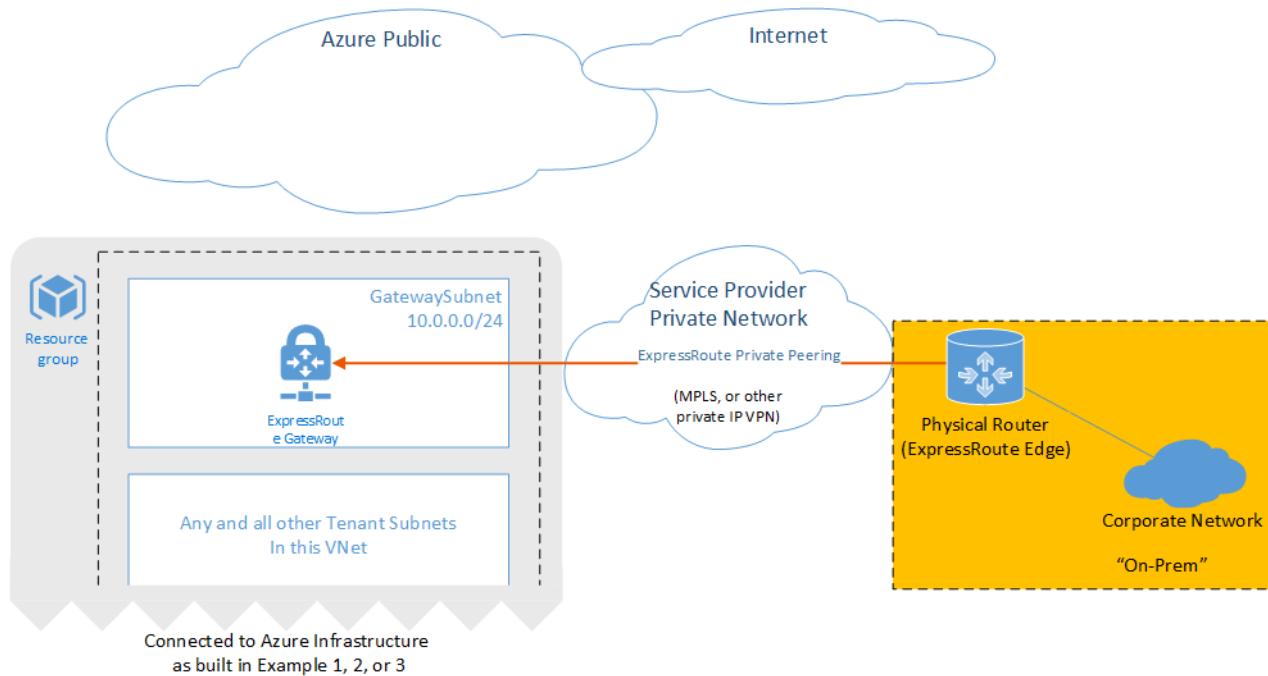
Conclusion

The addition of a site-to-site VPN hybrid network connection to an Azure virtual network can extend the on-premises network into Azure in a secure manner. Using the native Azure VPN gateway, your traffic is IPSec encrypted and routes via the Internet. Also, using the Azure VPN gateway can provide a lower-cost option (no additional licensing cost as with third-party NVAs). This option is most economical in example 1, where no NVA is used. For more information, see the detailed build instructions (forthcoming). These instructions include:

- How to build this example perimeter network with PowerShell scripts.
- How to build this example with an Azure Resource Manager template.
- Detailed traffic flow scenarios, showing how traffic flows through this design.

Example 6 Add a hybrid connection with ExpressRoute

[Back to Fast start](#) | Detailed build instructions available soon



Environment description

Hybrid networking using an ExpressRoute private peering connection can be added to either perimeter network type described in examples 1 or 2.

As shown in the preceding figure, ExpressRoute private peering provides a direct connection between your on-premises network and the Azure virtual network. Traffic transits only the service provider network and the Microsoft Azure network, never touching the Internet.

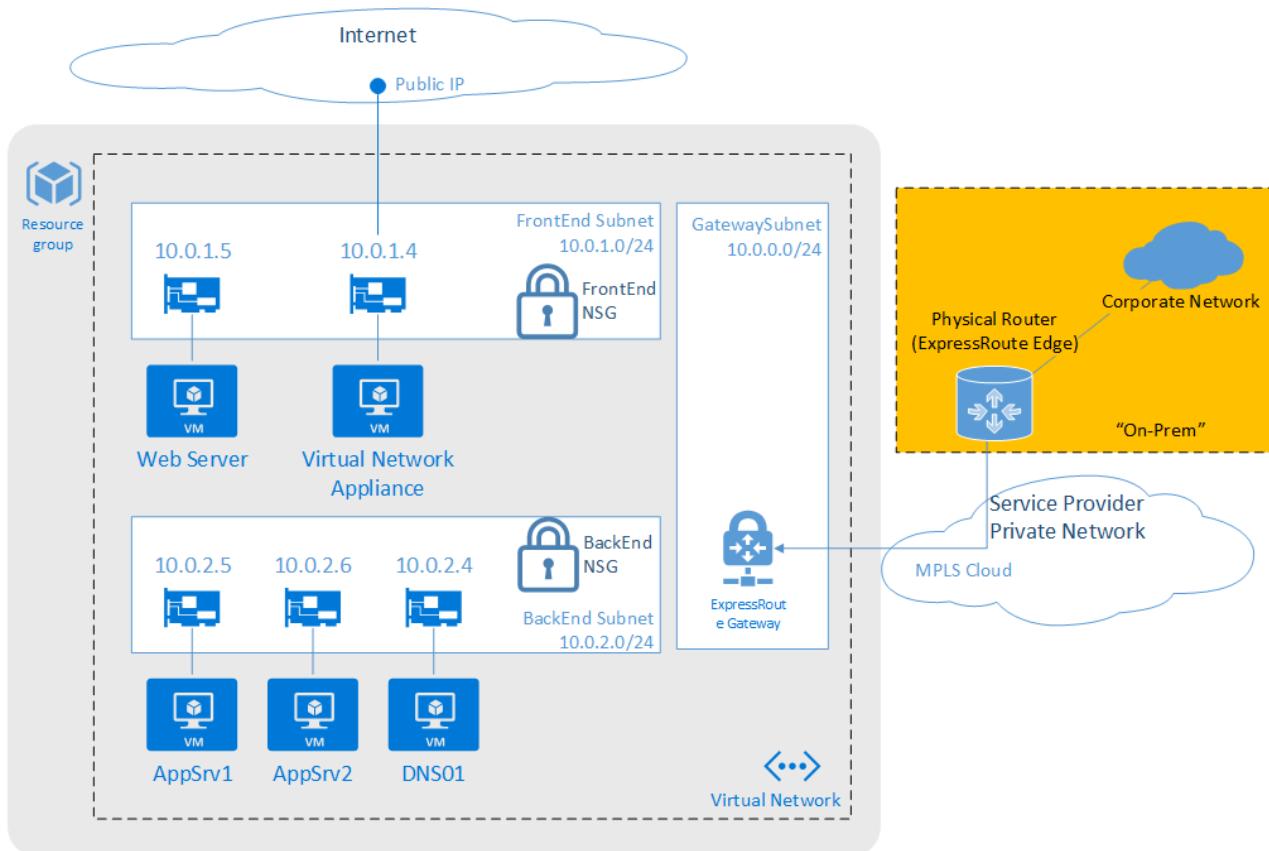
TIP

Using ExpressRoute keeps corporate network traffic off the Internet. It also allows for service level agreements from your ExpressRoute provider. The Azure Gateway can pass up to 10 Gbps with ExpressRoute, whereas with site-to-site VPNs, the Azure Gateway maximum throughput is 200 Mbps.

As seen in the following diagram, with this option the environment now has two network edges. The NVA and NSG control traffic flows for intra-Azure networks and between Azure and the Internet, while the gateway is a separate and isolated network edge between on-premises and Azure.

Traffic flows should be considered carefully, as they can be optimized or degraded by this design pattern, depending on the specific use case.

Using the environment built in example 1, and then adding an ExpressRoute hybrid network connection, produces the following design:



Conclusion

The addition of an ExpressRoute Private Peering network connection can extend the on-premises network into Azure in a secure, lower latency, higher performing manner. Also, using the native Azure Gateway, as in this example, provides a lower-cost option (no additional licensing as with third-party NVAs). For more information, see the detailed build instructions (forthcoming). These instructions include:

- How to build this example perimeter network with PowerShell scripts.
- How to build this example with an Azure Resource Manager template.

- Detailed traffic flow scenarios, showing how traffic flows through this design.

References

Helpful websites and documentation

- Access Azure with Azure Resource Manager:
- Accessing Azure with PowerShell: <https://docs.microsoft.com/powershell/azureps-cmdlets-docs/>
- Virtual networking documentation: <https://docs.microsoft.com/azure/virtual-network/>
- Network security group documentation: <https://docs.microsoft.com/azure/virtual-network/virtual-networks-nsg>
- User-defined routing documentation: <https://docs.microsoft.com/azure/virtual-network/virtual-networks-udr-overview>
- Azure virtual gateways: <https://docs.microsoft.com/azure/vpn-gateway/>
- Site-to-Site VPNs: <https://docs.microsoft.com/azure/vpn-gateway/vpn-gateway-create-site-to-site-rm-powershell>
- ExpressRoute documentation (be sure to check out the "Getting Started" and "How To" sections): <https://docs.microsoft.com/azure/expressroute/>

4 min to read •

Azure Data Security and Encryption Best Practices

6/27/2017 • 11 min to read • [Edit Online](#)

One of the keys to data protection in the cloud is accounting for the possible states in which your data may occur, and what controls are available for that state. For the purpose of Azure data security and encryption best practices the recommendations will be around the following data's states:

- At-rest: This includes all information storage objects, containers, and types that exist statically on physical media, be it magnetic or optical disk.
- In-Transit: When data is being transferred between components, locations or programs, such as over the network, across a service bus (from on-premises to cloud and vice-versa, including hybrid connections such as ExpressRoute), or during an input/output process, it is thought of as being in-motion.

In this article we will discuss a collection of Azure data security and encryption best practices. These best practices are derived from our experience with Azure data security and encryption and the experiences of customers like yourself.

For each best practice, we'll explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

This Azure Data Security and Encryption Best Practices article is based on a consensus opinion, and Azure platform capabilities and feature sets, as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure data security and encryption best practices discussed in this article include:

- Enforce multi-factor authentication
- Use role based access control (RBAC)
- Encrypt Azure virtual machines
- Use hardware security models
- Manage with Secure Workstations
- Enable SQL data encryption
- Protect data in transit
- Enforce file level data encryption

Enforce Multi-factor Authentication

The first step in data access and control in Microsoft Azure is to authenticate the user. [Azure Multi-Factor Authentication \(MFA\)](#) is a method of verifying user's identity by using another method than just a username and password. This authentication method helps safeguard access to data and applications while meeting user demand for a simple sign-in process.

By enabling Azure MFA for your users, you are adding a second layer of security to user sign-ins and transactions. In this case, a transaction might be accessing a document located in a file server or in your SharePoint Online. Azure MFA also helps IT to reduce the likelihood that a compromised credential will have access to organization's data.

For example: if you enforce Azure MFA for your users and configure it to use a phone call or text message as verification, if the user's credential is compromised, the attacker won't be able to access any resource since he will not have access to user's phone. Organizations that do not add this extra layer of identity protection are more susceptible for credential theft attack, which may lead to data compromise.

One alternative for organizations that want to keep the authentication control on-premises is to use [Azure Multi-Factor Authentication Server](#), also called MFA on-premises. By using this method you will still be able to enforce multi-factor authentication, while keeping the MFA server on-premises.

For more information on Azure MFA, please read the article [Getting started with Azure Multi-Factor Authentication in the cloud](#).

Use Role Based Access Control (RBAC)

Restrict access based on the [need to know](#) and [least privilege](#) security principles. This is imperative for organizations that want to enforce security policies for data access. Azure Role-Based Access Control (RBAC) can be used to assign permissions to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource.

You can leverage [built-in RBAC roles](#) in Azure to assign privileges to users. Consider using *Storage Account Contributor* for cloud operators that need to manage storage accounts and *Classic Storage Account Contributor* role to manage classic storage accounts. For cloud operators that needs to manage VMs and storage account, consider adding them to *Virtual Machine Contributor* role.

Organizations that do not enforce data access control by leveraging capabilities such as RBAC may be giving more privileges than necessary for their users. This can lead to data compromise by having some users having access to data that they shouldn't have in the first place.

You can learn more about Azure RBAC by reading the article [Azure Role-Based Access Control](#).

Encrypt Azure Virtual Machines

For many organizations, [data encryption at rest](#) is a mandatory step towards data privacy, compliance and data sovereignty. Azure Disk Encryption enables IT administrators to encrypt Windows and Linux IaaS Virtual Machine (VM) disks. Azure Disk Encryption leverages the industry standard BitLocker feature of Windows and the DM-Crypt feature of Linux to provide volume encryption for the OS and the data disks.

You can leverage Azure Disk Encryption to help protect and safeguard your data to meet your organizational security and compliance requirements. Organizations should also consider using encryption to help mitigate risks related to unauthorized data access. It is also recommended that you encrypt drives prior to writing sensitive data to them.

Make sure to encrypt your VM's data volumes and boot volume in order to protect data at rest in your Azure storage account. Safeguard the encryption keys and secrets by leveraging [Azure Key Vault](#).

For your on-premises Windows Servers, consider the following encryption best practices:

- Use [BitLocker](#) for data encryption
- Store recovery information in AD DS.
- If there is any concern that BitLocker keys have been compromised, we recommend that you either format the drive to remove all instances of the BitLocker metadata from the drive or that you decrypt and encrypt the entire drive again.

Organizations that do not enforce data encryption are more likely to be exposed to data integrity issues, such as malicious or rogue users stealing data and compromised accounts gaining unauthorized access to data in clear format. Besides these risks, companies that have to comply with industry regulations, must prove that they are

diligent and are using the correct security controls to enhance data security.

You can learn more about Azure Disk Encryption by reading the article [Azure Disk Encryption for Windows and Linux IaaS VMs](#).

Use Hardware Security Modules

Industry encryption solutions use secret keys to encrypt data. Therefore, it is critical that these keys are safely stored. Key management becomes an integral part of data protection, since it will be leveraged to store secret keys that are used to encrypt data.

Azure disk encryption uses [Azure Key Vault](#) to help you control and manage disk encryption keys and secrets in your key vault subscription, while ensuring that all data in the virtual machine disks are encrypted at rest in your Azure storage. You should use Azure Key Vault to audit keys and policy usage.

There are many inherent risks related to not having appropriate security controls in place to protect the secret keys that were used to encrypt your data. If attackers have access to the secret keys, they will be able to decrypt the data and potentially have access to confidential information.

You can learn more about general recommendations for certificate management in Azure by reading the article [Certificate Management in Azure: Do's and Don'ts](#).

For more information about Azure Key Vault, read [Get started with Azure Key Vault](#).

Manage with Secure Workstations

Since the vast majority of the attacks target the end user, the endpoint becomes one of the primary points of attack. If an attacker compromises the endpoint, he can leverage the user's credentials to gain access to organization's data. Most endpoint attacks are able to take advantage of the fact that end users are administrators in their local workstations.

You can reduce these risks by using a secure management workstation. We recommend that you use a [Privileged Access Workstations \(PAW\)](#) to reduce the attack surface in workstations. These secure management workstations can help you mitigate some of these attacks help ensure your data is safer. Make sure to use PAW to harden and lock down your workstation. This is an important step to provide high security assurances for sensitive accounts, tasks and data protection.

Lack of endpoint protection may put your data at risk, make sure to enforce security policies across all devices that are used to consume data, regardless of the data location (cloud or on-premises).

You can learn more about privileged access workstation by reading the article [Securing Privileged Access](#).

Enable SQL data encryption

[Azure SQL Database transparent data encryption](#) (TDE) helps protect against the threat of malicious activity by performing real-time encryption and decryption of the database, associated backups, and transaction log files at rest without requiring changes to the application. TDE encrypts the storage of an entire database by using a symmetric key called the database encryption key.

Even when the entire storage is encrypted, it is very important to also encrypt your database itself. This is an implementation of the defense in depth approach for data protection. If you are using [Azure SQL Database](#) and wish to protect sensitive data such as credit card or social security numbers, you can encrypt databases with FIPS 140-2 validated 256 bit AES encryption which meets the requirements of many industry standards (e.g., HIPAA, PCI).

It's important to understand that files related to [buffer pool extension](#) (BPE) are not encrypted when a database is encrypted using TDE. You must use file system level encryption tools like BitLocker or the [Encrypting File System](#)

(EFS) for BPE related files.

Since an authorized user such as a security administrator or a database administrator can access the data even if the database is encrypted with TDE, you should also follow the recommendations below:

- SQL authentication at the database level
- Azure AD authentication using RBAC roles
- Users and applications should use separate accounts to authenticate. This way you can limit the permissions granted to users and applications and reduce the risks of malicious activity
- Implement database-level security by using fixed database roles (such as db_datareader or db_datawriter), or you can create custom roles for your application to grant explicit permissions to selected database objects

Organizations that are not using database level encryption may be more susceptible for attacks that may compromise data located in SQL databases.

You can learn more about SQL TDE encryption by reading the article [Transparent Data Encryption with Azure SQL Database](#).

Protect data in transit

Protecting data in transit should be essential part of your data protection strategy. Since data will be moving back and forth from many locations, the general recommendation is that you always use SSL/TLS protocols to exchange data across different locations. In some circumstances, you may want to isolate the entire communication channel between your on-premises and cloud infrastructure by using a virtual private network (VPN).

For data moving between your on-premises infrastructure and Azure, you should consider appropriate safeguards such as HTTPS or VPN.

For organizations that need to secure access from multiple workstations located on-premises to Azure, use [Azure site-to-site VPN](#).

For organizations that need to secure access from one workstation located on-premises to Azure, use [Point-to-Site VPN](#).

Larger data sets can be moved over a dedicated high-speed WAN link such as [ExpressRoute](#). If you choose to use ExpressRoute, you can also encrypt the data at the application-level using [SSL/TLS](#) or other protocols for added protection.

If you are interacting with Azure Storage through the Azure Portal, all transactions occur via HTTPS. [Storage REST API](#) over HTTPS can also be used to interact with [Azure Storage](#) and [Azure SQL Database](#).

Organizations that fail to protect data in transit are more susceptible for [man-in-the-middle attacks](#), [eavesdropping](#) and session hijacking. These attacks can be the first step in gaining access to confidential data.

You can learn more about Azure VPN option by reading the article [Planning and design for VPN Gateway](#).

Enforce file level data encryption

Another layer of protection that can increase the level of security for your data is encrypting the file itself, regardless of the file location.

[Azure RMS](#) uses encryption, identity, and authorization policies to help secure your files and email. Azure RMS works across multiple devices — phones, tablets, and PCs by protecting both within your organization and outside your organization. This capability is possible because Azure RMS adds a level of protection that remains with the data, even when it leaves your organization's boundaries.

When you use Azure RMS to protect your files, you are using industry-standard cryptography with full support of [FIPS 140-2](#). When you leverage Azure RMS for data protection, you have the assurance that the protection stays

with the file, even if it is copied to storage that is not under the control of IT, such as a cloud storage service. The same occurs for files shared via e-mail, the file is protected as an attachment to an email message, with instructions how to open the protected attachment.

When planning for Azure RMS adoption we recommend the following:

- Install the [RMS sharing app](#). This app integrates with Office applications by installing an Office add-in so that users can easily protect files directly.
- Configure applications and services to support Azure RMS
- Create [custom templates](#) that reflect your business requirements. For example: a template for top secret data that should be applied in all top secret related emails.

Organizations that are weak on [data classification](#) and file protection may be more susceptible to data leakage. Without proper file protection, organizations won't be able to obtain business insights, monitor for abuse and prevent malicious access to files.

You can learn more about Azure RMS by reading the article [Getting Started with Azure Rights Management](#).

Azure Storage security guide

8/21/2017 • 44 min to read • [Edit Online](#)

Overview

Azure Storage provides a comprehensive set of security capabilities which together enable developers to build secure applications. The storage account itself can be secured using Role-Based Access Control and Azure Active Directory. Data can be secured in transit between an application and Azure by using [Client-Side Encryption](#), HTTPS, or SMB 3.0. Data can be set to be automatically encrypted when written to Azure Storage using [Storage Service Encryption \(SSE\)](#). OS and Data disks used by virtual machines can be set to be encrypted using [Azure Disk Encryption](#). Delegated access to the data objects in Azure Storage can be granted using [Shared Access Signatures](#).

This article will provide an overview of each of these security features that can be used with Azure Storage. Links are provided to articles that will give details of each feature so you can easily do further investigation on each topic.

Here are the topics to be covered in this article:

- [Management Plane Security](#) – Securing your Storage Account

The management plane consists of the resources used to manage your storage account. In this section, we'll talk about the Azure Resource Manager deployment model and how to use Role-Based Access Control (RBAC) to control access to your storage accounts. We will also talk about managing your storage account keys and how to regenerate them.

- [Data Plane Security](#) – Securing Access to Your Data

In this section, we'll look at allowing access to the actual data objects in your Storage account, such as blobs, files, queues, and tables, using Shared Access Signatures and Stored Access Policies. We will cover both service-level SAS and account-level SAS. We'll also see how to limit access to a specific IP address (or range of IP addresses), how to limit the protocol used to HTTPS, and how to revoke a Shared Access Signature without waiting for it to expire.

- [Encryption in Transit](#)

This section discusses how to secure data when you transfer it into or out of Azure Storage. We'll talk about the recommended use of HTTPS and the encryption used by SMB 3.0 for Azure File shares. We will also take a look at Client-side Encryption, which enables you to encrypt the data before it is transferred into Storage in a client application, and to decrypt the data after it is transferred out of Storage.

- [Encryption at Rest](#)

We will talk about Storage Service Encryption (SSE), and how you can enable it for a storage account, resulting in your block blobs, page blobs, and append blobs being automatically encrypted when written to Azure Storage. We will also look at how you can use Azure Disk Encryption and explore the basic differences and cases of Disk Encryption versus SSE versus Client-Side Encryption. We will briefly look at FIPS compliance for U.S. Government computers.

- Using [Storage Analytics](#) to audit access of Azure Storage

This section discusses how to find information in the storage analytics logs for a request. We'll take a look at real storage analytics log data and see how to discern whether a request is made with the Storage account key, with a Shared Access signature, or anonymously, and whether it succeeded or failed.

- [Enabling Browser-Based Clients using CORS](#)

This section talks about how to allow cross-origin resource sharing (CORS). We'll talk about cross-domain access, and how to handle it with the CORS capabilities built into Azure Storage.

Management Plane Security

The management plane consists of operations that affect the storage account itself. For example, you can create or delete a storage account, get a list of storage accounts in a subscription, retrieve the storage account keys, or regenerate the storage account keys.

When you create a new storage account, you select a deployment model of Classic or Resource Manager. The Classic model of creating resources in Azure only allows all-or-nothing access to the subscription, and in turn, the storage account.

This guide focuses on the Resource Manager model which is the recommended means for creating storage accounts. With the Resource Manager storage accounts, rather than giving access to the entire subscription, you can control access on a more finite level to the management plane using Role-Based Access Control (RBAC).

How to secure your storage account with Role-Based Access Control (RBAC)

Let's talk about what RBAC is, and how you can use it. Each Azure subscription has an Azure Active Directory. Users, groups, and applications from that directory can be granted access to manage resources in the Azure subscription that use the Resource Manager deployment model. This is referred to as Role-Based Access Control (RBAC). To manage this access, you can use the [Azure portal](#), the [Azure CLI tools](#), [PowerShell](#), or the [Azure Storage Resource Provider REST APIs](#).

With the Resource Manager model, you put the storage account in a resource group and control access to the management plane of that specific storage account using Azure Active Directory. For example, you can give specific users the ability to access the storage account keys, while other users can view information about the storage account, but cannot access the storage account keys.

Granting Access

Access is granted by assigning the appropriate RBAC role to users, groups, and applications, at the right scope. To grant access to the entire subscription, you assign a role at the subscription level. You can grant access to all of the resources in a resource group by granting permissions to the resource group itself. You can also assign specific roles to specific resources, such as storage accounts.

Here are the main points that you need to know about using RBAC to access the management operations of an Azure Storage account:

- When you assign access, you basically assign a role to the account that you want to have access. You can control access to the operations used to manage that storage account, but not to the data objects in the account. For example, you can grant permission to retrieve the properties of the storage account (such as redundancy), but not to a container or data within a container inside Blob Storage.
- For someone to have permission to access the data objects in the storage account, you can give them permission to read the storage account keys, and that user can then use those keys to access the blobs, queues, tables, and files.
- Roles can be assigned to a specific user account, a group of users, or to a specific application.
- Each role has a list of Actions and Not Actions. For example, the Virtual Machine Contributor role has an Action of "listKeys" that allows the storage account keys to be read. The Contributor has "Not Actions" such as updating the access for users in the Active Directory.
- Roles for storage include (but are not limited to) the following:
 - Owner – They can manage everything, including access.
 - Contributor – They can do anything the owner can do except assign access. Someone with this role can

view and regenerate the storage account keys. With the storage account keys, they can access the data objects.

- Reader – They can view information about the storage account, except secrets. For example, if you assign a role with reader permissions on the storage account to someone, they can view the properties of the storage account, but they can't make any changes to the properties or view the storage account keys.
- Storage Account Contributor – They can manage the storage account – they can read the subscription's resource groups and resources, and create and manage subscription resource group deployments. They can also access the storage account keys, which in turn means they can access the data plane.
- User Access Administrator – They can manage user access to the storage account. For example, they can grant Reader access to a specific user.
- Virtual Machine Contributor – They can manage virtual machines but not the storage account to which they are connected. This role can list the storage account keys, which means that the user to whom you assign this role can update the data plane.

In order for a user to create a virtual machine, they have to be able to create the corresponding VHD file in a storage account. To do that, they need to be able to retrieve the storage account key and pass it to the API creating the VM. Therefore, they must have this permission so they can list the storage account keys.

- The ability to define custom roles is a feature that allows you to compose a set of actions from a list of available actions that can be performed on Azure resources.
- The user has to be set up in your Azure Active Directory before you can assign a role to them.
- You can create a report of who granted/revoked what kind of access to/from whom and on what scope using PowerShell or the Azure CLI.

Resources

- [Azure Active Directory Role-based Access Control](#)

This article explains the Azure Active Directory Role-based Access Control and how it works.

- [RBAC: Built in Roles](#)

This article details all of the built-in roles available in RBAC.

- [Understanding Resource Manager deployment and classic deployment](#)

This article explains the Resource Manager deployment and classic deployment models, and explains the benefits of using the Resource Manager and resource groups. It explains how the Azure Compute, Network, and Storage Providers work under the Resource Manager model.

- [Managing Role-Based Access Control with the REST API](#)

This article shows how to use the REST API to manage RBAC.

- [Azure Storage Resource Provider REST API Reference](#)

This is the reference for the APIs you can use to manage your storage account programmatically.

- [Developer's guide to auth with Azure Resource Manager API](#)

This article shows how to authenticate using the Resource Manager APIs.

- [Role-Based Access Control for Microsoft Azure from Ignite](#)

This is a link to a video on Channel 9 from the 2015 MS Ignite conference. In this session, they talk about access management and reporting capabilities in Azure, and explore best practices around securing access to Azure subscriptions using Azure Active Directory.

Managing Your Storage Account Keys

Storage account keys are 512-bit strings created by Azure that, along with the storage account name, can be used to access the data objects stored in the storage account, e.g. blobs, entities within a table, queue messages, and files on an Azure File share. Controlling access to the storage account keys controls access to the data plane for that storage account.

Each storage account has two keys referred to as "Key 1" and "Key 2" in the [Azure portal](#) and in the PowerShell cmdlets. These can be regenerated manually using one of several methods, including, but not limited to using the [Azure portal](#), PowerShell, the Azure CLI, or programmatically using the .NET Storage Client Library or the Azure Storage Services REST API.

There are any number of reasons to regenerate your storage account keys.

- You might regenerate them on a regular basis for security reasons.
- You would regenerate your storage account keys if someone managed to hack into an application and retrieve the key that was hardcoded or saved in a configuration file, giving them full access to your storage account.
- Another case for key regeneration is if your team is using a Storage Explorer application that retains the storage account key, and one of the team members leaves. The application would continue to work, giving them access to your storage account after they're gone. This is actually the primary reason they created account-level Shared Access Signatures – you can use an account-level SAS instead of storing the access keys in a configuration file.

Key regeneration plan

You don't want to just regenerate the key you are using without some planning. If you do that, you could cut off all access to that storage account, which can cause major disruption. This is why there are two keys. You should regenerate one key at a time.

Before you regenerate your keys, be sure you have a list of all of your applications that are dependent on the storage account, as well as any other services you are using in Azure. For example, if you are using Azure Media Services that are dependent on your storage account, you must re-sync the access keys with your media service after you regenerate the key. If you are using any applications such as a storage explorer, you will need to provide the new keys to those applications as well. Note that if you have VMs whose VHD files are stored in the storage account, they will not be affected by regenerating the storage account keys.

You can regenerate your keys in the Azure portal. Once keys are regenerated they can take up to 10 minutes to be synchronized across Storage Services.

When you're ready, here's the general process detailing how you should change your key. In this case, the assumption is that you are currently using Key 1 and you are going to change everything to use Key 2 instead.

1. Regenerate Key 2 to ensure that it is secure. You can do this in the Azure portal.
2. In all of the applications where the storage key is stored, change the storage key to use Key 2's new value. Test and publish the application.
3. After all of the applications and services are up and running successfully, regenerate Key 1. This ensures that anybody to whom you have not expressly given the new key will no longer have access to the storage account.

If you are currently using Key 2, you can use the same process, but reverse the key names.

You can migrate over a couple of days, changing each application to use the new key and publishing it. After all of them are done, you should then go back and regenerate the old key so it no longer works.

Another option is to put the storage account key in an [Azure Key Vault](#) as a secret and have your applications retrieve the key from there. Then when you regenerate the key and update the Azure Key Vault, the applications will not need to be redeployed because they will pick up the new key from the Azure Key Vault automatically. Note that you can have the application read the key each time you need it, or you can cache it in memory and if it fails when using it, retrieve the key again from the Azure Key Vault.

Using Azure Key Vault also adds another level of security for your storage keys. If you use this method, you will

never have the storage key hardcoded in a configuration file, which removes that avenue of somebody getting access to the keys without specific permission.

Another advantage of using Azure Key Vault is you can also control access to your keys using Azure Active Directory. This means you can grant access to the handful of applications that need to retrieve the keys from Azure Key Vault, and know that other applications will not be able to access the keys without granting them permission specifically.

Note: it is recommended to use only one of the keys in all of your applications at the same time. If you use Key 1 in some places and Key 2 in others, you will not be able to rotate your keys without some application losing access.

Resources

- [About Azure Storage Accounts](#)

This article gives an overview of storage accounts and discusses viewing, copying, and regenerating storage access keys.

- [Azure Storage Resource Provider REST API Reference](#)

This article contains links to specific articles about retrieving the storage account keys and regenerating the storage account keys for an Azure Account using the REST API. Note: This is for Resource Manager storage accounts.

- [Operations on storage accounts](#)

This article in the Storage Service Manager REST API Reference contains links to specific articles on retrieving and regenerating the storage account keys using the REST API. Note: This is for the Classic storage accounts.

- [Say goodbye to key management – manage access to Azure Storage data using Azure AD](#)

This article shows how to use Active Directory to control access to your Azure Storage keys in Azure Key Vault. It also shows how to use an Azure Automation job to regenerate the keys on an hourly basis.

Data Plane Security

Data Plane Security refers to the methods used to secure the data objects stored in Azure Storage – the blobs, queues, tables, and files. We've seen methods to encrypt the data and security during transit of the data, but how do you go about allowing access to the objects?

There are basically two methods for controlling access to the data objects themselves. The first is by controlling access to the storage account keys, and the second is using Shared Access Signatures to grant access to specific data objects for a specific amount of time.

One exception to note is that you can allow public access to your blobs by setting the access level for the container that holds the blobs accordingly. If you set access for a container to Blob or Container, it will allow public read access for the blobs in that container. This means anyone with a URL pointing to a blob in that container can open it in a browser without using a Shared Access Signature or having the storage account keys.

Storage Account Keys

Storage account keys are 512-bit strings created by Azure that, along with the storage account name, can be used to access the data objects stored in the storage account.

For example, you can read blobs, write to queues, create tables, and modify files. Many of these actions can be performed through the Azure portal, or using one of many Storage Explorer applications. You can also write code to use the REST API or one of the Storage Client Libraries to perform these operations.

As discussed in the section on the [Management Plane Security](#), access to the storage keys for a Classic storage account can be granted by giving full access to the Azure subscription. Access to the storage keys for a storage

account using the Azure Resource Manager model can be controlled through Role-Based Access Control (RBAC).

How to delegate access to objects in your account using Shared Access Signatures and Stored Access Policies

A Shared Access Signature is a string containing a security token that can be attached to a URI that allows you to delegate access to storage objects and specify constraints such as the permissions and the date/time range of access.

You can grant access to blobs, containers, queue messages, files, and tables. With tables, you can actually grant permission to access a range of entities in the table by specifying the partition and row key ranges to which you want the user to have access. For example, if you have data stored with a partition key of geographical state, you could give someone access to just the data for California.

In another example, you might give a web application a SAS token that enables it to write entries to a queue, and give a worker role application a SAS token to get messages from the queue and process them. Or you could give one customer a SAS token they can use to upload pictures to a container in Blob Storage, and give a web application permission to read those pictures. In both cases, there is a separation of concerns – each application can be given just the access that they require in order to perform their task. This is possible through the use of Shared Access Signatures.

Why you want to use Shared Access Signatures

Why would you want to use an SAS instead of just giving out your storage account key, which is so much easier? Giving out your storage account key is like sharing the keys of your storage kingdom. It grants complete access. Someone could use your keys and upload their entire music library to your storage account. They could also replace your files with virus-infected versions, or steal your data. Giving away unlimited access to your storage account is something that should not be taken lightly.

With Shared Access Signatures, you can give a client just the permissions required for a limited amount of time. For example, if someone is uploading a blob to your account, you can grant them write access for just enough time to upload the blob (depending on the size of the blob, of course). And if you change your mind, you can revoke that access.

Additionally, you can specify that requests made using a SAS are restricted to a certain IP address or IP address range external to Azure. You can also require that requests are made using a specific protocol (HTTPS or HTTP/HTTPS). This means if you only want to allow HTTPS traffic, you can set the required protocol to HTTPS only, and HTTP traffic will be blocked.

Definition of a Shared Access Signature

A Shared Access Signature is a set of query parameters appended to the URL pointing at the resource

that provides information about the access allowed and the length of time for which the access is permitted. Here is an example; this URI provides read access to a blob for five minutes. Note that SAS query parameters must be URL Encoded, such as %3A for colon (:) or %20 for a space.

```
http://mystorage.blob.core.windows.net/mycontainer/myblob.txt (URL to the blob)
?sv=2015-04-05 (storage service version)
&st=2015-12-10T22%3A18%3A26Z (start time, in UTC time and URL encoded)
&se=2015-12-10T22%3A23%3A26Z (end time, in UTC time and URL encoded)
&sr=b (resource is a blob)
&sp=r (read access)
&sip=168.1.5.60-168.1.5.70 (requests can only come from this range of IP addresses)
&spr=https (only allow HTTPS requests)
&sig=Z%2FRHIX5Xcg0Mq2rqI301WTjEg2tYkboXr1P9ZUXDtkk%3D (signature used for the authentication of the SAS)
```

How the Shared Access Signature is authenticated by the Azure Storage Service

When the storage service receives the request, it takes the input query parameters and creates a signature using the same method as the calling program. It then compares the two signatures. If they agree, then the storage service can check the storage service version to make sure it's valid, verify that the current date and time are within

the specified window, make sure the access requested corresponds to the request made, etc.

For example, with our URL above, if the URL was pointing to a file instead of a blob, this request would fail because it specifies that the Shared Access Signature is for a blob. If the REST command being called was to update a blob, it would fail because the Shared Access Signature specifies that only read access is permitted.

Types of Shared Access Signatures

- A service-level SAS can be used to access specific resources in a storage account. Some examples of this are retrieving a list of blobs in a container, downloading a blob, updating an entity in a table, adding messages to a queue or uploading a file to a file share.
- An account-level SAS can be used to access anything that a service-level SAS can be used for. Additionally, it can give options to resources that are not permitted with a service-level SAS, such as the ability to create containers, tables, queues, and file shares. You can also specify access to multiple services at once. For example, you might give someone access to both blobs and files in your storage account.

Creating an SAS URI

1. You can create an ad hoc URI on demand, defining all of the query parameters each time.

This is really flexible, but if you have a logical set of parameters that are similar each time, using a Stored Access Policy is a better idea.

2. You can create a Stored Access Policy for an entire container, file share, table, or queue. Then you can use this as the basis for the SAS URIs you create. Permissions based on Stored Access Policies can be easily revoked. You can have up to 5 policies defined on each container, queue, table, or file share.

For example, if you were going to have many people read the blobs in a specific container, you could create a Stored Access Policy that says "give read access" and any other settings that will be the same each time.

Then you can create an SAS URI using the settings of the Stored Access Policy and specifying the expiration date/time. The advantage of this is that you don't have to specify all of the query parameters every time.

Revocation

Suppose your SAS has been compromised, or you want to change it because of corporate security or regulatory compliance requirements. How do you revoke access to a resource using that SAS? It depends on how you created the SAS URI.

If you are using ad hoc URI's, you have three options. You can issue SAS tokens with short expiration policies and simply wait for the SAS to expire. You can rename or delete the resource (assuming the token was scoped to a single object). You can change the storage account keys. This last option can have a big impact, depending on how many services are using that storage account, and probably isn't something you want to do without some planning.

If you are using a SAS derived from a Stored Access Policy, you can remove access by revoking the Stored Access Policy – you can just change it so it has already expired, or you can remove it altogether. This takes effect immediately, and invalidates every SAS created using that Stored Access Policy. Updating or removing the Stored Access Policy may impact people accessing that specific container, file share, table, or queue via SAS, but if the clients are written so they request a new SAS when the old one becomes invalid, this will work fine.

Because using a SAS derived from a Stored Access Policy gives you the ability to revoke that SAS immediately, it is the recommended best practice to always use Stored Access Policies when possible.

Resources

For more detailed information on using Shared Access Signatures and Stored Access Policies, complete with examples, please refer to the following articles:

- These are the reference articles.
 - [Service SAS](#)

This article provides examples of using a service-level SAS with blobs, queue messages, table ranges, and files.

- [Constructing a service SAS](#)
- [Constructing an account SAS](#)
- These are tutorials for using the .NET client library to create Shared Access Signatures and Stored Access Policies.
 - [Using Shared Access Signatures \(SAS\)](#)
 - [Shared Access Signatures, Part 2: Create and Use a SAS with the Blob Service](#)

This article includes an explanation of the SAS model, examples of Shared Access Signatures, and recommendations for the best practice use of SAS. Also discussed is the revocation of the permission granted.

- Limiting access by IP Address (IP ACLs)
 - [What is an endpoint Access Control List \(ACLs\)?](#)
 - [Constructing a Service SAS](#)

This is the reference article for service-level SAS; it includes an example of IP ACLing.

- [Constructing an Account SAS](#)

This is the reference article for account-level SAS; it includes an example of IP ACLing.

- Authentication
 - [Authentication for the Azure Storage Services](#)
- Shared Access Signatures Getting Started Tutorial
 - [SAS Getting Started Tutorial](#)

Encryption in Transit

Transport-Level Encryption – Using HTTPS

Another step you should take to ensure the security of your Azure Storage data is to encrypt the data between the client and Azure Storage. The first recommendation is to always use the [HTTPS](#) protocol, which ensures secure communication over the public Internet.

To have a secure communication channel, you should always use HTTPS when calling the REST APIs or accessing objects in storage. Also, **Shared Access Signatures**, which can be used to delegate access to Azure Storage objects, include an option to specify that only the HTTPS protocol can be used when using Shared Access Signatures, ensuring that anybody sending out links with SAS tokens will use the proper protocol.

You can enforce the use of HTTPS when calling the REST APIs to access objects in storage accounts by enabling [Secure transfer required](#) for the storage account. Connections using HTTP will be refused once this is enabled.

Using encryption during transit with Azure File shares

Azure File storage supports HTTPS when using the REST API, but is more commonly used as an SMB file share attached to a VM. SMB 2.1 does not support encryption, so connections are only allowed within the same region in Azure. However, SMB 3.0 supports encryption, and it's available in Windows Server 2012 R2, Windows 8, Windows 8.1, and Windows 10, allowing cross-region access and even access on the desktop.

Note that while Azure File shares can be used with Unix, the Linux SMB client does not yet support encryption, so access is only allowed within an Azure region. Encryption support for Linux is on the roadmap of Linux developers responsible for SMB functionality. When they add encryption, you will have the same ability for accessing an Azure

File share on Linux as you do for Windows.

You can enforce the use of encryption with the Azure Files service by enabling [Secure transfer required](#) for the storage account. If using the REST APIs, HTTPS is required. For SMB, only SMB connections that support encryption will connect successfully.

Resources

- [How to use Azure File storage with Linux](#)

This article shows how to mount an Azure File share on a Linux system and upload/download files.

- [Get started with Azure File storage on Windows](#)

This article gives an overview of Azure File shares and how to mount and use them using PowerShell and .NET.

- [Inside Azure File storage](#)

This article announces the general availability of Azure File storage and provides technical details about the SMB 3.0 encryption.

Using Client-side encryption to secure data that you send to storage

Another option that helps you ensure that your data is secure while being transferred between a client application and Storage is Client-side Encryption. The data is encrypted before being transferred into Azure Storage. When retrieving the data from Azure Storage, the data is decrypted after it is received on the client side. Even though the data is encrypted going across the wire, we recommend that you also use HTTPS, as it has data integrity checks built in which help mitigate network errors affecting the integrity of the data.

Client-side encryption is also a method for encrypting your data at rest, as the data is stored in its encrypted form. We'll talk about this in more detail in the section on [Encryption at Rest](#).

Encryption at Rest

There are three Azure features that provide encryption at rest. Azure Disk Encryption is used to encrypt the OS and data disks in IaaS Virtual Machines. The other two – Client-side Encryption and SSE – are both used to encrypt data in Azure Storage. Let's look at each of these, and then do a comparison and see when each one can be used.

While you can use Client-side Encryption to encrypt the data in transit (which is also stored in its encrypted form in Storage), you may prefer to simply use HTTPS during the transfer, and have some way for the data to be automatically encrypted when it is stored. There are two ways to do this -- Azure Disk Encryption and SSE. One is used to directly encrypt the data on OS and data disks used by VMs, and the other is used to encrypt data written to Azure Blob Storage.

Storage Service Encryption (SSE)

SSE allows you to request that the storage service automatically encrypt the data when writing it to Azure Storage. When you read the data from Azure Storage, it will be decrypted by the storage service before being returned. This enables you to secure your data without having to modify code or add code to any applications.

This is a setting that applies to the whole storage account. You can enable and disable this feature by changing the value of the setting. To do this, you can use the Azure portal, PowerShell, the Azure CLI, the Storage Resource Provider REST API, or the .NET Storage Client Library. By default, SSE is turned off.

At this time, the keys used for the encryption are managed by Microsoft. We generate the keys originally, and manage the secure storage of the keys as well as the regular rotation as defined by internal Microsoft policy. In the future, you will get the ability to manage your own encryption keys, and provide a migration path from Microsoft-managed keys to customer-managed keys.

This feature is available for Standard and Premium Storage accounts created using the Resource Manager

deployment model. SSE applies only to block blobs, page blobs, and append blobs. The other types of data, including tables, queues, and files, will not be encrypted.

Data is only encrypted when SSE is enabled and the data is written to Blob Storage. Enabling or disabling SSE does not impact existing data. In other words, when you enable this encryption, it will not go back and encrypt data that already exists; nor will it decrypt the data that already exists when you disable SSE.

If you want to use this feature with a Classic storage account, you can create a new Resource Manager storage account and use AzCopy to copy the data to the new account.

Client-side Encryption

We mentioned client-side encryption when discussing the encryption of the data in transit. This feature allows you to programmatically encrypt your data in a client application before sending it across the wire to be written to Azure Storage, and to programmatically decrypt your data after retrieving it from Azure Storage.

This does provide encryption in transit, but it also provides the feature of Encryption at Rest. Note that although the data is encrypted in transit, we still recommend using HTTPS to take advantage of the built-in data integrity checks which help mitigate network errors affecting the integrity of the data.

An example of where you might use this is if you have a web application that stores blobs and retrieves blobs, and you want the application and data to be as secure as possible. In that case, you would use client-side encryption. The traffic between the client and the Azure Blob Service contains the encrypted resource, and nobody can interpret the data in transit and reconstitute it into your private blobs.

Client-side encryption is built into the Java and the .NET storage client libraries, which in turn use the Azure Key Vault APIs, making it pretty easy for you to implement. The process of encrypting and decrypting the data uses the envelope technique, and stores metadata used by the encryption in each storage object. For example, for blobs, it stores it in the blob metadata, while for queues, it adds it to each queue message.

For the encryption itself, you can generate and manage your own encryption keys. You can also use keys generated by the Azure Storage Client Library, or you can have the Azure Key Vault generate the keys. You can store your encryption keys in your on-premises key storage, or you can store them in an Azure Key Vault. Azure Key Vault allows you to grant access to the secrets in Azure Key Vault to specific users using Azure Active Directory. This means that not just anybody can read the Azure Key Vault and retrieve the keys you're using for client-side encryption.

Resources

- [Encrypt and decrypt blobs in Microsoft Azure Storage using Azure Key Vault](#)

This article shows how to use client-side encryption with Azure Key Vault, including how to create the KEK and store it in the vault using PowerShell.

- [Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage](#)

This article gives an explanation of client-side encryption, and provides examples of using the storage client library to encrypt and decrypt resources from the four storage services. It also talks about Azure Key Vault.

Using Azure Disk Encryption to encrypt disks used by your virtual machines

Azure Disk Encryption is a new feature. This feature allows you to encrypt the OS disks and Data disks used by an IaaS Virtual Machine. For Windows, the drives are encrypted using industry-standard BitLocker encryption technology. For Linux, the disks are encrypted using the DM-Crypt technology. This is integrated with Azure Key Vault to allow you to control and manage the disk encryption keys.

The solution supports the following scenarios for IaaS VMs when they are enabled in Microsoft Azure:

- Integration with Azure Key Vault
- Standard tier VMs: [A](#), [D](#), [DS](#), [G](#), [GS](#), and so forth series IaaS VMs
- Enabling encryption on Windows and Linux IaaS VMs

- Disabling encryption on OS and data drives for Windows IaaS VMs
- Disabling encryption on data drives for Linux IaaS VMs
- Enabling encryption on IaaS VMs that are running Windows client OS
- Enabling encryption on volumes with mount paths
- Enabling encryption on Linux VMs that are configured with disk striping (RAID) by using mdadm
- Enabling encryption on Linux VMs by using LVM for data disks
- Enabling encryption on Windows VMs that are configured by using storage spaces
- All Azure public regions are supported

The solution does not support the following scenarios, features, and technology in the release:

- Basic tier IaaS VMs
- Disabling encryption on an OS drive for Linux IaaS VMs
- IaaS VMs that are created by using the classic VM creation method
- Integration with your on-premises Key Management Service
- Azure File storage (shared file system), Network File System (NFS), dynamic volumes, and Windows VMs that are configured with software-based RAID systems

NOTE

Linux OS disk encryption is currently supported on the following Linux distributions: RHEL 7.2, CentOS 7.2n, and Ubuntu 16.04.

This feature ensures that all data on your virtual machine disks is encrypted at rest in Azure Storage.

Resources

- [Azure Disk Encryption for Windows and Linux IaaS VMs](#)

Comparison of Azure Disk Encryption, SSE, and Client-Side Encryption

IaaS VMs and their VHD files

For disks used by IaaS VMs, we recommend using Azure Disk Encryption. You can turn on SSE to encrypt the VHD files that are used to back those disks in Azure Storage, but it only encrypts newly written data. This means if you create a VM and then enable SSE on the storage account that holds the VHD file, only the changes will be encrypted, not the original VHD file.

If you create a VM using an image from the Azure Marketplace, Azure performs a [shallow copy](#) of the image to your storage account in Azure Storage, and it is not encrypted even if you have SSE enabled. After it creates the VM and starts updating the image, SSE will start encrypting the data. For this reason, it's best to use Azure Disk Encryption on VMs created from images in the Azure Marketplace if you want them fully encrypted.

If you bring a pre-encrypted VM into Azure from on-premises, you will be able to upload the encryption keys to Azure Key Vault, and continue using the encryption for that VM that you were using on-premises. Azure Disk Encryption is enabled to handle this scenario.

If you have non-encrypted VHD from on-premises, you can upload it into the gallery as a custom image and provision a VM from it. If you do this using the Resource Manager templates, you can ask it to turn on Azure Disk Encryption when it boots up the VM.

When you add a data disk and mount it on the VM, you can turn on Azure Disk Encryption on that data disk. It will encrypt that data disk locally first, and then the service management layer will do a lazy write against storage so the storage content is encrypted.

Client-side encryption

Client-side encryption is the most secure method of encrypting your data, because it encrypts it before transit, and

encrypts the data at rest. However, it does require that you add code to your applications using storage, which you may not want to do. In those cases, you can use HTTPs for your data in transit, and SSE to encrypt the data at rest.

With client-side encryption, you can encrypt table entities, queue messages, and blobs. With SSE, you can only encrypt blobs. If you need table and queue data to be encrypted, you should use client-side encryption.

Client-side encryption is managed entirely by the application. This is the most secure approach, but does require you to make programmatic changes to your application and put key management processes in place. You would use this when you want the extra security during transit, and you want your stored data to be encrypted.

Client-side encryption is more load on the client, and you have to account for this in your scalability plans, especially if you are encrypting and transferring a lot of data.

Storage Service Encryption (SSE)

SSE is managed by Azure Storage. Using SSE does not provide for the security of the data in transit, but it does encrypt the data as it is written to Azure Storage. There is no impact on the performance when using this feature.

You can only encrypt block blobs, append blobs, and page blobs using SSE. If you need to encrypt table data or queue data, you should consider using client-side encryption.

If you have an archive or library of VHD files that you use as a basis for creating new virtual machines, you can create a new storage account, enable SSE, and then upload the VHD files to that account. Those VHD files will be encrypted by Azure Storage.

If you have Azure Disk Encryption enabled for the disks in a VM and SSE enabled on the storage account holding the VHD files, it will work fine; it will result in any newly-written data being encrypted twice.

Storage Analytics

Using Storage Analytics to monitor authorization type

For each storage account, you can enable Azure Storage Analytics to perform logging and store metrics data. This is a great tool to use when you want to check the performance metrics of a storage account, or need to troubleshoot a storage account because you are having performance problems.

Another piece of data you can see in the storage analytics logs is the authentication method used by someone when they access storage. For example, with Blob Storage, you can see if they used a Shared Access Signature or the storage account keys, or if the blob accessed was public.

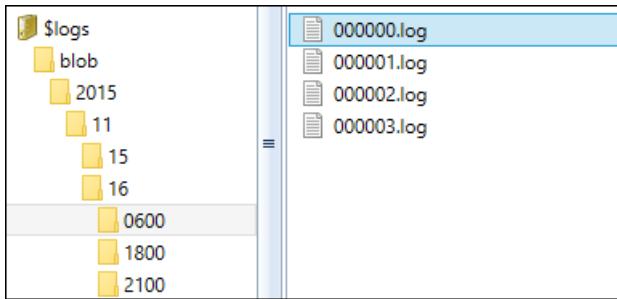
This can be really helpful if you are tightly guarding access to storage. For example, in Blob Storage you can set all of the containers to private and implement the use of an SAS service throughout your applications. Then you can check the logs regularly to see if your blobs are accessed using the storage account keys, which may indicate a breach of security, or if the blobs are public but they shouldn't be.

What do the logs look like?

After you enable the storage account metrics and logging through the Azure portal, analytics data will start to accumulate quickly. The logging and metrics for each service is separate; the logging is only written when there is activity in that storage account, while the metrics will be logged every minute, every hour, or every day, depending on how you configure it.

The logs are stored in block blobs in a container named \$logs in the storage account. This container is automatically created when Storage Analytics is enabled. Once this container is created, you can't delete it, although you can delete its contents.

Under the \$logs container, there is a folder for each service, and then there are subfolders for the year/month/day/hour. Under hour, the logs are simply numbered. This is what the directory structure will look like:



Every request to Azure Storage is logged. Here's a snapshot of a log file, showing the first few fields.

```
1.0;2015-11-16T06:13:26.9046078Z;GetBlobServiceProperties;Success;200;3;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:13:27.2588724Z;GetBlobServiceProperties;Success;200;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:14:28.0166751Z;GetBlobServiceProperties;Success;200;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:14:29.2558837Z;GetBlobServiceProperties;Success;200;3;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:14:43.4307865Z;BlobPreflightRequest;AnonymousSuccess;200;2;anonymous;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:14:43.4528051Z;GetBlobServiceProperties;Success;200;1;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:15:30.3567270Z;GetBlobServiceProperties;Success;200;2;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:15:29.2735098Z;GetBlobServiceProperties;Success;200;5;5;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:32.9445742Z;GetBlobServiceProperties;Success;200;4;3;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:44.2766486Z;ListContainers;Success;200;4;4;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:56.0216743Z;CreateContainer;Success;201;10;10;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:56.0517020Z;ListContainers;Success;200;2;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:59.9423538Z;ListContainers;Success;200;3;3;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:59.9984102Z;ListBlobs;Success;200;3;3;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:17:23.7717291Z;GetBlobProperties;ClientOtherError;404;3;3;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:17:23.8347867Z;PutBlob;Success;201;71;8;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:17:23.9549008Z;GetBlobProperties;Success;200;2;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:17:31.9243814Z;GetBlobProperties;Success;200;2;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:17:31.9554107Z;GetBlob;Success;206;81;5;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:17:46.1437305Z;GetContainerACL;Success;200;2;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";1.0;2015-11-16T06:16:30.3890982Z;GetBlobServiceProperties;Success;200;2;2;authenticated;mystorage;mystorage/blob;"https://mystorage.blob.core.windows.net/";
```

You can see that you can use the logs to track any kind of calls to a storage account.

What are all of those fields for?

There is an article listed in the resources below that provides the list of the many fields in the logs and what they are used for. Here is the list of fields in order:

```
<version-number>;<request-start-time>;<b>operation-type</b>;<request-status>;<http-status-code>;<end-to-end-latency-in-ms>;<server-latency-in-ms>;<authentication-type>;<requester-account-name>;<owner-account-name>;<service-type>;<request-url>;<requested-object-key>;<request-id-header>;<operation-count>;<requester-ip-address>;<request-version-header>;<request-header-size>;<request-packet-size>;<response-header-size>;<response-packet-size>;<request-content-length>;<request-md5>;<server-md5>;<etag-identifier>;<last-modified-time>;<conditions-used>;<user-agent-header>;<referrer-header>;<client-request-id>
```

We're interested in the entries for GetBlob, and how they are authenticated, so we need to look for entries with operation-type "Get-Blob", and check the request-status (4th column) and the authorization-type (8th column).

For example, in the first few rows in the listing above, the request-status is "Success" and the authorization-type is "authenticated". This means the request was validated using the storage account key.

How are my blobs being authenticated?

We have three cases that we are interested in.

1. The blob is public and it is accessed using a URL without a Shared Access Signature. In this case, the request-status is "AnonymousSuccess" and the authorization-type is "anonymous".

1:0:2015-11-17T02:01:29.0488963Z:GetBlob:**AnonymousSuccess**:200:124:37:**anonymous**::mystorage...

2. The blob is private and was used with a Shared Access Signature. In this case, the request-status is "SASSuccess" and the authorization-type is "sas".

1.0;2015-11-16T18:30:05.655Z;GetBlob;SASSuccess;200;416;64;sas::mystorage...

3. The blob is private and the storage key was used to access it. In this case, the request-status is "**Success**" and the authorization-type is "**authenticated**".

You can use the Microsoft Message Analyzer to view and analyze these logs. It includes search and filter capabilities. For example, you might want to search for instances of GetBlob to see if the usage is what you expect, i.e. to make sure someone is not accessing your storage account inappropriately.

Resources

- [Storage Analytics](#)

This article is an overview of storage analytics and how to enable them.

- [Storage Analytics Log Format](#)

This article illustrates the Storage Analytics Log Format, and details the fields available therein, including authentication-type, which indicates the type of authentication used for the request.

- [Monitor a Storage Account in the Azure portal](#)

This article shows how to configure monitoring of metrics and logging for a storage account.

- [End-to-End Troubleshooting using Azure Storage Metrics and Logging, AzCopy, and Message Analyzer](#)

This article talks about troubleshooting using the Storage Analytics and shows how to use the Microsoft Message Analyzer.

- [Microsoft Message Analyzer Operating Guide](#)

This article is the reference for the Microsoft Message Analyzer and includes links to a tutorial, quick start, and feature summary.

Cross-Origin Resource Sharing (CORS)

Cross-domain access of resources

When a web browser running in one domain makes an HTTP request for a resource from a different domain, this is called a cross-origin HTTP request. For example, an HTML page served from contoso.com makes a request for a jpeg hosted on fabrikam.blob.core.windows.net. For security reasons, browsers restrict cross-origin HTTP requests initiated from within scripts, such as JavaScript. This means that when some JavaScript code on a web page on contoso.com requests that jpeg on fabrikam.blob.core.windows.net, the browser will not allow the request.

What does this have to do with Azure Storage? Well, if you are storing static assets such as JSON or XML data files in Blob Storage using a storage account called Fabrikam, the domain for the assets will be fabrikam.blob.core.windows.net, and the contoso.com web application will not be able to access them using JavaScript because the domains are different. This is also true if you're trying to call one of the Azure Storage Services – such as Table Storage – that return JSON data to be processed by the JavaScript client.

Possible solutions

One way to resolve this is to assign a custom domain like "storage.contoso.com" to fabrikam.blob.core.windows.net. The problem is that you can only assign that custom domain to one storage account. What if the assets are stored in multiple storage accounts?

Another way to resolve this is to have the web application act as a proxy for the storage calls. This means if you are uploading a file to Blob Storage, the web application would either write it locally and then copy it to Blob Storage, or it would read all of it into memory and then write it to Blob Storage. Alternately, you could write a dedicated web application (such as a Web API) that uploads the files locally and writes them to Blob Storage. Either way, you have to account for that function when determining the scalability needs.

How can CORS help?

Azure Storage allows you to enable CORS – Cross Origin Resource Sharing. For each storage account, you can specify domains that can access the resources in that storage account. For example, in our case outlined above, we

can enable CORS on the fabrikam.blob.core.windows.net storage account and configure it to allow access to contoso.com. Then the web application contoso.com can directly access the resources in fabrikam.blob.core.windows.net.

One thing to note is that CORS allows access, but it does not provide authentication, which is required for all non-public access of storage resources. This means you can only access blobs if they are public or you include a Shared Access Signature giving you the appropriate permission. Tables, queues, and files have no public access, and require a SAS.

By default, CORS is disabled on all services. You can enable CORS by using the REST API or the storage client library to call one of the methods to set the service policies. When you do that, you include a CORS rule, which is in XML. Here's an example of a CORS rule that has been set using the Set Service Properties operation for the Blob Service for a storage account. You can perform that operation using the storage client library or the REST APIs for Azure Storage.

```
<Cors>
  <CorsRule>
    <AllowedOrigins>http://www.contoso.com, http://www.fabrikam.com</AllowedOrigins>
    <AllowedMethods>PUT,GET</AllowedMethods>
    <AllowedHeaders>x-ms-meta-data*,x-ms-meta-target*,x-ms-meta-abc</AllowedHeaders>
    <ExposedHeaders>x-ms-meta-*</ExposedHeaders>
    <MaxAgeInSeconds>200</MaxAgeInSeconds>
  </CorsRule>
<Cors>
```

Here's what each row means:

- **AllowedOrigins** This tells which non-matching domains can request and receive data from the storage service. This says that both contoso.com and fabrikam.com can request data from Blob Storage for a specific storage account. You can also set this to a wildcard (*) to allow all domains to access requests.
- **AllowedMethods** This is the list of methods (HTTP request verbs) that can be used when making the request. In this example, only PUT and GET are allowed. You can set this to a wildcard (*) to allow all methods to be used.
- **AllowedHeaders** This is the request headers that the origin domain can specify when making the request. In this example, all metadata headers starting with x-ms-meta-data, x-ms-meta-target, and x-ms-meta-abc are permitted. The wildcard character (*) indicates that any header beginning with the specified prefix is allowed.
- **ExposedHeaders** This tells which response headers should be exposed by the browser to the request issuer. In this example, any header starting with "x-ms-meta-" will be exposed.
- **MaxAgeInSeconds** This is the maximum amount of time that a browser will cache the preflight OPTIONS request. (For more information about the preflight request, check the first article below.)

Resources

For more information about CORS and how to enable it, please check out these resources.

- [Cross-Origin Resource Sharing \(CORS\) Support for the Azure Storage Services on Azure.com](#)

This article provides an overview of CORS and how to set the rules for the different storage services.

- [Cross-Origin Resource Sharing \(CORS\) Support for the Azure Storage Services on MSDN](#)

This is the reference documentation for CORS support for the Azure Storage Services. This has links to articles applying to each storage service, and shows an example and explains each element in the CORS file.

- [Microsoft Azure Storage: Introducing CORS](#)

This is a link to the initial blog article announcing CORS and showing how to use it.

Frequently asked questions about Azure Storage security

1. How can I verify the integrity of the blobs I'm transferring into or out of Azure Storage if I can't use the HTTPS protocol?

If for any reason you need to use HTTP instead of HTTPS and you are working with block blobs, you can use MD5 checking to help verify the integrity of the blobs being transferred. This will help with protection from network/transport layer errors, but not necessarily with intermediary attacks.

If you can use HTTPS, which provides transport level security, then using MD5 checking is redundant and unnecessary.

For more information, please check out the [Azure Blob MD5 Overview](#).

2. What about FIPS-Compliance for the U.S. Government?

The United States Federal Information Processing Standard (FIPS) defines cryptographic algorithms approved for use by U.S. Federal government computer systems for the protection of sensitive data. Enabling FIPS mode on a Windows server or desktop tells the OS that only FIPS-validated cryptographic algorithms should be used. If an application uses non-compliant algorithms, the applications will break. With .NET Framework versions 4.5.2 or higher, the application automatically switches the cryptography algorithms to use FIPS-compliant algorithms when the computer is in FIPS mode.

Microsoft leaves it up to each customer to decide whether to enable FIPS mode. We believe there is no compelling reason for customers who are not subject to government regulations to enable FIPS mode by default.

Resources

- [Why We're Not Recommending "FIPS Mode" Anymore](#)

This blog article gives an overview of FIPS and explains why they don't enable FIPS mode by default.

- [FIPS 140 Validation](#)

This article provides information on how Microsoft products and cryptographic modules comply with the FIPS standard for the U.S. Federal government.

- ["System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing" security settings effects in Windows XP and in later versions of Windows](#)

This article talks about the use of FIPS mode in older Windows computers.

Best practices for Azure VM security

8/10/2017 • 7 min to read • [Edit Online](#)

In most infrastructure as a service (IaaS) scenarios, [Azure virtual machines \(VMs\)](#) are the main workload for organizations that use cloud computing. This fact is especially evident in [hybrid scenarios](#) where organizations want to slowly migrate workloads to the cloud. In such scenarios, follow the [general security considerations for IaaS](#), and apply security best practices to all your VMs.

This article discusses various VM security best practices, each derived from our customers' and our own direct experiences with VMs.

The best practices are based on a consensus of opinion, and they work with current Azure platform capabilities and feature sets. Because opinions and technologies can change over time, we plan to update this article regularly to reflect those changes.

For each best practice, the article explains:

- What the best practice is.
- Why it's a good idea to enable it.
- How you can learn to enable it.
- What might happen if you fail to enable it.
- Possible alternatives to the best practice.

The article examines the following VM security best practices:

- VM authentication and access control
- VM availability and network access
- Protect data at rest in VMs by enforcing encryption
- Manage your VM updates
- Manage your VM security posture
- Monitor VM performance

VM authentication and access control

The first step in protecting your VM is to ensure that only authorized users are able to set up new VMs. You can use [Azure Resource Manager policies](#) to establish conventions for resources in your organization, create customized policies, and apply these policies to resources, such as [resource groups](#).

VMs that belong to a resource group naturally inherit its policies. Although we recommend this approach to managing VMs, you can also control access to individual VM policies by using [role-based access control \(RBAC\)](#).

When you enable Resource Manager policies and RBAC to control VM access, you help improve overall VM security. We recommend that you consolidate VMs with the same life cycle into the same resource group. By using resource groups, you can deploy, monitor, and roll up billing costs for your resources. To enable users to access and set up VMs, use a [least privilege approach](#). And when you assign privileges to users, plan to use the following built-in Azure roles:

- [Virtual Machine Contributor](#): Can manage VMs, but not the virtual network or storage account to which they are connected.
- [Classic Virtual Machine Contributor](#): Can manage VMs created by using the classic deployment model, but not the virtual network or storage account to which the VMs are connected.
- [Security Manager](#): Can manage security components, security policies, and VMs.
- [DevTest Labs User](#): Can view everything and connect, start, restart, and shut down VMs.

Don't share accounts and passwords between administrators, and don't reuse passwords across multiple user accounts or services, particularly passwords for social media or other non-administrative activities. Ideally, you should use [Azure Resource Manager](#) templates to set up your VMs securely. By using this approach, you can strengthen your deployment choices and enforce security settings throughout the deployment.

Organizations that do not enforce data-access control by taking advantage of capabilities such as RBAC might be granting their users more privileges than necessary. Inappropriate user access to certain data can directly compromise that data.

VM availability and network access

If your VM runs critical applications that need to have high availability, we strongly recommend that you use multiple VMs. For better availability, create at least two VMs in the [availability set](#).

[Azure Load Balancer](#) also requires that load-balanced VMs belong to the same availability set. If these VMs must be accessed from the Internet, you must configure an [Internet-facing load balancer](#).

When VMs are exposed to the Internet, it is important that you [control network traffic flow with network security groups \(NSGs\)](#). Because NSGs can be applied to subnets, you can minimize the number of NSGs by grouping your resources by subnet and then applying NSGs to the subnets. The intent is to create a layer of network isolation, which you can do by properly configuring the [network security](#) capabilities in Azure.

You can also use the just-in-time (JIT) VM-access feature from Azure Security Center to control who has remote access to a specific VM, and for how long.

Organizations that don't enforce network-access restrictions to Internet-facing VMs are exposed to security risks, such as a Remote Desktop Protocol (RDP) Brute Force attack.

Protect data at rest in your VMs by enforcing encryption

[Data encryption at rest](#) is a mandatory step toward data privacy, compliance, and data sovereignty. [Azure Disk Encryption](#) enables IT administrators to encrypt Windows and Linux IaaS VM disks. Disk Encryption combines the industry-standard Windows BitLocker feature and the Linux dm-crypt feature to provide volume encryption for the OS and the data disks.

You can apply Disk Encryption to help safeguard your data to meet your organizational security and compliance requirements. Your organization should consider using encryption to help mitigate risks related to unauthorized data access. We also recommend that you encrypt your drives before you write sensitive data to them.

Be sure to encrypt your VM data volumes to protect them at rest in your Azure storage account. Safeguard the encryption keys and secret by using [Azure Key Vault](#).

Organizations that do not enforce data encryption are more exposed to data-integrity issues. For example, unauthorized or rogue users might steal data in compromised accounts or gain unauthorized access to data coded in ClearFormat. Besides taking on such risks, to comply with industry regulations, companies must prove that they are exercising diligence and using correct security controls to enhance their data security.

To learn more about Disk Encryption, see [Azure Disk Encryption for Windows and Linux IaaS VMs](#).

Manage your VM updates

Because Azure VMs, like all on-premises VMs, are intended to be user-managed, Azure doesn't push Windows updates to them. You are, however, encouraged to leave the automatic Windows Update setting enabled. Another option is to deploy [Windows Server Update Services \(WSUS\)](#) or another suitable update-management product either on another VM or on-premises. Both WSUS and Windows Update keep VMs current. We also recommend that you use a scanning product to verify that all your IaaS VMs are up to date.

Stock images provided by Azure are routinely updated to include the most recent round of Windows updates. However, there is no guarantee that the images will be current at deployment time. A slight lag (of no more than a few weeks) following public releases might be possible. Checking for and installing all Windows updates should be the first step of every deployment. This measure is especially important to apply when you deploy images that come from either you or your own library. Images that are provided as part of the Azure Marketplace are updated automatically by default.

Organizations that don't enforce software-update policies are more exposed to threats that exploit known, previously fixed vulnerabilities. Besides risking such threats, to comply with industry regulations, companies must prove that they are exercising diligence and using correct security controls to help ensure the security of their workload located in the cloud.

It is important to emphasize that software-update best practices for traditional datacenters and Azure IaaS have many similarities. We therefore recommend that you evaluate your current software update policies to include VMs.

Manage your VM security posture

Cyber threats are evolving, and safeguarding your VMs requires a rich monitoring capability that can quickly detect threats,

prevent unauthorized access to your resources, trigger alerts, and reduce false positives. The security posture for such a workload comprises all security aspects of the VM, from update management to secure network access.

To monitor the security posture of your [Windows](#) and [Linux VMs](#), use [Azure Security Center](#). In Azure Security Center, safeguard your VMs by taking advantage of the following capabilities:

- Apply OS security settings with recommended configuration rules
- Identify and download system security and critical updates that might be missing
- Deploy Endpoint antimalware protection recommendations
- Validate disk encryption
- Assess and remediate vulnerabilities
- Detect threats

Security Center can actively monitor for threats, and potential threats are exposed under **Security Alerts**. Correlated threats are aggregated in a single view called **Security Incident**.

To understand how Security Center can help you identify potential threats in your VMs located in Azure, watch the following video:



Organizations that don't enforce a strong security posture for their VMs remain unaware of potential attempts by unauthorized users to circumvent established security controls.

Monitor VM performance

Resource abuse can be a problem when VM processes consume more resources than they should. Performance issues with a VM can lead to service disruption, which violates the security principle of availability. For this reason, it is imperative to monitor VM access not only reactively, while an issue is occurring, but also proactively, against baseline performance as measured during normal operation.

By analyzing [Azure diagnostic log files](#), you can monitor your VM resources and identify potential issues that might compromise performance and availability. The Azure Diagnostics Extension provides monitoring and diagnostics capabilities on Windows-based VMs. You can enable these capabilities by including the extension as part of the [Azure Resource Manager template](#).

You can also use [Azure Monitor](#) to gain visibility into your resource's health.

Organizations that don't monitor VM performance are unable to determine whether certain changes in performance patterns are normal or abnormal. If the VM is consuming more resources than normal, such an anomaly could indicate a potential attack from an external resource or a compromised process running in the VM.

Security best practices for IaaS workloads in Azure

8/30/2017 • 14 min to read • [Edit Online](#)

As you started thinking about moving workloads to Azure infrastructure as a service (IaaS), you probably realized that some considerations are familiar. You might already have experience securing virtual environments. When you move to Azure IaaS, you can apply your expertise in securing virtual environments and use a new set of options to help secure your assets.

Let's start by saying that we should not expect to bring on-premises resources as one-to-one to Azure. The new challenges and the new options bring an opportunity to reevaluate existing designs, tools, and processes.

Your responsibility for security is based on the type of cloud service. The following chart summarizes the balance of responsibility for both Microsoft and you:

Responsibility	On-Prem	IaaS	PaaS	SaaS
Data classification & accountability	■	■	■	■
Client & end-point protection	■	■	■	■
Identity & access management	■	■	■	■
Application level controls	■	■	■	■
Network controls	■	■	■	■
Host infrastructure	■	■	■	■
Physical security	■	■	■	■

■ Cloud Customer ■ Cloud Provider

We'll discuss some of the options available in Azure that can help you meet your organization's security requirements. Keep in mind that security requirements can vary for different types of workloads. Not one of these best practices can by itself secure your systems. Like anything else in security, you have to choose the appropriate options and see how the solutions can complement each other by filling gaps.

Use Privileged Access Workstations

Organizations often fall prey to cyberattacks because administrators perform actions while using accounts with elevated rights. Usually this isn't done maliciously but because existing configuration and processes allow it. Most of these users understand the risk of these actions from a conceptual standpoint but still choose to do them.

Doing things like checking email and browsing the Internet seem innocent enough. But they might expose elevated accounts to compromise by malicious actors. Browsing activities, specially crafted emails, or other techniques can be used to gain access to your enterprise. We highly recommend the use of secure management workstations (SAWs) for conducting all Azure administration tasks. SAWs are a way of reducing exposure to accidental compromise.

Privileged Access Workstations (PAWs) provide a dedicated operating system for sensitive tasks--one that is protected from Internet attacks and threat vectors. Separating these sensitive tasks and accounts from the daily-use workstations and devices provides strong protection. This separation limits the impact of phishing attacks, application and OS vulnerabilities, various impersonation attacks, and credential theft attacks. (keystroke logging, Pass-the-Hash, and Pass-the-Ticket)

The PAW approach is an extension of the well-established and recommended practice to use an individually assigned administrative account. The administrative account is separate from a standard user account. A PAW provides a trustworthy workstation for those sensitive accounts.

For more information and implementation guidance, see [Privileged Access Workstations](#).

Use Multi-Factor Authentication

In the past, your network perimeter was used to control access to corporate data. In a cloud-first, mobile-first world, identity is the control plane: You use it to control access to IaaS services from any device. You also use it to get visibility and insight into where and how your data is being used. Protecting the digital identity of your Azure users is the cornerstone of protecting your subscriptions from identity theft and other cybercrimes.

One of the most beneficial steps that you can take to secure an account is to enable two-factor authentication. Two-factor authentication is a way of authenticating by using something in addition to a password. It helps mitigate the risk of access by someone who manages to get someone else's password.

[Azure Multi-Factor Authentication](#) helps safeguard access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication with a range of easy verification options--phone call, text message, or mobile app notification. Users choose the method that they prefer.

The easiest way to use Multi-Factor Authentication is the Microsoft Authenticator mobile app that can be used on mobile devices running Windows, iOS, and Android. With the latest release of Windows 10 and the integration of on-premises Active Directory with Azure Active Directory (Azure AD), [Windows Hello for Business](#) can be used for seamless single sign-on to Azure resources. In this case, the Windows 10 device is used as the second factor for authentication.

For accounts that manage your Azure subscription and for accounts that can sign in to virtual machines, using Multi-Factor Authentication gives you a much greater level of security than using only a password. Other forms of two-factor authentication might work just as well, but deploying them might be complicated if they're not already in production.

The following screenshot shows some of the options available for Azure Multi-Factor Authentication:

what's your preferred option?

We'll use this verification option by default.

Call my authentication phone ▾

how would you like to respond?

Set up one or more of these options. [Learn more](#)

<input checked="" type="checkbox"/> Authentication phone	United States (+1) ▾	67
<input type="checkbox"/> Office phone	Select your country or region ▾	
		Extension ▾
<input checked="" type="checkbox"/> Alternate authentication phone	United States (+1) ▾	30
<input checked="" type="checkbox"/> Azure Authenticator app	Configure	Mobile app has been configured.

Limit and constrain administrative access

Securing the accounts that can manage your Azure subscription is extremely important. The compromise of any of those accounts negates the value of all the other steps that you might take to ensure the confidentiality and integrity of your data. As recently illustrated by the [Edward Snowden](#) internal attacks pose a huge threat to the overall security of any organization.

Evaluate individuals for administrative rights by following criteria similar to these:

- Are they performing tasks that require administrative privileges?
- How often are the tasks performed?
- Is there a specific reason why the tasks cannot be performed by another administrator on their behalf?

Document all other known alternative approaches to granting the privilege and why each isn't acceptable.

The use of just-in-time administration prevents the unnecessary existence of accounts with elevated rights during periods when those rights are not needed. Accounts have elevated rights for a limited time so that administrators can do their jobs. Then, those rights are removed at the end of a shift or when a task is completed.

You can use [Privileged Identity Management](#) to manage, monitor, and control access in your organization. It helps you remain aware of the actions that individuals take in your organization. It also brings just-in-time administration to Azure AD by introducing the concept of eligible admins. These are individuals who have accounts with the potential to be granted admin rights. These types of users can go through an activation process and be granted admin rights for a limited time.

Use DevTest Labs

Using Azure for labs and development environments enables organizations to gain agility in testing and development by taking away the delays that hardware procurement introduces. Unfortunately, a lack of familiarity with Azure or a desire to help expedite its adoption might lead the administrator to be overly permissive with rights assignment. This risk might unintentionally expose the organization to internal attacks. Some users might be granted a lot more access than they should have.

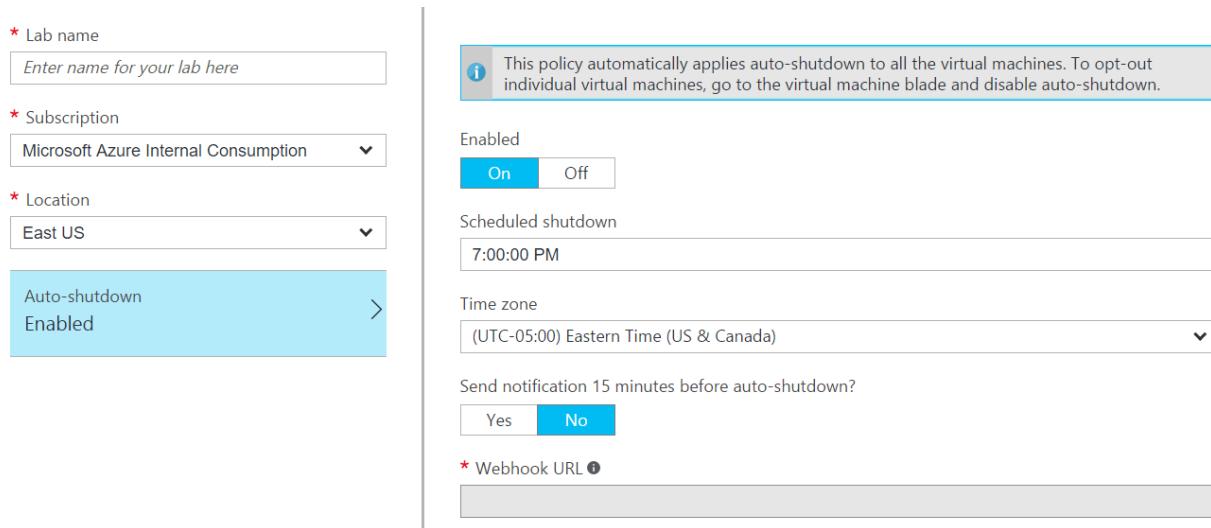
The [Azure DevTest Labs](#) service uses [Azure Role-Based Access Control](#) (RBAC). By using RBAC, you can segregate duties within your team into roles that grant only the level of access necessary for users to do their jobs. RBAC

comes with predefined roles (owner, lab user, and contributor). You can even use these roles to assign rights to external partners and greatly simplify collaboration.

Because DevTest Labs uses RBAC, it's possible to create additional, [custom roles](#). DevTest Labs not only simplifies the management of permissions, it simplifies the process of getting environments provisioned. It also helps you deal with other typical challenges of teams that are working on development and test environments. It requires some preparation, but in the long term, it will make things easier for your team.

Azure DevTest Labs features include:

- Administrative control over the options available to users. The administrator can centrally manage things like allowed VM sizes, maximum number of VMs, and when VMs are started and shut down.
- Automation of lab environment creation.
- Cost tracking.
- Simplified distribution of VMs for temporary collaborative work.
- Self-service that enables users to provision their labs by using templates.
- Managing and limiting consumption.



No additional cost is associated with the usage of DevTest Labs. The creation of labs, policies, templates, and artifacts is free. You pay for only the Azure resources used in your labs, such as virtual machines, storage accounts, and virtual networks.

Control and limit endpoint access

Hosting labs or production systems in Azure means that your systems need to be accessible from the Internet. By default, a new Windows virtual machine has the RDP port accessible from the Internet, and a Linux virtual machine has the SSH port open. Taking steps to limit exposed endpoints is necessary to minimize the risk of unauthorized access.

Technologies in Azure can help you limit the access to those administrative endpoints. In Azure, you can use [network security groups](#) (NSGs). When you use Azure Resource Manager for deployment, NSGs limit the access from all networks to just the management endpoints (RDP or SSH). When you think NSGs, think router ACLs. You can use them to tightly control the network communication between various segments of your Azure networks. This is similar to creating networks in perimeter networks or other isolated networks. They do not inspect the traffic, but they do help with network segmentation.

In Azure, you can configure a [site-to-site VPN](#) from your on-premises network. A site-to-site VPN extends your on-premises network to the cloud. This gives you another opportunity to use NSGs, because you can also modify the NSGs to not allow access from anywhere other than the local network. You can then require that administration is done by first connecting to the Azure network via VPN.

The site-to-site VPN option might be most attractive in cases where you are hosting production systems that are closely integrated with your on-premises resources in Azure.

Alternatively, you can use the [point-to-site](#) option in situations where you want to manage systems that don't need access to on-premises resources. Those systems can be isolated in their own Azure virtual network. Administrators can VPN into the Azure hosted environment from their administrative workstation.

NOTE

You can use either VPN option to reconfigure the ACLs on the NSGs to not allow access to management endpoints from the Internet.

Another option worth considering is a [Remote Desktop Gateway](#) deployment. You can use this deployment to securely connect to Remote Desktop servers over HTTPS, while applying more detailed controls to those connections.

Features that you would have access to include:

- Administrator options to limit connections to requests from specific systems.
- Smart-card authentication or Azure Multi-Factor Authentication.
- Control over which systems someone can connect to via the gateway.
- Control over device and disk redirection.

Use a key management solution

Secure key management is essential to protecting data in the cloud. With [Azure Key Vault](#), you can securely store encryption keys and small secrets like passwords in hardware security modules (HSMs). For added assurance, you can import or generate keys in HSMs.

Microsoft processes your keys in FIPS 140-2 Level 2 validated HSMs (hardware and firmware). Monitor and audit key use with Azure logging: pipe logs into Azure or your Security Information and Event Management (SIEM) system for additional analysis and threat detection.

Anyone with an Azure subscription can create and use key vaults. Although Key Vault benefits developers and security administrators, it can be implemented and managed by an administrator who is responsible for managing Azure services in an organization.

Encrypt virtual disks and disk storage

[Azure Disk Encryption](#) addresses the threat of data theft or exposure from unauthorized access that's achieved by moving a disk. The disk can be attached to another system as a way of bypassing other security controls. Disk encryption uses [BitLocker](#) in Windows and DM-Crypt in Linux to encrypt operating system and data drives. Azure Disk Encryption integrates with Key Vault to control and manage the encryption keys. It's available for standard VMs and VMs with premium storage.

For more information, see [Azure Disk Encryption in Windows and Linux IaaS VMs](#).

[Azure Storage Service Encryption](#) helps protect your data at rest. It's enabled at the storage account level. It encrypts data as it's written in our datacenters, and it's automatically decrypted as you access it. It supports the following scenarios:

- Encryption of block blobs, append blobs, and page blobs
- Encryption of archived VHDs and templates brought to Azure from on-premises
- Encryption of underlying OS and data disks for IaaS VMs that you created by using your VHDs

Before you proceed with Azure Storage Encryption, be aware of two limitations:

- It is not available on classic storage accounts.
- It encrypts only data written after encryption is enabled.

Use a centralized security management system

Your servers need to be monitored for patching, configuration, events, and activities that might be considered security concerns. To address those concerns, you can use [Security Center](#) and [Operations Management Suite Security and Compliance](#). Both of these options go beyond the configuration in the operating system. They also provide monitoring of the configuration of the underlying infrastructure, like network configuration and virtual appliance use.

Manage operating systems

In an IaaS deployment, you are still responsible for the management of the systems that you deploy, just like any other server or workstation in your environment. Patching, hardening, rights assignments, and any other activity related to the maintenance of your system are still your responsibility. For systems that are tightly integrated with your on-premises resources, you might want to use the same tools and procedures that you're using on-premises for things like antivirus, antimalware, patching, and backup.

Harden systems

All virtual machines in Azure IaaS should be hardened so that they expose only service endpoints that are required for the applications that are installed. For Windows virtual machines, follow the recommendations that Microsoft publishes as baselines for the [Security Compliance Manager](#) solution.

Security Compliance Manager is a free tool. You can use it to quickly configure and manage your desktops, traditional datacenter, and private and public cloud by using Group Policy and System Center Configuration Manager.

Security Compliance Manager provides ready-to-deploy policies and Desired Configuration Management configuration packs that are tested. These baselines are based on [Microsoft Security Guidance](#) recommendations and industry best practices. They help you manage configuration drift, address compliance requirements, and reduce security threats.

You can use Security Compliance Manager to import the current configuration of your computers by using two different methods. First, you can import Active Directory-based group policies. Second, you can import the configuration of a "golden master" reference machine by using the [LocalGPO tool](#) to back up the local group policy. You can then import the local group policy into Security Compliance Manager.

Compare your standards to industry best practices, customize them, and create new policies and Desired Configuration Management configuration packs. Baselines have been published for all supported operating systems, including Windows 10 Anniversary Update and Windows Server 2016.

Install and manage antimalware

For environments that are hosted separately from your production environment, you can use an antimalware extension to help protect your virtual machines and cloud services. It integrates with [Azure Security Center](#).

[Microsoft Antimalware](#) includes features like real-time protection, scheduled scanning, malware remediation, signature updates, engine updates, samples reporting, exclusion event collection, and [PowerShell support](#).

The screenshot shows three blades of the Azure portal:

- New resource**: A list of available extensions, including Microsoft Antimalware, Chef Client, CloudLink SecureVM Agent, Custom Script, McAfee Endpoint Protection, Microsoft Antimalware, Octopus Deploy Tentacle Agent, PowerShell Desired State Configuration, and Puppet Enterprise Agent.
- Microsoft Antimalware**: Details about the Microsoft Antimalware extension, including its publisher (Microsoft Corp.) and useful links (Documentation, Powershell Cmdlets).
- Add Extension**: A configuration blade for Microsoft Antimalware, featuring sections for EXCLUDED FILES AND LOCATIONS, EXCLUDED FILE EXTENSIONS, EXCLUDED PROCESSES, REAL-TIME PROTECTION (checked), RUN A SCHEDULED SCAN (unchecked), SCAN TYPE (Quick), SCAN DAY (Saturday), and SCAN TIME (120).

Install the latest security updates

Some of the first workloads that customers move to Azure are labs and external-facing systems. If your Azure-hosted virtual machines host applications or services that need to be accessible to the Internet, be vigilant about patching. Patch beyond the operating system. Unpatched vulnerabilities on third-party applications can also lead to problems that can be avoided if good patch management is in place.

Deploy and test a backup solution

Just like security updates, a backup needs to be handled the same way that you handle any other operation. This is true of systems that are part of your production environment extending to the cloud. Test and dev systems must follow backup strategies that provide restore capabilities that are similar to what users have grown accustomed to, based on their experience with on-premises environments.

Production workloads moved to Azure should integrate with existing backup solutions when possible. Or, you can use [Azure Backup](#) to help address your backup requirements.

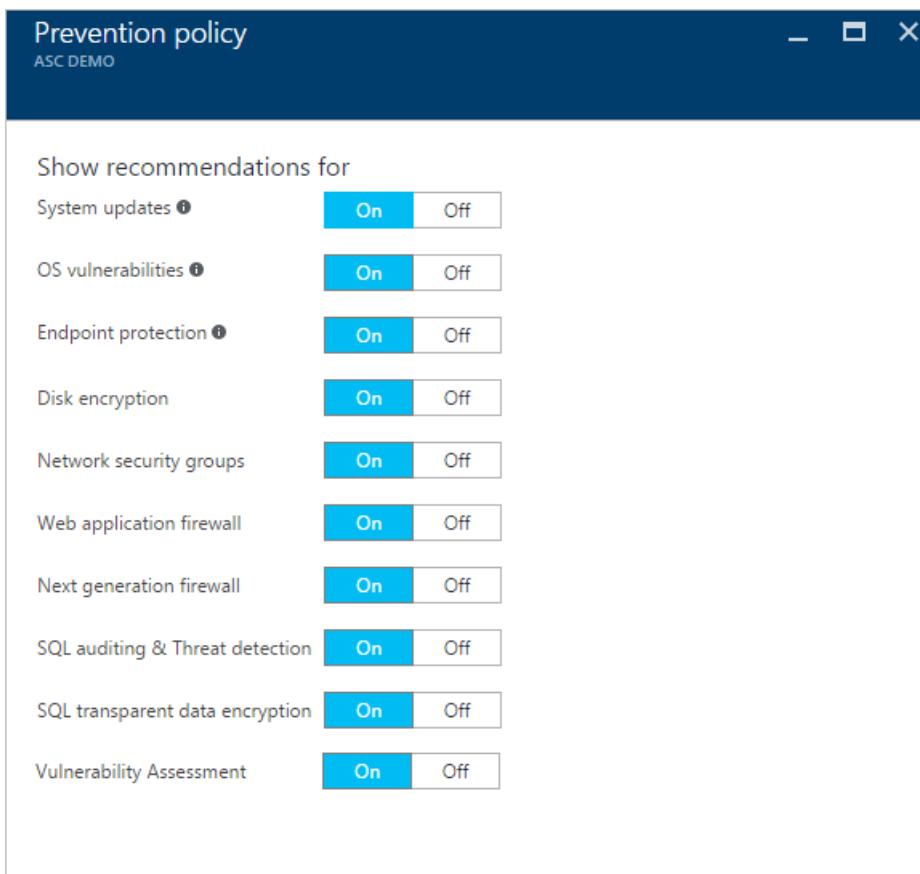
Monitor

[Security Center](#) provides ongoing evaluation of the security state of your Azure resources to identify potential security vulnerabilities. A list of recommendations guides you through the process of configuring needed controls.

Examples include:

- Provisioning antimalware to help identify and remove malicious software.
- Configuring network security groups and rules to control traffic to virtual machines.
- Provisioning web application firewalls to help defend against attacks that target your web applications.
- Deploying missing system updates.
- Addressing OS configurations that do not match the recommended baselines.

The following image shows some of the options that you can enable in Security Center.



[Operations Management Suite](#) is a Microsoft cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure. Because Operations Management Suite is implemented as a cloud-based service, it can be deployed quickly and with minimal investment in infrastructure resources.

New features are delivered automatically, saving you from ongoing maintenance and upgrade costs. Operations Management Suite also integrates with System Center Operations Manager. It has different components to help you better manage your Azure workloads, including a [Security and Compliance](#) module.

You can use the security and compliance features in Operations Management Suite to view information about your resources. The information is organized into four major categories:

- **Security domains:** Further explore security records over time. Access malware assessment, update assessment, network security information, identity and access information, and computers with security events. Take advantage of quick access to the Azure Security Center dashboard.
- **Notable issues:** Quickly identify the number of active issues and the severity of these issues.
- **Detections (preview):** Identify attack patterns by visualizing security alerts as they happen against your resources.
- **Threat intelligence:** Identify attack patterns by visualizing the total number of servers with outbound malicious IP traffic, the malicious threat type, and a map that shows where these IPs are coming from.
- **Common security queries:** See a list of the most common security queries that you can use to monitor your environment. When you click one of those queries, the **Search** blade opens and shows the results for that query.

The following screenshot shows an example of the information that Operations Management Suite can display.

COMPUTERS COMPARED TO BASELINE

Computers assessed	Average passed
7	46%

REQUIRED RULES STATUS

Failed rules by severity

Severity	Count
Critical	61
Warning	32
Info	53
Total	146

Failed rules by type

Type	Count
Registry Key	100
Security Policy	28
Audit Policy	18
Total	146

COMPUTERS MISSING BASELINE ASSESSMENT

Computers not assessed due to OS incompatability or failures
3

COMPUTER	TOTAL RULES	PASSED
sql-0.contoso77.com	136	45%
C77-ATA-Center	138	46%
sql-w.contoso77.com	138	46%
SPS-APP-0.contoso77.com	133	47%
SPS-APP-1.contoso77.com	133	47%
SPS-WEB-0.contoso77.com	133	47%
SPS-WEB-1.contoso77.com	133	47%

[See all...](#)

Next steps

- [Azure Security Team Blog](#)
- [Microsoft Security Response Center](#)
- [Azure security best practices and patterns](#)

Microsoft Antimalware for Azure Cloud Services and Virtual Machines

6/27/2017 • 10 min to read • [Edit Online](#)

The modern threat landscape for cloud environments is extremely dynamic, increasing the pressure on business IT cloud subscribers to maintain effective protection in order to meet compliance and security requirements.

Microsoft Antimalware for Azure is free real-time protection capability that helps identify and remove viruses, spyware, and other malicious software, with configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems.

The solution is built on the same antimalware platform as Microsoft Security Essentials [MSE], Microsoft Forefront Endpoint Protection, Microsoft System Center Endpoint Protection, Windows Intune, and Windows Defender for Windows 8.0 and higher. Microsoft Antimalware for Azure is a single-agent solution for applications and tenant environments, designed to run in the background without human intervention. You can deploy protection based on the needs of your application workloads, with either basic secure-by-default or advanced custom configuration, including antimalware monitoring.

When you deploy and enable Microsoft Antimalware for Azure for your applications, the following core features are available:

- **Real-time protection** - monitors activity in Cloud Services and on Virtual Machines to detect and block malware execution.
- **Scheduled scanning** - periodically performs targeted scanning to detect malware, including actively running programs.
- **Malware remediation** - automatically takes action on detected malware, such as deleting or quarantining malicious files and cleaning up malicious registry entries.
- **Signature updates** - automatically installs the latest protection signatures (virus definitions) to ensure protection is up-to-date on a pre-determined frequency.
- **Antimalware Engine updates** – automatically updates the Microsoft Antimalware engine.
- **Antimalware Platform updates** – automatically updates the Microsoft Antimalware platform.
- **Active protection** - reports telemetry metadata about detected threats and suspicious resources to Microsoft Azure to ensure rapid response to the evolving threat landscape, as well as enabling real-time synchronous signature delivery through the Microsoft Active Protection System (MAPS).
- **Samples reporting** - provides and reports samples to the Microsoft Antimalware service to help refine the service and enable troubleshooting.
- **Exclusions** – allows application and service administrators to configure certain files, processes, and drives to exclude them from protection and scanning for performance and/or other reasons.
- **Antimalware event collection** - records the antimalware service health, suspicious activities, and remediation actions taken in the operating system event log and collects them into the customer's Azure Storage account.

NOTE

Microsoft Antimalware can also be deployed using Azure Security Center. Read [Install Endpoint Protection in Azure Security Center](#) for more information.

Architecture

The Microsoft Antimalware for Azure solution includes the Microsoft Antimalware Client and Service, Antimalware classic deployment model, Antimalware PowerShell cmdlets and Azure Diagnostics Extension. The Microsoft Antimalware solution is supported on Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2 operating system families. It is not supported on the Windows Server 2008 operating system. Support for Windows Server 2016 with Defender has been released, you can read more about this update [here](#).

The Microsoft Antimalware Client and Service is installed by default in a disabled state in all supported Azure guest operating system families in the Cloud Services platform. The Microsoft Antimalware Client and Service is not installed by default in the Virtual Machines platform and is available as an optional feature through the Azure portal and Visual Studio Virtual Machine configuration under Security Extensions.

When using Azure Websites, the underlying service that hosts the web app has Microsoft Antimalware enabled on it. This is used to protect Azure Websites infrastructure and does not run on customer content.

Microsoft antimalware workflow

The Azure service administrator can enable Antimalware for Azure with a default or custom configuration for your Virtual Machines and Cloud Services using the following options:

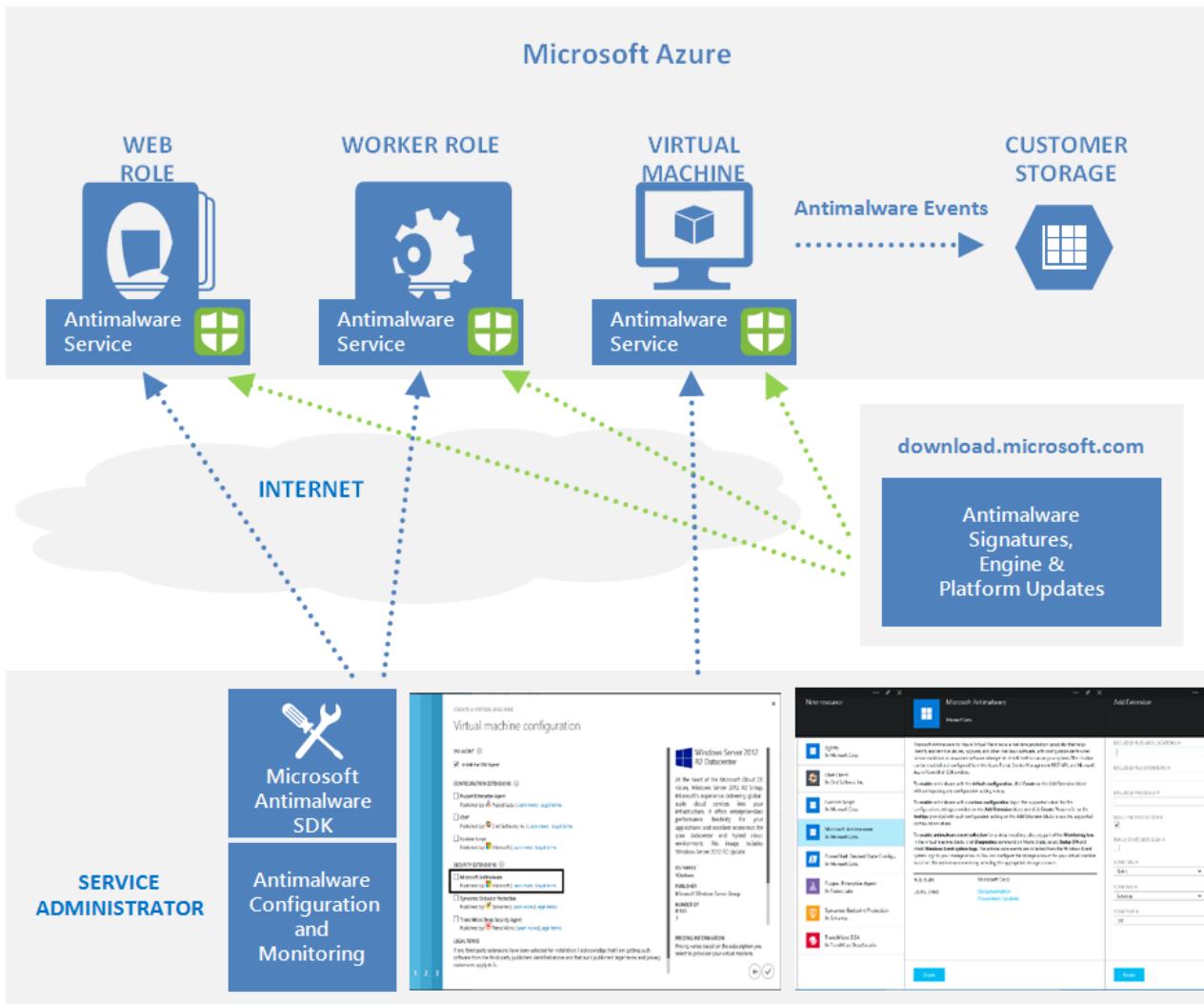
- Virtual Machines – In the Azure portal, under **Security Extensions**
- Virtual Machines – Using the Visual Studio virtual machines configuration in Server Explorer
- Virtual Machines and Cloud Services – Using the Antimalware [classic deployment model](#)
- Virtual Machines and Cloud Services – Using Antimalware PowerShell cmdlets

The Azure portal or PowerShell cmdlets push the Antimalware extension package file to the Azure system at a pre-determined fixed location. The Azure Guest Agent (or the Fabric Agent) launches the Antimalware Extension, applying the Antimalware configuration settings supplied as input. This step enables the Antimalware service with either default or custom configuration settings. If no custom configuration is provided, then the antimalware service is enabled with the default configuration settings. Refer to the *Antimalware configuration* section in the [Microsoft Antimalware for Azure – Code Samples](#) for more details.

Once running, the Microsoft Antimalware client downloads the latest protection engine and signature definitions from the Internet and loads them on the Azure system. The Microsoft Antimalware service writes service-related events to the system OS events log under the “Microsoft Antimalware” event source. Events include the Antimalware client health state, protection and remediation status, new and old configuration settings, engine updates and signature definitions, and others.

You can enable Antimalware monitoring for your Cloud Service or Virtual Machine to have the Antimalware event log events written as they are produced to your Azure storage account. The Antimalware Service uses the Azure Diagnostics extension to collect Antimalware events from the Azure system into tables in the customer’s Azure Storage account.

The deployment workflow including configuration steps and options supported for the above scenarios are documented in [Antimalware deployment scenarios](#) section of this document.



NOTE

You can however use Powershell/APIs and Azure Resource Manager templates to deploy Virtual Machine Scale Sets with the Microsoft Anti-Malware extension. For installing an extension on an already running Virtual Machine, you can use the sample python script `vmssextn.py` located [here](#). This script gets the existing extension config on the Scale Set and adds an extension to the list of existing extensions on the VM Scale Sets.

Default and Custom Antimalware Configuration

The default configuration settings are applied to enable Antimalware for Azure Cloud Services or Virtual Machines when you do not provide custom configuration settings. The default configuration settings have been pre-optimized for running in the Azure environment. Optionally, you can customize these default configuration settings as required for your Azure application or service deployment and apply them for other deployment scenarios.

NOTE

By default the Microsoft Antimalware User Interface on Azure Resource Manager is disabled, `cleanuppolicy.xml` file to bypass this error message is not supported. For information on how to create a custom policy, read [Enabling Microsoft Antimalware User Interface on Azure Resource Manager VMs Post Deployment](#).

The following table summarizes the configuration settings available for the Antimalware service. The default configuration settings are marked under the column labeled "Default" below.

Setting	Options	Default	Description
Enable Antimalware	true	None	true - Enables the Antimalware service

	(lower case sensitive)		false – not supported Note – This is a required configuration setting to enable the Antimalware service
Exclusions Extensions	extension1, extension2,	None	List of file extensions to exclude from scanning. Example: gif, log, txt excludes files with the .gif, .log, or .txt extension from being scanned. Each excluded file extension should be added as a separate row element value in your antimalware XML configuration or semicolon separated in antimalware JSON configuration
Exclusions Paths	path1, path2	None	List of paths to files or folders to exclude from scanning. Example: e:\approot\worker.dll, e:\approot\temp excludes the file worker.dll in the e:\approot folder and anything under the folder e:\approot\temp from being scanned. Note: For antimalware JSON configuration for virtual machines, use two backslashes (\\\) instead of one to escape properly. For example: e:\\approot\\worker.dll Each excluded path should be added as a separate row element value in your antimalware XML configuration or semicolon separated in antimalware JSON configuration
Exclusions Processes	process1, process2,	None	List of process exclusions. Any file opened by an excluded process will not be scanned (the process itself will still be scanned – to exclude the process itself, use the ExcludedPaths option). Example: C:\Program Files\MyApp.exe excludes any files opened by MyApp.exe from being scanned. Each excluded process should be added as a separate row element value in your antimalware XML configuration or semicolon separated in antimalware JSON configuration
RealtimeProtectionEnabled	true false (lower case sensitive)	true	true – Enables real-time protection false – Disables real-time protection Default = true when AntimalwareEnabled = true
ScheduledScanSettings isEnabled	true false (lower case sensitive)	false	Enables or disables a periodic scan for active malware on the system Default = false
ScheduledScanSettings Day	0 – 8	7	0 – scan daily, 1 – Sunday, 2 – Monday, 3 – Tuesday..., 7 – Saturday, 8 – disabled Default = 7 if only ScheduledScanSettings isEnabled = true
ScheduledScanSettings Time	0 – 1440	120	Hour at which to begin the scheduled scan. Measured in 60 minute increments from midnight 60 mins = 1:00 AM 120 mins = 2:00 AM

			<p>...</p> <p>1380 mins = 11:00 PM</p> <p>Default = 120 mins if ScheduledScanSettings isEnabled = true</p>
ScheduledScanSettings Scan Type	Quick/Full	Quick	Default = Quick if ScheduledScanSettings isEnabled = true
Monitoring	ON OFF	OFF	<p>ON - Enable Antimalware event collection to user subscription storage using Azure Diagnostics extension</p> <p>OFF – Disable Antimalware event collection to user subscription storage by removing antimalware monitoring configuration in Azure Diagnostics extension if it was previously turned ON</p>
StorageAccountName	Storage Account Name	None	Storage account name for your Azure store table to collect antimalware events in storage Note - Storage account name is required if monitoring is specified as ON

Antimalware Deployment Scenarios

The scenarios to enable and configure antimalware, including monitoring for Azure Cloud Services and Virtual Machines, are discussed in this section.

Virtual machines - enable and configure antimalware

Deployment using Azure Portal

To enable the Antimalware service, click **Add** on the Extensions blade, select **Microsoft Antimalware** on the New resource blade, click **Create** on the Microsoft Antimalware blade. Click **Create** without inputting any configuration values to enable Antimalware with the default settings, or enter the Antimalware configuration settings for the Virtual Machine configured as shown in Figure 2 below. Please refer to the **tooltips** provided with each configuration setting on the Add Extension blade to see the supported configuration values.

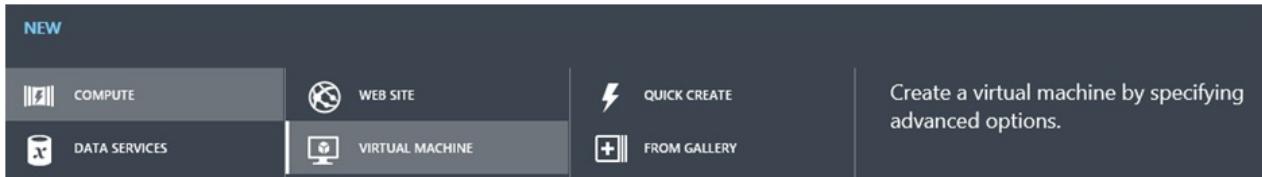
The screenshot shows the Azure portal interface for managing extensions. On the left, a sidebar lists various extensions like BgInfo, Chef Client, CloudLink SecureVM Agent, Custom Script, McAfee Endpoint Protection, Microsoft Antimalware (selected), Octopus Deploy Tentacle Agent, PowerShell Desired State Configuration, and Puppet Enterprise Agent. The central pane details the Microsoft Antimalware extension, its publisher (Microsoft Corp.), useful links (Documentation, Powershell Cmdlets), and a description of its real-time protection capabilities. The right pane is titled 'Add Extension' and contains configuration fields for 'EXCLUDED FILES AND LOCATIONS', 'EXCLUDED FILE EXTENSIONS', 'EXCLUDED PROCESSES', 'REAL-TIME PROTECTION' (with a checked checkbox), 'RUN A SCHEDULED SCAN', 'SCAN TYPE' (set to 'Quick'), 'SCAN DAY' (set to 'Saturday'), and 'SCAN TIME' (set to '120'). At the bottom of the central and right panes are 'Create' buttons.

Deployment using the Azure classic portal

To enable and configure Microsoft Antimalware for Azure Virtual Machines using the Azure portal while provisioning a Virtual Machine, follow the steps below:

1. Log onto the Azure portal at <https://portal.azure.com>

2. To create a new virtual machine, click **New, Compute, Virtual Machine, From Gallery** (do not use Quick Create) as shown below:



3. Select the **Microsoft Windows Server** image on the **Choose an Image** page.

4. Click the right arrow and input the Virtual Machine configuration.

5. Check the **Microsoft Antimalware** checkbox under **Security Extensions** on the Virtual Machine configuration page.

6. Click the Submit button to enable and configure Microsoft Antimalware for Azure Virtual Machines with the default configuration settings.

CREATE A VIRTUAL MACHINE

Virtual machine configuration

VM AGENT 

Install the VM Agent

CONFIGURATION EXTENSIONS 

Puppet Enterprise Agent
Published by:  Puppet Labs | [Learn more](#) | Legal terms

Chef
Published by:  Chef Software, Inc. | [Learn more](#) | Legal terms

Custom Script
Published by:  Microsoft | [Learn more](#) | Legal terms

SECURITY EXTENSIONS 

Microsoft Antimalware
Published by:  Microsoft | [Learn more](#) | Legal terms

Symantec Endpoint Protection
Published by:  Symantec | [Learn more](#) | Legal terms

Trend Micro Deep Security Agent
Published by:  Trend Micro | [Learn more](#) | Legal terms

LEGAL TERMS

If any third-party extensions have been selected for installation, I acknowledge that I am getting such software from the third-party publishers identified above and that such publishers' legal terms and privacy statements apply to it.

 Windows Server 2012 R2 Datacenter

At the heart of the Microsoft Cloud OS vision, Windows Server 2012 R2 brings Microsoft's experience delivering global-scale cloud services into your infrastructure. It offers enterprise-class performance, flexibility for your applications and excellent economics for your datacenter and hybrid cloud environment. This image includes Windows Server 2012 R2 Update.

OS FAMILY
Windows

PUBLISHER
Microsoft Windows Server Group

NUMBER OF DISKS
1

PRICING INFORMATION
Pricing varies based on the subscription you select to provision your virtual machine.

Deployment Using the Visual Studio virtual machine configuration

To enable and configure the Microsoft Antimalware service using Visual Studio:

1. Connect to Microsoft Azure in Visual Studio.
 2. Choose your Virtual Machine in the **Virtual Machines** node in **Server Explorer**.

Configuration

Virtual Machine

Shutdown Connect... Capture Image... Update Refresh

Settings

Status: Started Location: [redacted]

DNS Name: cloudapp.net Deployment Name: [redacted]

Subscription ID: [redacted]

Size: Small (1 cores, 1792 MB) Availability Set: [(none)]

Public Endpoints

Port Name	Public Port	Private Port	Protocol	Load Balance Set
PowerShell	5986	5986	TCP	
Remote Desktop	50669	3389	TCP	

Select an endpoint... Add Remove

Installed Extensions

Name	Publisher	Version	Enabled
Windows Azure BGInfo Extension for IaaS	Microsoft.Compute	1.*	<input checked="" type="checkbox"/>

Microsoft Antimalware Add Remove Configure...

Azure Cloud Services Notification Hubs Service Bus SQL Databases Storage Virtual Machines PowerShell Remote Desktop

- 3.Right click **configure** to view the Virtual Machine configuration page

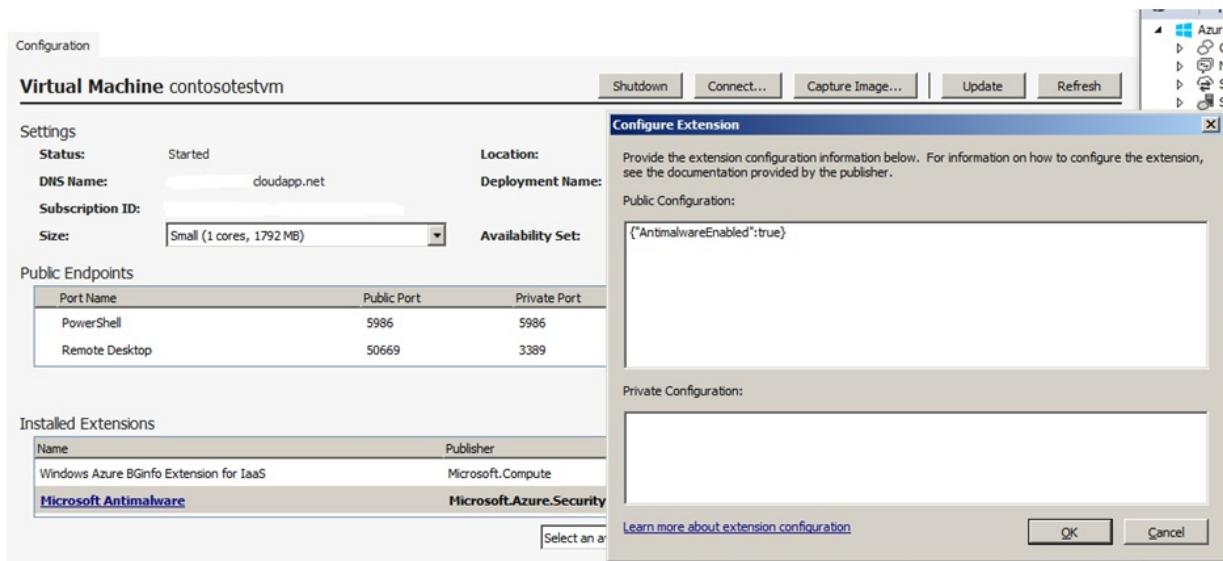
4. Select **Microsoft Antimalware** extension from the dropdown list under **Installed Extensions** and click **Add** to configure with default antimalware configuration.

Installed Extensions				
Name	Publisher	Version	Enabled	
Windows Azure BGInfo Extension for IaaS	Microsoft.Compute	1.*	<input checked="" type="checkbox"/>	
Microsoft Antimalware	Microsoft.Azure.Security	1.1	<input checked="" type="checkbox"/>	

- 5.To customize the default Antimalware configuration, select (highlight) the Antimalware extension in the installed extensions list and click **Configure**.

6.Replace the default Antimalware configuration with your custom configuration in supported JSON format in the **public configuration** textbox and click OK.

7.Click the **Update** button to push the configuration updates to your Virtual Machine.



Note: The Visual Studio Virtual Machines configuration for Antimalware supports only JSON format configuration. The Antimalware JSON configuration settings template is included in the [Microsoft Antimalware For Azure - Code Samples](#), showing the supported Antimalware configuration settings.

Deployment Using PowerShell cmdlets

An Azure application or service can enable and configure Microsoft Antimalware for Azure Virtual Machines using PowerShell cmdlets.

To enable and configure Microsoft antimalware using antimalware PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-powershell>
2. Use the Set-AzureVMMicrosoftAntimalwareExtension Antimalware cmdlet to enable and configure Microsoft Antimalware for your Virtual Machine as documented at <http://msdn.microsoft.com/library/azure/dn771718.aspx>

Note: The Azure Virtual Machines configuration for Antimalware supports only JSON format configuration. The Antimalware JSON configuration settings template is included in the [Microsoft Antimalware For Azure - Code Samples](#), showing the supported Antimalware configuration settings.

Enable and Configure Antimalware Using PowerShell cmdlets

An Azure application or service can enable and configure Microsoft Antimalware for Azure Cloud Services using PowerShell cmdlets. Note that Microsoft Antimalware is installed in a disabled state in the Cloud Services platform and requires an action by an Azure application to enable it.

To enable and configure Microsoft Antimalware using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-sdk-tools#get-started>
2. Use the Set-AzureServiceAntimalwareExtension Antimalware cmdlet to enable and configure Microsoft Antimalware for your Cloud Service as documented at <http://msdn.microsoft.com/library/azure/dn771718.aspx>

The Antimalware XML configuration settings template is included in the [Microsoft Antimalware For Azure - Code Samples](#), showing the supported Antimalware configuration settings.

Cloud Services and Virtual Machines - Configuration Using PowerShell cmdlets

An Azure application or service can retrieve the Microsoft Antimalware configuration for Cloud Services and Virtual Machines using PowerShell cmdlets.

To retrieve the Microsoft Antimalware configuration using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-sdk-tools#get-started>
2. **For Virtual Machines:** Use the Get-AzureVMMicrosoftAntimalwareExtension Antimalware cmdlet to get the antimalware configuration as documented at <http://msdn.microsoft.com/library/azure/dn771719.aspx>
3. **For Cloud Services:** Use the Get-AzureServiceAntimalwareConfig Antimalware cmdlet to get the Antimalware configuration as documented at <http://msdn.microsoft.com/library/azure/dn771722.aspx>

Remove Antimalware Configuration Using PowerShell cmdlets

An Azure application or service can remove the Antimalware configuration and any associated Antimalware monitoring configuration from the relevant Azure Antimalware and diagnostics service extensions associated with the Cloud Service or Virtual Machine.

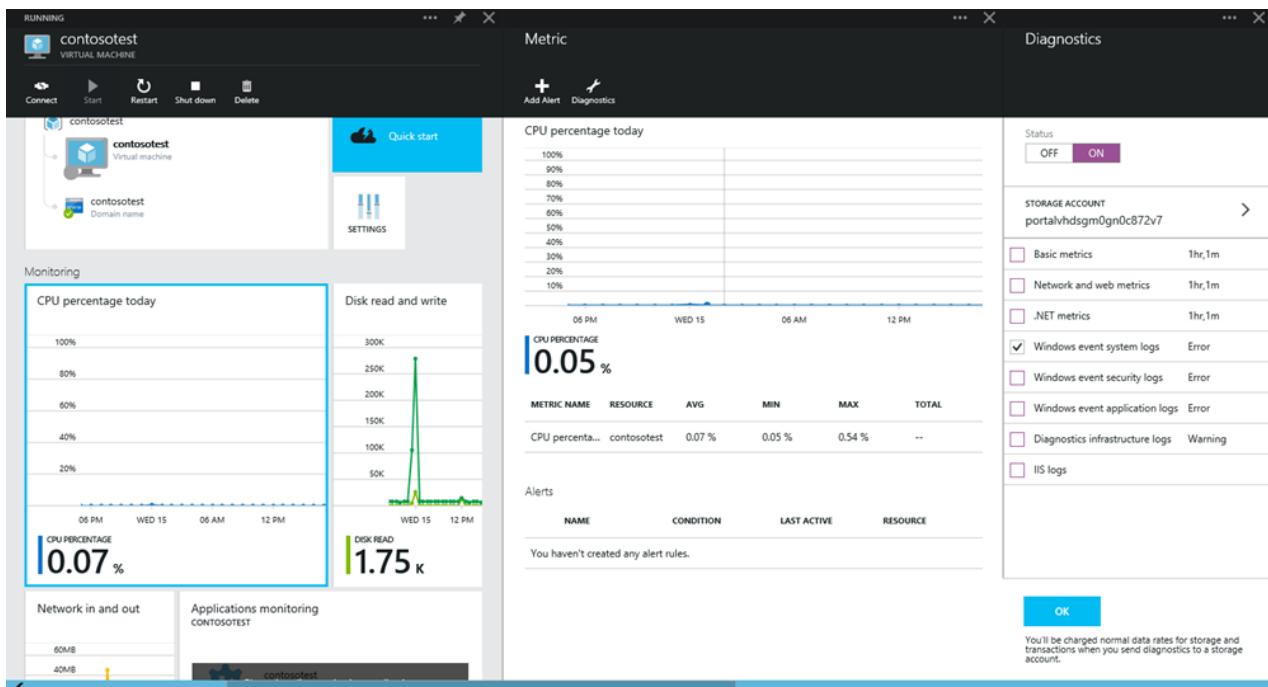
To remove Microsoft Antimalware using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-sdk-tools#get-started>
2. **For Virtual Machines:** Use the Remove-AzureVMMicrosoftAntimalwareExtension Antimalware cmdlet as documented at <http://msdn.microsoft.com/library/azure/dn771720.aspx>
3. **For Cloud Services:** Use the Remove-AzureServiceAntimalwareExtension Antimalware cmdlet as documented at <http://msdn.microsoft.com/library/azure/dn771717.aspx>

To **enable** antimalware event collection for a virtual machine using the Azure Preview Portal:

1. Click any part of the Monitoring lens in the Virtual Machine blade
2. Click the Diagnostics command on Metric blade
3. Select **Status ON** and check the option for Windows event system
4. You can choose to uncheck all other options in the list, or leave them enabled per your application service needs.
5. The Antimalware event categories "Error", "Warning", "Informational", etc., are captured in your Azure Storage account.

Antimalware events are collected from the Windows event system logs to your Azure Storage account. You can configure the Storage Account for your Virtual Machine to collect Antimalware events by selecting the appropriate storage account.



NOTE

For more information on how to Diagnostics Logging for Azure Antimalware, read [Enabling Diagnostics Logging for Azure Antimalware](#).

Enable and configure antimalware monitoring using PowerShell cmdlets

You can enable collection of Microsoft Antimalware events for your Cloud Service or Virtual Machine using Azure Diagnostics through Antimalware PowerShell cmdlets. The Azure Diagnostics extension can be configured to capture events from the System event log source "Microsoft Antimalware" to your Azure Storage account. The Antimalware event categories "Error", "Warning", "Informational", etc., are captured in your Azure Storage account.

To enable Antimalware event collection to your Azure Storage account using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to <https://github.com/Azure/azure-sdk-tools#get-started>
2. **For Virtual Machines** - Use the Set-AzureVMMicrosoftAntimalwareExtension Antimalware cmdlet with Monitoring ON option as documented at <http://msdn.microsoft.com/library/azure/dn771716.aspx>
3. **For Cloud Services** - Use the Set-AzureServiceAntimalwareExtension Antimalware cmdlet with Monitoring ON option as documented at <http://msdn.microsoft.com/library/azure/dn771718.aspx>

You can view the Antimalware raw events by looking at the WADWindowsEventLogsTable table in your Azure Storage account that you configured to enable Antimalware monitoring. This can be useful to validate that Antimalware event collection is working, including getting insight into the Antimalware service's health. For more details, including sample code on how to extract Antimalware events from your storage account, refer to [Microsoft Antimalware For Azure - Code Samples](#).

Azure Disk Encryption for Windows and Linux IaaS VMs

8/9/2017 • 49 min to read • [Edit Online](#)

Microsoft Azure is strongly committed to ensuring your data privacy, data sovereignty and enables you to control your Azure hosted data through a range of advanced technologies to encrypt, control and manage encryption keys, control & audit access of data. This provides Azure customers the flexibility to choose the solution that best meets their business needs. In this paper, we will introduce you to a new technology solution "Azure Disk Encryption for Windows and Linux IaaS VM's" to help protect and safeguard your data to meet your organizational security and compliance commitments. The paper provides detailed guidance on how to use the Azure disk encryption features including the supported scenarios and the user experiences.

NOTE

Certain recommendations might increase data, network, or compute resource usage, resulting in additional license or subscription costs.

Overview

Azure Disk Encryption is a new capability that helps you encrypt your Windows and Linux IaaS virtual machine disks. Azure Disk Encryption leverages the industry standard [BitLocker](#) feature of Windows and the [DM-Crypt](#) feature of Linux to provide volume encryption for the OS and the data disks. The solution is integrated with [Azure Key Vault](#) to help you control and manage the disk-encryption keys and secrets in your key vault subscription. The solution also ensures that all data on the virtual machine disks are encrypted at rest in your Azure storage.

Azure disk encryption for Windows and Linux IaaS VMs is now in **General Availability** in all Azure public regions and AzureGov regions for Standard VMs and VMs with premium storage.

Encryption scenarios

The Azure Disk Encryption solution supports the following customer scenarios:

- Enable encryption on new IaaS VMs created from pre-encrypted VHD and encryption keys
- Enable encryption on new IaaS VMs created from the supported Azure Gallery images
- Enable encryption on existing IaaS VMs running in Azure
- Disable encryption on Windows IaaS VMs
- Disable encryption on data drives for Linux IaaS VMs
- Enable encryption of managed disk VMs
- Update encryption settings of an existing encrypted non-premium storage VM
- Backup and restore of encrypted VMs, encrypted with key encryption key

The solution supports the following scenarios for IaaS VMs when they are enabled in Microsoft Azure:

- Integration with Azure Key Vault
- Standard tier VMs: [A](#), [D](#), [DS](#), [G](#), [GS](#), [F](#), and so forth series IaaS VMs
- Enable encryption on Windows and Linux IaaS VMs and managed disk VMs from the supported Azure Gallery images
- Disable encryption on OS and data drives for Windows IaaS VMs and managed disk VMs
- Disable encryption on data drives for Linux IaaS VMs and managed disk VMs

- Enable encryption on IaaS VMs running Windows Client OS
- Enable encryption on volumes with mount paths
- Enable encryption on Linux VMs configured with disk striping (RAID) using mdadm
- Enable encryption on Linux VMs using LVM for data disks
- Enable encryption on Windows VMs configured with Storage Spaces
- Update encryption settings of an existing encrypted non-premium storage VM
- All Azure Public and AzureGov regions are supported

The solution does not support the following scenarios, features, and technology:

- Basic tier IaaS VMs
- Disabling encryption on an OS drive for Linux IaaS VMs
- Disabling encryption on a data drive if the OS drive is encrypted for Linux IaaS VMs
- IaaS VMs that are created by using the classic VM creation method
- Enable encryption on Windows and Linux IaaS VMs customer custom images is NOT supported. Enable encryption on Linux LVM OS disk is not supported currently. This support will come soon.
- Integration with your on-premises Key Management Service
- Azure Files (shared file system), Network File System (NFS), dynamic volumes, and Windows VMs that are configured with software-based RAID systems
- Backup and restore of encrypted VMs, encrypted without key encryption key.
- Update encryption settings of an existing encrypted premium storage VM.

NOTE

Backup and restore of encrypted VMs is supported only for VMs that are encrypted with the KEK configuration. It is not supported on VMs that are encrypted without KEK. KEK is an optional parameter that enables VM encryption. This support is coming soon. Update encryption settings of an existing encrypted premium storage VM are not supported. This support is coming soon.

Encryption features

When you enable and deploy Azure Disk Encryption for Azure IaaS VMs, the following capabilities are enabled, depending on the configuration provided:

- Encryption of the OS volume to protect the boot volume at rest in your storage
- Encryption of data volumes to protect the data volumes at rest in your storage
- Disabling encryption on the OS and data drives for Windows IaaS VMs
- Disabling encryption on the data drives for Linux IaaS VMs (only if OS drive IS NOT encrypted)
- Safeguarding the encryption keys and secrets in your key vault subscription
- Reporting the encryption status of the encrypted IaaS VM
- Removal of disk-encryption configuration settings from the IaaS virtual machine
- Backup and restore of encrypted VMs by using the Azure Backup service

NOTE

Backup and restore of encrypted VMs is supported only for VMs that are encrypted with the KEK configuration. It is not supported on VMs that are encrypted without KEK. KEK is an optional parameter that enables VM encryption.

Azure Disk Encryption for IaaS VMS for Windows and Linux solution includes:

- The disk-encryption extension for Windows.
- The disk-encryption extension for Linux.

- The disk-encryption PowerShell cmdlets.
- The disk-encryption Azure command-line interface (CLI) cmdlets.
- The disk-encryption Azure Resource Manager templates.

The Azure Disk Encryption solution is supported on IaaS VMs that are running Windows or Linux OS. For more information about the supported operating systems, see the "Prerequisites" section.

NOTE

There is no additional charge for encrypting VM disks with Azure Disk Encryption.

Value proposition

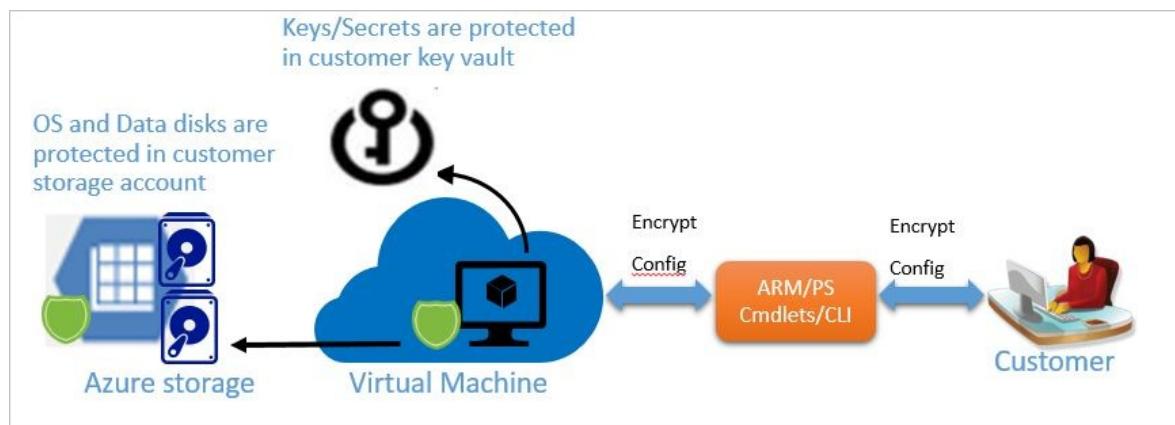
When you apply the Azure Disk Encryption-management solution, you can satisfy the following business needs:

- IaaS VMs are secured at rest, because you can use industry-standard encryption technology to address organizational security and compliance requirements.
- IaaS VMs boot under customer-controlled keys and policies, and you can audit their usage in your key vault.

Encryption workflow

To enable disk encryption for Windows and Linux VMs, do the following:

1. Choose an encryption scenario from among the preceding encryption scenarios.
2. Opt in to enabling disk encryption via the Azure Disk Encryption Resource Manager template, PowerShell cmdlets, or CLI command, and specify the encryption configuration.
 - For the customer-encrypted VHD scenario, upload the encrypted VHD to your storage account and the encryption key material to your key vault. Then, provide the encryption configuration to enable encryption on a new IaaS VM.
 - For new VMs that are created from the Marketplace and existing VMs that are already running in Azure, provide the encryption configuration to enable encryption on the IaaS VM.
3. Grant access to the Azure platform to read the encryption-key material (BitLocker encryption keys for Windows systems and Passphrase for Linux) from your key vault to enable encryption on the IaaS VM.
4. Provide the Azure Active Directory (Azure AD) application identity to write the encryption key material to your key vault. Doing so enables encryption on the IaaS VM for the scenarios mentioned in step 2.
5. Azure updates the VM service model with encryption and the key vault configuration, and sets up your encrypted VM.



Decryption workflow

To disable disk encryption for IaaS VMs, complete the following high-level steps:

1. Choose to disable encryption (decryption) on a running IaaS VM in Azure via the Azure Disk Encryption

Resource Manager template or PowerShell cmdlets, and specify the decryption configuration.

This step disables encryption of the OS or the data volume or both on the running Windows IaaS VM. However, as mentioned in the previous section, disabling OS disk encryption for Linux is not supported. The decryption step is allowed only for data drives on Linux VMs as long as the OS disk is not encrypted.

2. Azure updates the VM service model, and the IaaS VM is marked decrypted. The contents of the VM are no longer encrypted at rest.

NOTE

The disable-encryption operation does not delete your key vault and the encryption key material (BitLocker encryption keys for Windows systems or Passphrase for Linux). Disabling OS disk encryption for Linux is not supported. The decryption step is allowed only for data drives on Linux VMs. Disabling data disk encryption for Linux is not supported if the OS drive is encrypted.

Prerequisites

Before you enable Azure Disk Encryption on Azure IaaS VMs for the supported scenarios that were discussed in the "Overview" section, see the following prerequisites:

- You must have a valid active Azure subscription to create resources in Azure in the supported regions.
- Azure Disk Encryption is supported on the following Windows Server versions: Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016.
- Azure Disk Encryption is supported on the following Windows client versions: Windows 8 client and Windows 10 client.

NOTE

For Windows Server 2008 R2, you must have .NET Framework 4.5 installed before you enable encryption in Azure. You can install it from Windows Update by installing the optional update Microsoft .NET Framework 4.5.2 for Windows Server 2008 R2 x64-based systems ([KB2901983](#)).

- Azure Disk Encryption is supported on the following Azure Gallery based Linux server distributions and versions:

LINUX DISTRIBUTION	VERSION	VOLUME TYPE SUPPORTED FOR ENCRYPTION
Ubuntu	16.04-DAILY-LTS	OS and Data disk
Ubuntu	14.04.5-DAILY-LTS	OS and Data disk
Ubuntu	12.10	Data disk
Ubuntu	12.04	Data disk
RHEL	7.3	OS and Data disk
RHEL	7.2	OS and Data disk
RHEL	6.8	OS and Data disk

LINUX DISTRIBUTION	VERSION	VOLUME TYPE SUPPORTED FOR ENCRYPTION
RHEL	6.7	Data disk
CentOS	7.3	OS and Data disk
CentOS	7.2n	OS and Data disk
CentOS	6.8	OS and Data disk
CentOS	7.1	Data disk
CentOS	7.0	Data disk
CentOS	6.7	Data disk
CentOS	6.6	Data disk
CentOS	6.5	Data disk
openSUSE	13.2	Data disk
SLES	12 SP1	Data disk
SLES	12-SP1 (Premium)	Data disk
SLES	HPC 12	Data disk
SLES	11-SP4 (Premium)	Data disk
SLES	11 SP4	Data disk

- Azure Disk Encryption requires that your key vault and VMs reside in the same Azure region and subscription.

NOTE

Configuring the resources in separate regions causes a failure in enabling the Azure Disk Encryption feature.

- To set up and configure your key vault for Azure Disk Encryption, see section **Set up and configure your key vault for Azure Disk Encryption** in the *Prerequisites* section of this article.
- To set up and configure Azure AD application in Azure Active directory for Azure Disk Encryption, see section **Set up the Azure AD application in Azure Active Directory** in the *Prerequisites* section of this article.
- To set up and configure the key vault access policy for the Azure AD application, see section **Set up the key vault access policy for the Azure AD application** in the *Prerequisites* section of this article.
- To prepare a pre-encrypted Windows VHD, see section **Prepare a pre-encrypted Windows VHD** in the *Appendix*.
- To prepare a pre-encrypted Linux VHD, see section **Prepare a pre-encrypted Linux VHD** in the *Appendix*.
- The Azure platform needs access to the encryption keys or secrets in your key vault to make them available to the virtual machine when it boots and decrypts the virtual machine OS volume. To grant permissions to Azure platform, set the **EnabledForDiskEncryption** property in the key vault. For more information, see **Set up and configure your key vault for Azure Disk Encryption** in the *Appendix*.

- Your key vault secret and KEK URLs must be versioned. Azure enforces this restriction of versioning. For valid secret and KEK URLs, see the following examples:
 - Example of a valid secret URL:
<https://contosovault.vault.azure.net/secrets/BitLockerEncryptionSecretWithKek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - Example of a valid KEK URL:
<https://contosovault.vault.azure.net/keys/diskencryptionkek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
- Azure Disk Encryption does not support specifying port numbers as part of key vault secrets and KEK URLs. For examples of non-supported and supported key vault URLs, see the following:
 - Unacceptable key vault URL
<https://contosovault.vault.azure.net:443/secrets/contososecret/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - Acceptable key vault URL:
<https://contosovault.vault.azure.net/secrets/contososecret/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
- To enable the Azure Disk Encryption feature, the IaaS VMs must meet the following network endpoint configuration requirements:
 - To get a token to connect to your key vault, the IaaS VM must be able to connect to an Azure Active Directory endpoint, [login.microsoftonline.com].
 - To write the encryption keys to your key vault, the IaaS VM must be able to connect to the key vault endpoint.
 - The IaaS VM must be able to connect to an Azure storage endpoint that hosts the Azure extension repository and an Azure storage account that hosts the VHD files.

NOTE

If your security policy limits access from Azure VMs to the Internet, you can resolve the preceding URI and configure a specific rule to allow outbound connectivity to the IPs.

To configure and access Azure Key Vault behind a firewall(<https://docs.microsoft.com/en-us/azure/key-vault/key-vault-access-behind-firewall>)

- Use the latest version of Azure PowerShell SDK version to configure Azure Disk Encryption. Download the latest version of [Azure PowerShell release](#)

NOTE

Azure Disk Encryption is not supported on [Azure PowerShell SDK version 1.1.0](#). If you are receiving an error related to using Azure PowerShell 1.1.0, see [Azure Disk Encryption Error Related to Azure PowerShell 1.1.0](#).

- To run any Azure CLI command and associate it with your Azure subscription, you must first install Azure CLI:
 - To install Azure CLI and associate it with your Azure subscription, see [How to install and configure Azure CLI](#).
 - To use Azure CLI for Mac, Linux, and Windows with Azure Resource Manager, see [Azure CLI commands in Resource Manager mode](#).
- When encrypting a managed disk, it is mandatory prerequisite to take a snapshot of the managed disk or a backup of the disk outside of Azure Disk Encryption prior to enabling encryption. Without a backup in place, any unexpected failure during encryption may render the disk and VM inaccessible without a recovery option. Set-AzureRmVMDiskEncryptionExtension does not currently back up managed disks and will error if used against a managed disk unless the -skipVmBackup parameter has been specified. This parameter is unsafe to use unless a backup has already been made outside of Azure Disk Encryption.

When the `-skipVmBackup` parameter is specified, the cmdlet will not make a backup of the managed disk prior to encryption. For this reason, it is considered a mandatory prerequisite to make sure a backup of the managed disk VM is in place prior to enabling Azure Disk Encryption in case recovery is later needed.

NOTE

The `-skipVmBackup` parameter should never be used unless a snapshot or backup has already been made outside of Azure Disk Encryption.

- The Azure Disk Encryption solution uses the BitLocker external key protector for Windows IaaS VMs. For domain joined VMs, DO NOT push any group policies that enforce TPM protectors. For information about the group policy for "Allow BitLocker without a compatible TPM," see [BitLocker Group Policy Reference](#).
- Bitlocker policy on domain joined virtual machines with custom group policy must include the following setting: `Configure user storage of bitlocker recovery information -> Allow 256-bit recovery key`. Azure Disk Encryption will fail when custom group policy settings for Bitlocker are incompatible. On machines that did not have the correct policy setting, applying the new policy, forcing the new policy to update (`gpupdate.exe /force`), and then restarting may be required.
- To create an Azure AD application, create a key vault, or set up an existing key vault and enable encryption, see the [Azure Disk Encryption prerequisite PowerShell script](#).
- To configure disk-encryption prerequisites using the Azure CLI, see [this Bash script](#).
- To use the Azure Backup service to back up and restore encrypted VMs, when encryption is enabled with Azure Disk Encryption, encrypt your VMs by using the Azure Disk Encryption key configuration. The Backup service supports VMs that are encrypted using KEK configuration only. See [How to back up and restore encrypted virtual machines with Azure Backup encryption](#).
- When encrypting a Linux OS volume, note that a VM restart is currently required at the end of the process. This can be done via the portal, powershell, or CLI. To track the progress of encryption, periodically poll the status message returned by `Get-AzureRmVMDiskEncryptionStatus` <https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/get-azurermvmdiskencryptionstatus>. Once encryption is complete, the the status message returned by this command will indicate this. For example, "ProgressMessage: OS disk successfully encrypted, please reboot the VM" At this point the VM can be restarted and used.
- Azure Disk Encryption for Linux requires data disks to have a mounted file system in Linux prior to encryption
- Recursively mounted data disks are not supported by the Azure Disk Encryption for Linux. For example, if the target system has mounted a disk on `/foo/bar` and then another on `/foo/bar/baz`, the encryption of `/foo/bar/baz` will succeed, but encryption of `/foo/bar` will fail.
- Azure Disk Encryption is only supported on Azure gallery supported images that meet the aforementioned prerequisites. Customer custom images are not supported due to custom partition schemes and process behaviors that may exist on these images. Further, even gallery image based VM's that initially met prerequisites but have been modified after creation may be incompatible. For that reason, the suggested procedure for encrypting a Linux VM is to start from a clean gallery image, encrypt the VM, and then add custom software or data to the VM as needed.

NOTE

Backup and restore of encrypted VMs is supported only for VMs that are encrypted with the KEK configuration. It is not supported on VMs that are encrypted without KEK. KEK is an optional parameter that enables VM.

When you need encryption to be enabled on a running VM in Azure, Azure Disk Encryption generates and writes the encryption keys to your key vault. Managing encryption keys in your key vault requires Azure AD authentication.

For this purpose, create an Azure AD application. You can find detailed steps for registering an application in the "Get an Identity for the Application" section of the blog post [Azure Key Vault - Step by Step](#). This post also contains a number of helpful examples for setting up and configuring your key vault. For authentication purposes, you can use either client secret-based authentication or client certificate-based Azure AD authentication.

Client secret-based authentication for Azure AD

The sections that follow can help you configure a client secret-based authentication for Azure AD.

Create an Azure AD application by using Azure PowerShell

Use the following PowerShell cmdlet to create an Azure AD application:

```
$aadClientSecret = "yourSecret"  
$azureAdApplication = New-AzureRmADApplication -DisplayName "<Your Application Display Name>" -HomePage "  
<https://YourApplicationHomePage>" -IdentifierUris "<https://YouApplicationUri>" -Password $aadClientSecret  
$servicePrincipal = New-AzureRmADServicePrincipal -ApplicationId $azureAdApplication.ApplicationId
```

NOTE

\$azureAdApplication.ApplicationId is the Azure AD ClientID and \$aadClientSecret is the client secret that you should use later to enable Azure Disk Encryption. Safeguard the Azure AD client secret appropriately.

Setting up the Azure AD client ID and secret from the Azure classic portal

You can also set up your Azure AD client ID and secret by using the [Azure classic portal](#). To perform this task, do the following:

1. Click the **Active Directory** tab.

NAME	STATUS	ROLE	SUBSCRIPTION	DATACENTER REGION	COUNTRY OR REGI...
Microsoft	Active	User	Shared by all Microsoft su...	United States	United States

2. Click **Add Application**, and then type the application name.

ADD APPLICATION

Tell us about your application

NAME

testkv1 X

Type

WEB APPLICATION AND/OR WEB API ?

NATIVE CLIENT APPLICATION ?

→ 2

3. Click the arrow button, and then configure the application properties.

ADD APPLICATION

X

App properties

SIGN-ON URL ?

https://testkv1.azure.com



APP ID URI ?

https://testkv1.azure.com



1



4. Click the check mark in the lower left corner to finish. The application configuration page appears, and the Azure AD client ID is displayed at the bottom of the page.

testkv1

DASHBOARD USERS CONFIGURE OWNERS

properties

NAME

testkv1

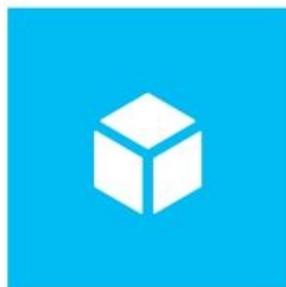


SIGN-ON URL

https://testkv1.azure.com



LOGO



APPLICATION IS MULTI-TENANT

YES

NO



CLIENT ID

2fdcce7-f4ee-47c8-9469-72bcfc639f



USER ASSIGNMENT REQUIRED TO
ACCESS APP

YES

NO



5. Save the Azure AD client secret by clicking the **Save** button. Note the Azure AD client secret in the keys text box. Safeguard it appropriately.

keys

2 years 10/14/2015 10/14/2017

2 years 10/14/2015 10/14/2017 THE KEY VALUE WILL BE DISPLAYED AFTER YOU SAVE IT.

Select du... VALID FROM EXPIRES ON THE KEY VALUE WILL BE DISPLAYED AFTER YOU SAVE IT.

Copy and store the key value. You won't be able to retrieve it after you leave this page.

single sign-on

APP ID URI

REPLY URL
(ENTER A REPLY URL)

permissions to other applications

Windows Azure Active Directory Application Permissions: 0 Delegated Permissions: 1

VIEW ENDPOINTS UPLOAD LOGO MANAGE MANIFEST DELETE SAVE DISCARD

NOTE

The preceding flow is not supported on the Azure classic portal.

Use an existing application

To execute the following commands, obtain and use the [Azure AD PowerShell module](#).

NOTE

The following commands must be executed from a new PowerShell window. Do not use Azure PowerShell or the Azure Resource Manager window to execute the commands. We recommend this approach because these cmdlets are in the MSOnline module or Azure AD PowerShell.

```
$clientSecret = '<yourAadClientSecret>'  
$aadClientID = '<Client ID of your Azure AD application>'  
connect-msolservice  
New-MsolServicePrincipalCredential -AppPrincipalId $aadClientID -Type password -Value $clientSecret
```

Certificate-based authentication for Azure AD

NOTE

Azure AD certificate-based authentication is currently not supported on Linux VMs.

The sections that follow show how to configure a certificate-based authentication for Azure AD.

Create an Azure AD application

To create an Azure AD application, execute the following PowerShell cmdlets:

NOTE

Replace the following `yourpassword` string with your secure password, and safeguard the password.

```
$cert = New-Object  
System.Security.Cryptography.X509Certificates.X509Certificate("C:\certificates\examplecert.pfx",  
"yourpassword")  
$keyValue = [System.Convert]::ToBase64String($cert.GetRawCertData())  
$azureAdApplication = New-AzureRmADApplication -DisplayName "<Your Application Display Name>" -HomePage "  
<https://YourApplicationHomePage>" -IdentifierUris "<https://YouApplicationUri>" -KeyValue $keyValue -KeyType  
AsymmetricX509Cert  
$servicePrincipal = New-AzureRmADServicePrincipal -ApplicationId $azureAdApplication.ApplicationId
```

After you finish this step, upload a PFX file to your key vault and enable the access policy needed to deploy that certificate to a VM.

Use an existing Azure AD application

If you are configuring certificate-based authentication for an existing application, use the PowerShell cmdlets shown here. Be sure to execute them from a new PowerShell window.

```
$certLocalPath = 'C:\certs\myaadapp.cer'  
$aadClientID = '<Client ID of your Azure AD application>'  
connect-msolservice  
$cer = New-Object System.Security.Cryptography.X509Certificates.X509Certificate  
$cer.Import($certLocalPath)  
$binCert = $cer.GetRawCertData()  
$credValue = [System.Convert]::ToBase64String($binCert);  
New-MsolServicePrincipalCredential -AppPrincipalId $aadClientID -Type asymmetric -Value $credValue -Usage  
verify
```

After you finish this step, upload a PFX file to your key vault and enable the access policy that's needed to deploy the certificate to a VM.

Upload a PFX file to your key vault

For a detailed explanation of this process, see [The Official Azure Key Vault Team Blog](#). However, the following PowerShell cmdlets are all you need for the task. Be sure to execute them from Azure PowerShell console.

NOTE

Replace the following `yourpassword` string with your secure password, and safeguard the password.

```

$certLocalPath = 'C:\certs\myaadapp.pfx'
$certPassword = "yourpassword"
$resourceGroupName = 'yourResourceGroup'
$keyVaultName = 'yourKeyVaultName'
$keyVaultSecretName = 'yourAadCertSecretName'

$fileContentBytes = get-content $certLocalPath -Encoding Byte
$fileContentEncoded = [System.Convert]::ToBase64String($fileContentBytes)

$jsonObject = @"
{
  "data": "$fileContentEncoded",
  "dataType": "pfx",
  "password": "$certPassword"
}
"@

$jsonObjectBytes = [System.Text.Encoding]::UTF8.GetBytes($jsonObject)
$jsonEncoded = [System.Convert]::ToBase64String($jsonObjectBytes)

Switch-AzureMode -Name AzureResourceManager
$secret = ConvertTo-SecureString -String $jsonEncoded -AsPlainText -Force
Set-AzureKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName -SecretValue $secret
Set-AzureRmKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $resourceGroupName -EnabledForDeployment

```

Deploy a certificate in your key vault to an existing VM

After you finish uploading the PFX, deploy a certificate in the key vault to an existing VM with the following:

```

$resourceGroupName = 'yourResourceGroup'
$keyVaultName = 'yourKeyVaultName'
$keyVaultSecretName = 'yourAadCertSecretName'
$vmName = 'yourVMName'
$certUrl = (Get-AzureKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName).Id
$sourceVaultId = (Get-AzureRmKeyVault -VaultName $keyVaultName -ResourceGroupName $resourceGroupName).ResourceId
$vm = Get-AzureRmVM -ResourceGroupName $resourceGroupName -Name $vmName
$vm = Add-AzureRmVMSecret -VM $vm -SourceVaultId $sourceVaultId -CertificateStore "My" -CertificateUrl $certUrl
Update-AzureRmVM -VM $vm -ResourceGroupName $resourceGroupName

```

Set up the key vault access policy for the Azure AD application

Your Azure AD application needs rights to access the keys or secrets in the vault. Use the

[Set-AzureKeyVaultAccessPolicy](#) cmdlet to grant permissions to the application, using the client ID (which was generated when the application was registered) as the `-ServicePrincipalName` parameter value. To learn more, see the blog post [Azure Key Vault - Step by Step](#). Here is an example of how to perform this task via PowerShell:

```

$keyVaultName = '<yourKeyVaultName>'
$aadClientID = '<yourAadAppClientID>'
$rgname = '<yourResourceGroup>'
Set-AzureRmKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -PermissionsToKeys 'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $rgname

```

NOTE

Azure Disk Encryption requires you to configure the following access policies to your Azure AD client application: *WrapKey* and *Set* permissions.

Terminology

To understand some of the common terms used by this technology, use the following terminology table:

TERMINOLOGY	DEFINITION
Azure AD	Azure AD is Azure Active Directory . An Azure AD account is a prerequisite for authenticating, storing, and retrieving secrets from a key vault.
Azure Key Vault	Key Vault is a cryptographic, key management service that's based on Federal Information Processing Standards (FIPS)-validated hardware security modules, which help safeguard your cryptographic keys and sensitive secrets. For more information, see Key Vault documentation.
ARM	Azure Resource Manager
BitLocker	BitLocker is an industry-recognized Windows volume encryption technology that's used to enable disk encryption on Windows IaaS VMs.
BEK	BitLocker encryption keys are used to encrypt the OS boot volume and data volumes. The BitLocker keys are safeguarded in a key vault as secrets.
CLI	See Azure command-line interface .
DM-Crypt	DM-Crypt is the Linux-based, transparent disk-encryption subsystem that's used to enable disk encryption on Linux IaaS VMs.
KEK	Key encryption key is the asymmetric key (RSA 2048) that you can use to protect or wrap the secret. You can provide a hardware security modules (HSM)-protected key or software-protected key. For more details, see Azure Key Vault documentation.
PS cmdlets	See Azure PowerShell cmdlets .

Set up and configure your key vault for Azure Disk Encryption

Azure Disk Encryption helps safeguard the disk-encryption keys and secrets in your key vault. To set up your key vault for Azure Disk Encryption, complete the steps in each of the following sections.

Create a key vault

To create a key vault, use one of the following options:

- ["101-Key-Vault-Create" Resource Manager template](#)
- [Azure PowerShell key vault cmdlets](#)
- [Azure Resource Manager](#)
- How to [Secure your key vault](#)

NOTE

If you have already set up a key vault for your subscription, skip to the next section.

Everything

Filter

Key Vault

Results

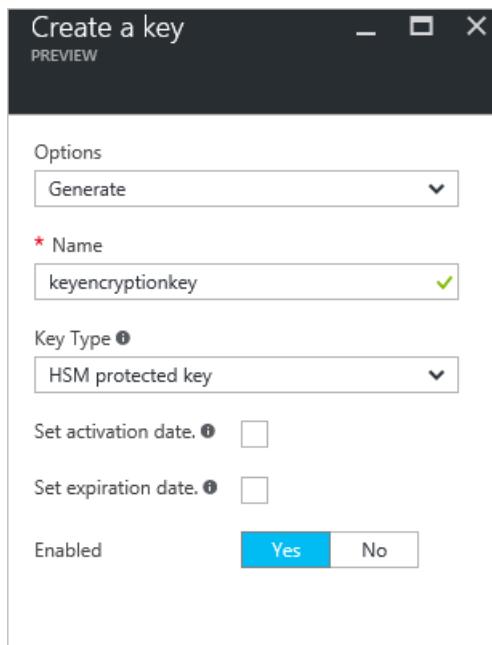
NAME	PUBLISHER	CATEGORY
 Key Vault (preview)	Microsoft	Management

Set up a key encryption key (optional)

If you want to use a KEK for an additional layer of security for the BitLocker encryption keys, add a KEK to your key vault. Use the [Add-AzureKeyVaultKey](#) cmdlet to create a key encryption key in the key vault. You can also import a KEK from your on-premises key management HSM. For more details, see [Key Vault Documentation](#).

```
Add-AzureKeyVaultKey [-VaultName] <string> [-Name] <string> -Destination <string> {HSM | Software}
```

You can add the KEK by going to Azure Resource Manager or by using your key vault interface.



Set key vault permissions

The Azure platform needs access to the encryption keys or secrets in your key vault to make them available to the VM for booting and decrypting the volumes. To grant permissions to the Azure platform, set the **EnabledForDiskEncryption** property in the key vault by using the key vault PowerShell cmdlet:

```
Set-AzureRmKeyVaultAccessPolicy -VaultName <yourVaultName> -ResourceGroupName <yourResourceGroup> -EnabledForDiskEncryption
```

You can also set the **EnabledForDiskEncryption** property by visiting the [Azure Resource Explorer](#).

As mentioned earlier, you must set the **EnabledForDiskEncryption** property on your key vault. Otherwise, the deployment will fail.

You can set up access policies for your Azure AD application from the key vault interface, as shown here:

Add new permissions - □ X

Add a new access policy - PREVIEW

* Select principal
vmencrypt >

Configure from template (optional)

Key permissions
1 selected >

Secret permissions
1 selected >

Authorized application ⓘ
None selected

Key permissions

All Key Operations

All

Key Management Operations

Get

List

Update

Create

Import

Delete

Backup

Restore

Cryptographic Operations

Decrypt

Encrypt

UnwrapKey

WrapKey

Verify

Sign

Add new permissions - □ X

Add a new access policy - PREVIEW

* Select principal
vmencrypt >

Configure from template (optional)

Key permissions
1 selected >

Secret permissions
1 selected >

Authorized application ⓘ
None selected

Secret permissions

All Secret Operations

All

Secret Management Operations

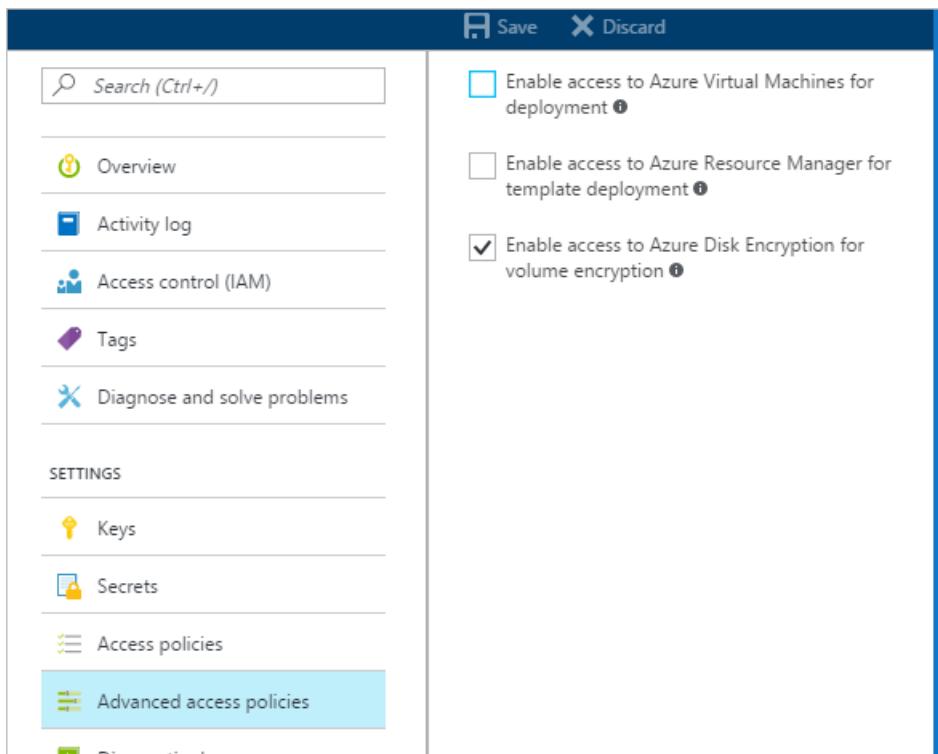
Get

List

Set

Delete

On the **Advanced access policies** tab, make sure that your key vault is enabled for Azure Disk Encryption:



Disk-encryption deployment scenarios and user experiences

You can enable many disk-encryption scenarios, and the steps may vary according to the scenario. The following sections cover the scenarios in greater detail.

Enable encryption on new IaaS VMs that are created from the Marketplace

You can enable disk encryption on new IaaS Windows VM from the Marketplace in Azure by using the [Resource Manager template](#).

1. On the Azure quick-start template, click **Deploy to Azure**, enter the encryption configuration on the **Parameters** blade, and then click **OK**.
2. Select the subscription, resource group, resource group location, legal terms, and agreement, and then click **Create** to enable encryption on a new IaaS VM.

NOTE

This template creates a new encrypted Windows VM that uses the Windows Server 2012 gallery image.

You can enable disk encryption on a new IaaS RedHat Linux 7.2 VM with a 200-GB RAID-0 array by using this [Resource Manager template](#). After you deploy the template, verify the VM encryption status by using the `Get-AzureRmVmDiskEncryptionStatus` cmdlet, as described in [Encrypting OS drive on a running Linux VM](#). When the machine returns a status of *VMRestartPending*, restart the VM.

The following table lists the Resource Manager template parameters for new VMs from the Marketplace scenario using Azure AD client ID:

PARAMETER	DESCRIPTION
adminUserName	Admin user name for the virtual machine.
adminPassword	Admin user password for the virtual machine.

PARAMETER	DESCRIPTION
newStorageAccountName	Name of the storage account to store OS and data VHDS.
vmSize	Size of the VM. Currently, only Standard A, D, and G series are supported.
virtualNetworkName	Name of the VNet that the VM NIC should belong to.
subnetName	Name of the subnet in the VNet that the VM NIC should belong to.
AADClientID	Client ID of the Azure AD application that has permissions to write secrets to your key vault.
AADClientSecret	Client secret of the Azure AD application that has permissions to write secrets to your key vault.
keyVaultURL	URL of the key vault that the BitLocker key should be uploaded to. You can get it by using the cmdlet <div style="border: 1px solid black; padding: 5px;"> <pre>(Get-AzureRmKeyVault -VaultName -ResourceGroupName).VaultURI</pre> </div> .
keyEncryptionKeyURL	URL of the key encryption key that's used to encrypt the generated BitLocker key (optional).
keyVaultResourceGroup	Resource group of the key vault.
vmName	Name of the VM that the encryption operation is to be performed on.

NOTE

KeyEncryptionKeyURL is an optional parameter. You can bring your own KEK to further safeguard the data encryption key (Passphrase secret) in your key vault.

Enable encryption on new IaaS VMs that are created from customer-encrypted VHD and encryption keys

In this scenario, you can enable encrypting by using the Resource Manager template, PowerShell cmdlets, or CLI commands. The following sections explain in greater detail the Resource Manager template and CLI commands.

Follow the instructions from one of these sections for preparing pre-encrypted images that can be used in Azure. After the image is created, you can use the steps in the next section to create an encrypted Azure VM.

- [Prepare a pre-encrypted Windows VHD](#)
- [Prepare a pre-encrypted Linux VHD](#)

Using the Resource Manager template

You can enable disk encryption on your encrypted VHD by using the [Resource Manager template](#).

1. On the Azure quick-start template, click **Deploy to Azure**, enter the encryption configuration on the **Parameters** blade, and then click **OK**.
2. Select the subscription, resource group, resource group location, legal terms, and agreement, and then click **Create** to enable encryption on the new IaaS VM.

The following table lists the Resource Manager template parameters for your encrypted VHD:

PARAMETER	DESCRIPTION
newStorageAccountName	Name of the storage account to store the encrypted OS VHD. This storage account should already have been created in the same resource group and same location as the VM.
osVhdUri	URI of the OS VHD from the storage account.
osType	OS product type (Windows/Linux).
virtualNetworkName	Name of the VNet that the VM NIC should belong to. The name should already have been created in the same resource group and same location as the VM.
subnetName	Name of the subnet on the VNet that the VM NIC should belong to.
vmSize	Size of the VM. Currently, only Standard A, D, and G series are supported.
keyVaultResourceID	The ResourceID that identifies the key vault resource in Azure Resource Manager. You can get it by using the PowerShell cmdlet <pre>(Get-AzureRmKeyVault -VaultName <yourKeyVaultName> -ResourceGroupName <yourResourceGroupName>).ResourceId</pre>
keyVaultSecretUrl	URL of the disk-encryption key that's set up in the key vault.
keyVaultKekUrl	URL of the key encryption key for encrypting the generated disk-encryption key.
vmName	Name of the IaaS VM.

Using PowerShell cmdlets

You can enable disk encryption on your encrypted VHD by using the PowerShell cmdlet [Set-AzureRmVMOSDisk](#).

Using CLI commands

To enable disk encryption for this scenario by using CLI commands, do the following:

1. Set access policies in your key vault:

- Set the **EnabledForDiskEncryption** flag:

```
azure keyvault set-policy --vault-name <keyVaultName> --enabled-for-disk-encryption true
```

- Set permissions to Azure AD application to write secrets to your key vault:

```
azure keyvault set-policy --vault-name <keyVaultName> --spn <aadClientID> --perms-to-keys '['"wrapKey"]' --perms-to-secrets '['"set"]'
```

2. To enable encryption on an existing or running VM, type:

```
azure vm enable-disk-encryption --resource-group <resourceGroupName> --name <vmName> --aad-client-id <aadClientId> --aad-client-secret <aadClientSecret> --disk-encryption-key-vault-url <keyVaultURL> --disk-encryption-key-vault-id <keyVaultResourceId> --volume-type [All|OS|Data]
```

3. Get encryption status:

```
azure vm show-disk-encryption-status --resource-group <resourceGroupName> --name <vmName> --json
```

4. To enable encryption on a new VM from your encrypted VHD, use the following parameters with the `azure vm create` command:

```
* disk-encryption-key-vault-id <disk-encryption-key-vault-id>
* disk-encryption-key-url <disk-encryption-key-url>
* key-encryption-key-vault-id <key-encryption-key-vault-id>
* key-encryption-key-url <key-encryption-key-url>
```

Enable encryption on existing or running IaaS Windows VM in Azure

In this scenario, you can enable encrypting by using the Resource Manager template, PowerShell cmdlets, or CLI commands. The following sections explain in greater detail how to enable it by using the Resource Manager template and CLI commands.

Using the Resource Manager template

You can enable disk encryption on existing or running IaaS Windows VMs in Azure by using the [Resource Manager template](#).

1. On the Azure quick-start template, click **Deploy to Azure**, enter the encryption configuration on the **Parameters** blade, and then click **OK**.
2. Select the subscription, resource group, resource group location, legal terms, and agreement, and then click **Create** to enable encryption on the existing or running IaaS VM.

The following table lists the Resource Manager template parameters for existing or running VMs that use an Azure AD client ID:

PARAMETER	DESCRIPTION
AADClientID	Client ID of the Azure AD application that has permissions to write secrets to the key vault.
AADClientSecret	Client secret of the Azure AD application that has permissions to write secrets to the key vault.
keyVaultName	Name of the key vault that the BitLocker key should be uploaded to. You can get it by using the cmdlet <pre>(Get-AzureRmKeyVault -ResourceGroupName <yourResourceGroupName>).Vaultname</pre>
keyEncryptionKeyURL	URL of the key encryption key that's used to encrypt the generated BitLocker key. This parameter is optional if you select nokek in the UseExistingKek drop-down list. If you select kek in the UseExistingKek drop-down list, you must enter the keyEncryptionKeyURL value.
volumeType	Type of volume that the encryption operation is performed on. Valid values are <i>OS</i> , <i>Data</i> , and <i>All</i> .
sequenceVersion	Sequence version of the BitLocker operation. Increment this version number every time a disk-encryption operation is performed on the same VM.
vmName	Name of the VM that the encryption operation is to be performed on.

NOTE

KeyEncryptionKeyURL is an optional parameter. You can bring your own KEK to further safeguard the data encryption key (BitLocker encryption secret) in the key vault.

Using PowerShell cmdlets

For information about enabling encryption with Azure Disk Encryption by using PowerShell cmdlets, see the blog posts [Explore Azure Disk Encryption with Azure PowerShell - Part 1](#) and [Explore Azure Disk Encryption with Azure PowerShell - Part 2](#).

Using CLI commands

To enable encryption on existing or running IaaS Windows VM in Azure using CLI commands, do the following:

1. To set access policies in the key vault:

- Set the **EnabledForDiskEncryption** flag:

```
azure keyvault set-policy --vault-name <keyVaultName> --enabled-for-disk-encryption true
```

- Set permissions to Azure AD application to write secrets to your key vault:

```
azure keyvault set-policy --vault-name <keyVaultName> --spn <aadClientID> --perms-to-keys  
'[ "wrapKey" ]' --perms-to-secrets '[ "set" ]'
```

2. To enable encryption on an existing or running VM:

```
azure vm enable-disk-encryption --resource-group <resourceGroupName> --name <vmName> --aad-client-id  
<aadClientId> --aad-client-secret <aadClientSecret> --disk-encryption-key-vault-url <keyVaultURL> --  
disk-encryption-key-vault-id <keyVaultResourceId> --volume-type [All|OS|Data]
```

3. To get encryption status:

```
azure vm show-disk-encryption-status --resource-group <resourceGroupName> --name <vmName> --json
```

4. To enable encryption on a new VM from your encrypted VHD, use the following parameters with the

```
azure vm create
```

 command:

```
* disk-encryption-key-vault-id <disk-encryption-key-vault-id>  
* disk-encryption-key-url <disk-encryption-key-url>  
* key-encryption-key-vault-id <key-encryption-key-vault-id>  
* key-encryption-key-url <key-encryption-key-url>
```

Enable encryption on an existing or running IaaS Linux VM in Azure

You can enable disk encryption on an existing or running IaaS Linux VM in Azure by using the [Resource Manager template](#).

1. Click **Deploy to Azure** on the Azure quick-start template, enter the encryption configuration on the **Parameters** blade, and then click **OK**.
2. Select the subscription, resource group, resource group location, legal terms, and agreement, and then click **Create** to enable encryption on the existing or running IaaS VM.

The following table lists Resource Manager template parameters for existing or running VMs that use an Azure AD client ID:

PARAMETER	DESCRIPTION
AADClientID	Client ID of the Azure AD application that has permissions to write secrets to the key vault.

PARAMETER	DESCRIPTION
AADClientSecret	Client secret of the Azure AD application that has permissions to write secrets to your key vault.
keyVaultName	Name of the key vault that the BitLocker key should be uploaded to. You can get it by using the cmdlet <pre>(Get-AzureRmKeyVault -ResourceGroupName <yourResourceGroupName>).Vaultname</pre>
keyEncryptionKeyURL	URL of the key encryption key that's used to encrypt the generated BitLocker key. This parameter is optional if you select nokek in the UseExistingKek drop-down list. If you select kek in the UseExistingKek drop-down list, you must enter the <i>keyEncryptionKeyURL</i> value.
volumeType	Type of volume that the encryption operation is performed on. Valid supported values are <i>OS</i> or <i>All</i> (for RHEL 7.2, CentOS 7.2, and Ubuntu 16.04), and <i>Data</i> (for all other distributions).
sequenceVersion	Sequence version of the BitLocker operation. Increment this version number every time a disk-encryption operation is performed on the same VM.
vmName	Name of the VM that the encryption operation is to be performed on.
passPhrase	Type a strong passphrase as the data encryption key.

NOTE

KeyEncryptionKeyURL is an optional parameter. You can bring your own KEK to further safeguard the data encryption key (passphrase secret) in your key vault.

CLI commands

You can enable disk encryption on your encrypted VHD by installing and using the [CLI command](#). To enable encryption on existing or running IaaS Linux VMs in Azure by using CLI commands, do the following:

1. Set access policies in the key vault:

- Set the **EnabledForDiskEncryption** flag:

```
azure keyvault set-policy --vault-name <keyVaultName> --enabled-for-disk-encryption true
```

- Set permissions to Azure AD application to write secrets to your key vault:

```
azure keyvault set-policy --vault-name <keyVaultName> --spn <aadClientID> --perms-to-keys '['"wrapKey"]' --perms-to-secrets '['"set"]'
```

2. To enable encryption on an existing or running VM:

```
azure vm enable-disk-encryption --resource-group <resourceGroupName> --name <vmName> --aad-client-id <aadClientId> --aad-client-secret <aadClientSecret> --disk-encryption-key-vault-url <keyVaultURL> --disk-encryption-key-vault-id <keyVaultResourceId> --volume-type [All|OS|Data]
```

3. Get encryption status:

```
azure vm show-disk-encryption-status --resource-group <resourceGroupName> --name <vmName> --json
```

4. To enable encryption on a new VM from your encrypted VHD, use the following parameters with the

```
azure vm create command:
```

```
* disk-encryption-key-vault-id <disk-encryption-key-vault-id>
* disk-encryption-key-url <disk-encryption-key-url>
* key-encryption-key-vault-id <key-encryption-key-vault-id>
* key-encryption-key-url <key-encryption-key-url>
```

Get the encryption status of an encrypted IaaS VM

You can get the encryption status by using Azure Resource Manager, [PowerShell cmdlets](#), or CLI commands. The following sections explain how to use the Azure classic portal and CLI commands to get the encryption status.

Get the encryption status of an encrypted Windows VM by using Azure Resource Manager

You can get the encryption status of the IaaS VM from Azure Resource Manager by doing the following:

1. Sign in to the [Azure classic portal](#), and then click **Virtual machines** in the left pane to see a summary view of the virtual machines in your subscription. You can filter the virtual machines view by selecting the subscription name in the **Subscription** drop-down list.
2. At the top of the **Virtual machines** page, click **Columns**.
3. On the **Choose column** blade, select **Disk Encryption**, and then click **Update**. You should see the disk-encryption column showing the encryption state *Enabled* or *Not Enabled* for each VM, as shown in the following figure:

The screenshot shows the Azure classic portal interface. The left sidebar has 'Virtual machines' selected. The main area is titled 'Virtual machines' and lists four VMs: ADEDemoCAT, ADEPreDemoCAT, at-east, and at-prevm10. A column header 'DISK ENCRYPTION' is highlighted with a black border. The data for each VM is as follows:

NAME	STATUS	LOCATION	SIZE	DISK ENCRYPTION
ADEDemoCAT	Running	Australia East	Standard_D1	Enabled
ADEPreDemoCAT	Running	Australia East	Standard_D1	Enabled
at-east	Running	East US	Standard_A1	Enabled
at-prevm10	Running	Australia East	Standard_D2	Enabled

Get the encryption status of an encrypted (Windows/Linux) IaaS VM by using the disk-encryption PowerShell cmdlet

You can get the encryption status of the IaaS VM from the disk-encryption PowerShell cmdlet

```
Get-AzureRmVMDiskEncryptionStatus
```

. To get the encryption settings for your VM, enter the following:

```
C:\> Get-AzureRmVmDiskEncryptionStatus -ResourceGroupName $ResourceGroupName -VMName $VMName -ExtensionName $ExtensionName

OsVolumeEncrypted      : NotEncrypted
DataVolumesEncrypted   : Encrypted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : https://rheltest1keyvault.vault.azure.net/secrets/bdb6bf1-5431-4c28-af46-b18d0025ef2a/abebacb83d864a5fa729508315020f8a
```

You can inspect the output of `Get-AzureRmVMDiskEncryptionStatus` for encryption key URLs.

```
C:\> $status = Get-AzureRmVmDiskEncryptionStatus -ResourceGroupName $ResourceGroupName -VMName
e $VMName -ExtensionName $ExtensionName
C:\> $status.OsVolumeEncryptionSettings

DiskEncryptionKey                               KeyEncryptionKey
Enabled
-----
-----
Microsoft.Azure.Management.Compute.Models.KeyVaultSecretReference
Microsoft.Azure.Management.Compute.Models.KeyVaultKeyReference   True

C:\> $status.OsVolumeEncryptionSettings.DiskEncryptionKey.SecretUrl
https://rheltest1keyvault.vault.azure.net/secrets/bdb6bfb1-5431-4c28-af46-
b18d0025ef2a/abebacb83d864a5fa729508315020f8a
C:\> $status.OsVolumeEncryptionSettings.DiskEncryptionKey

SecretUrl
SourceVault
-----
-----
https://rheltest1keyvault.vault.azure.net/secrets/bdb6bfb1-5431-4c28-af46-
b18d0025ef2a/abebacb83d864a5fa729508315020f8a Microsoft.Azure.Management....
```

The OSVolumeEncrypted and DataVolumesEncrypted settings values are set to *Encrypted*, which shows that both volumes are encrypted using Azure Disk Encryption. For information about enabling encryption with Azure Disk Encryption by using PowerShell cmdlets, see the blog posts [Explore Azure Disk Encryption with Azure PowerShell - Part 1](#) and [Explore Azure Disk Encryption with Azure PowerShell - Part 2](#).

NOTE

On Linux VMs, it takes three to four minutes for the `Get-AzureRmVmDiskEncryptionStatus` cmdlet to report the encryption status.

Get the encryption status of the IaaS VM from the disk-encryption CLI command

You can get the encryption status of the IaaS VM by using the disk-encryption CLI command

`azure vm show-disk-encryption-status`. To get the encryption settings for your VM, enter your Azure CLI session:

```
azure vm show-disk-encryption-status --resource-group <yourResourceGroupName> --name <yourVMName> --json
```

Disable encryption on running Windows IaaS VM

You can disable encryption on a running Windows or Linux IaaS VM via the Azure Disk Encryption Resource Manager template or PowerShell cmdlets and specify the decryption configuration.

Windows VM

The disable-encryption step disables encryption of the OS, the data volume, or both on the running Windows IaaS VM. You cannot disable the OS volume and leave the data volume encrypted. When the disable-encryption step is performed, the Azure classic deployment model updates the VM service model, and the Windows IaaS VM is marked decrypted. The contents of the VM are no longer encrypted at rest. The decryption does not delete your key vault and the encryption key material (BitLocker encryption keys for Windows and Passphrase for Linux).

Linux VM

The disable-encryption step disables encryption of the data volume on the running Linux IaaS VM. This step only works if the OS disk is not encrypted.

NOTE

Disabling encryption on the OS disk is not allowed on Linux VMs.

Disable encryption on an existing or running IaaS VM

You can disable disk encryption on running Windows IaaS VMs by using the [Resource Manager template](#).

1. On the Azure quick-start template, click **Deploy to Azure**, enter the decryption configuration on the **Parameters** blade, and then click **OK**.
2. Select the subscription, resource group, resource group location, legal terms, and agreement, and then click **Create** to enable encryption on a new IaaS VM.

For Linux VMs, you can disable encryption by using the [Disable encryption on a running Linux VM](#) template.

The following table lists Resource Manager template parameters for disabling encryption on a running IaaS VM:

PARAMETER	DESCRIPTION
vmName	Name of the VM that the encryption operation is to be performed on.
volumeType	Type of volume that a decryption operation is performed on. Valid values are <i>OS</i> , <i>Data</i> , and <i>All</i> . You cannot disable encryption on running Windows IaaS VM OS/boot volume without disabling encryption on the <i>Data</i> volume. Also note that disabling encryption on the OS disk is not allowed on Linux VMs.
sequenceVersion	Sequence version of the BitLocker operation. Increment this version number every time a disk decryption operation is performed on the same VM.

Disable encryption on an existing or running IaaS VM

To disable encryption on an existing or running IaaS VM by using the PowerShell cmdlet, see

[Disable-AzureRmVMDiskEncryption](#). This cmdlet supports both Windows and Linux VMs. To disable encryption, it installs an extension on the virtual machine. If the *Name* parameter is not specified, an extension with the default name *AzureDiskEncryption* for Windows VMs is created.

On Linux VMs, the *AzureDiskEncryptionForLinux* extension is used.

NOTE

This cmdlet reboots the virtual machine.

Enable encryption on pre-encrypted IaaS VM with Azure Managed Disk

Use the Azure Managed Disk ARM template to create a encrypted VM from a pre-encrypted VHD using the ARM template located at

[Create a new encrypted managed disk from a pre-encrypted VHD/storage blob](#)

Enable encryption on a new Linux IaaS VM with Azure Managed Disk

Use the Azure Managed Disk ARM template to create a new encrypted Linux IaaS VM using the ARM template located at

[Deployment of RHEL 7.2 with full disk encryption](#)

Enable encryption on a new Windows IaaS VM with Azure Managed Disk

Use the Azure Managed Disk ARM template to create a new encrypted Linux IaaS VM using the ARM template located at

[Create a new encrypted Windows IaaS Managed Disk VM from gallery image](#)

NOTE

It is mandatory to snapshot and/or backup a managed disk based VM instance outside of and prior to enabling Azure Disk Encryption. A snapshot of the managed disk can be taken from the portal, or Azure Backup can be used. Backups ensure that a recovery option is possible in the case of any unexpected failure during encryption. Once a backup is made, the Set-AzureRmVMDiskEncryptionExtension cmdlet can be used to encrypt managed disks by specifying the -skipVmBackup parameter. This command will fail against managed disk based VM's until a backup has been made and this parameter has been specified.

Update encryption settings of an existing encrypted non-premium VM

Use the existing Azure disk encryption supported interfaces for running VM [PS cmdlets, CLI or ARM templates] to update the encryption settings like AAD client ID/secret, Key encryption key [KEK], BitLocker encryption key for Windows VM or Passphrase for Linux VM etc. The update encryption setting is supported only for VMs backed by non-premium storage. It is NOT supported for VMs backed by premium storage.

Appendix

Connect to your subscription

Before you proceed, review the *Prerequisites* section in this article. After you ensure that all prerequisites have been met, connect to your subscription by doing the following:

1. Start an Azure PowerShell session, and sign in to your Azure account with the following command:

```
Login-AzureRmAccount
```

2. If you have multiple subscriptions and want to specify one to use, type the following to see the subscriptions for your account:

```
Get-AzureRmSubscription
```

3. To specify the subscription you want to use, type:

```
Select-AzureRmSubscription -SubscriptionName <Yoursubscriptionname>
```

4. To verify that the subscription configured is correct, type:

```
Get-AzureRmSubscription
```

5. To confirm the Azure Disk Encryption cmdlets are installed, type:

```
Get-command *diskencryption*
```

6. The following output confirms the Azure Disk Encryption PowerShell installation:

```
PS C:\Windows\System32\WindowsPowerShell\v1.0> get-command *diskencryption*
 CommandType     Name                                               Source
 Cmdlet          Get-AzureRmVMDiskEncryptionStatus                 AzureRM.Compute
 Cmdlet          Disable-AzureRmVMDiskEncryption                  AzureRM.Compute
 Cmdlet          Set-AzureRmVMDiskEncryptionExtension             AzureRM.Compute
```

Prepare a pre-encrypted Windows VHD

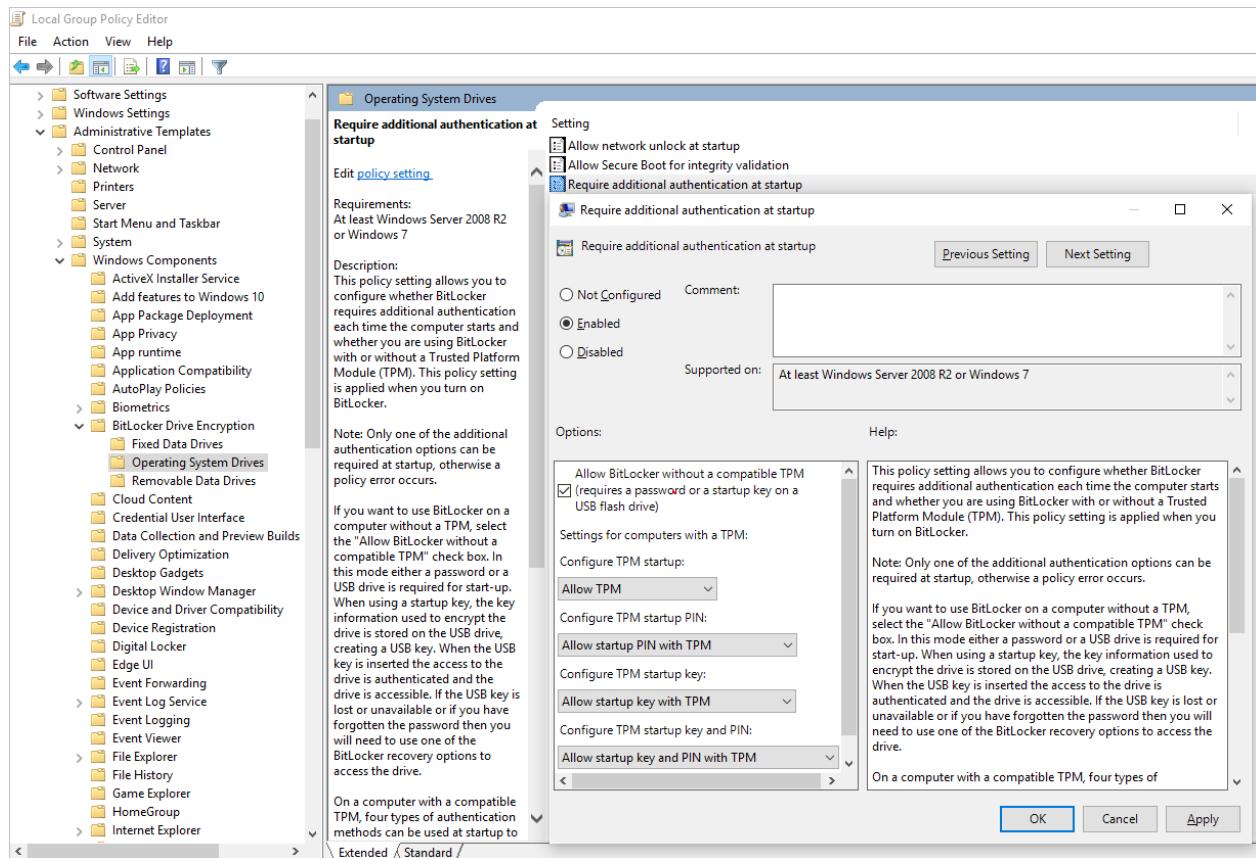
The sections that follow are necessary to prepare a pre-encrypted Windows VHD for deployment as an encrypted

VHD in Azure IaaS. Use the information to prepare and boot a fresh Windows VM (VHD) on Azure Site Recovery or Azure.

Update group policy to allow non-TPM for OS protection

Configure the BitLocker Group Policy setting **BitLocker Drive Encryption**, which you'll find under **Local Computer Policy > Computer Configuration > Administrative Templates > Windows Components**.

Change this setting to **Operating System Drives > Require additional authentication at startup > Allow BitLocker without a compatible TPM**, as shown in the following figure:



Install BitLocker feature components

For Windows Server 2012 and later, use the following command:

```
dism /online /Enable-Feature /all /FeatureName:BitLocker /quiet /norestart
```

For Windows Server 2008 R2, use the following command:

```
ServerManagerCmd -install BitLockers
```

Prepare the OS volume for BitLocker by using `bdehdcfg`

To compress the OS partition and prepare the machine for BitLocker, execute the following command:

```
bdehdcfg -target c: shrink -quiet
```

Protect the OS volume by using BitLocker

Use the `manage-bde` command to enable encryption on the boot volume using an external key protector. Also place the external key (.bek file) on the external drive or volume. Encryption is enabled on the system/boot volume after the next reboot.

```
manage-bde -on %systemdrive% -sk [ExternalDriveOrVolume]
reboot
```

NOTE

Prepare the VM with a separate data/resource VHD for getting the external key by using BitLocker.

Encrypting an OS drive on a running Linux VM

Encryption of an OS drive on a running Linux VM is supported on the following distributions:

- RHEL 7.2
- CentOS 7.2
- Ubuntu 16.04

Prerequisites for OS disk encryption

- The VM must be created from the Marketplace image in Azure Resource Manager.
- Azure VM with at least 4 GB of RAM (recommended size is 7 GB).
- (For RHEL and CentOS) Disable SELinux. To disable SELinux, see "4.4.2. Disabling SELinux" in the [SELinux User's and Administrator's Guide](#) on the VM.
- After you disable SELinux, reboot the VM at least once.

Steps

1. Create a VM by using one of the distributions specified previously.

For CentOS 7.2, OS disk encryption is supported via a special image. To use this image, specify "7.2n" as the SKU when you create the VM:

```
Set-AzureRmVMSourceImage -VM $VirtualMachine -PublisherName "OpenLogic" -Offer "CentOS" -Skus "7.2n"
-Version "latest"
```

2. Configure the VM according to your needs. If you are going to encrypt all the (OS + data) drives, the data drives need to be specified and mountable from /etc/fstab.

NOTE

Use UUID=... to specify data drives in /etc/fstab instead of specifying the block device name (for example, /dev/sdb1). During encryption, the order of drives changes on the VM. If your VM relies on a specific order of block devices, it will fail to mount them after encryption.

3. Sign out of the SSH sessions.

4. To encrypt the OS, specify volumeType as **All** or **OS** when you [enable encryption](#).

NOTE

All user-space processes that are not running as `systemd` services should be killed with a `SIGKILL`. Reboot the VM. When you enable OS disk encryption on a running VM, plan on VM downtime.

5. Periodically monitor the progress of encryption by using the instructions in the [next section](#).

6. After Get-AzureRmVmDiskEncryptionStatus shows "VMRestartPending," restart your VM either by signing in to it or by using the portal, PowerShell, or CLI.

```
C:\> Get-AzureRmVmDiskEncryptionStatus -ResourceGroupName $ResourceGroupName -VMName $VMName
-ExtensionName $ExtensionName

OsVolumeEncrypted      : VMRestartPending
DataVolumesEncrypted    : NotMounted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : OS disk successfully encrypted, reboot the VM
```

Before you reboot, we recommend that you save [boot diagnostics](#) of the VM.

Monitoring OS encryption progress

You can monitor OS encryption progress in three ways:

- Use the `Get-AzureRmVmDiskEncryptionStatus` cmdlet and inspect the ProgressMessage field:

```
OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted    : NotMounted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : OS disk encryption started
```

After the VM reaches "OS disk encryption started," it takes about 40 to 50 minutes on a Premium-storage backed VM.

Because of [issue #388](#) in WALinuxAgent, `OsVolumeEncrypted` and `DataVolumesEncrypted` show up as `Unknown` in some distributions. With WALinuxAgent version 2.1.5 and later, this issue is fixed automatically. If you see `Unknown` in the output, you can verify disk-encryption status by using the Azure Resource Explorer.

Go to [Azure Resource Explorer](#), and then expand this hierarchy in the selection panel on left:

```
-- subscriptions
  |-- [Your subscription]
    |-- resourceGroups
      |-- [Your resource group]
        |-- providers
          |-- Microsoft.Compute
            |-- virtualMachines
              |-- [Your virtual machine]
                |-- InstanceView
```

In the InstanceView, scroll down to see the encryption status of your drives.

The screenshot shows the Azure Resource Explorer interface. On the left, there's a navigation tree with the following structure:

- Linux disk encryption
 - + providers
 - resourceGroups
 - + CentOSTest1ResourceGroup
 - CentOSTest2ResourceGroup
 - providers
 - * Show all
 - Microsoft.Compute
 - * Show all
 - virtualMachines
 - CentOSTest2VM
 - InstanceView**

The **InstanceView** node is selected and highlighted in blue. To its right, the main pane displays a JSON log entry for disk provisioning:

```
60-
61  {
62    "code": "Provisioning5State/succeeded",
63    "level": "Info",
64    "displayStatus": "Provisioning succeeded",
65    "time": "2016-09-22T02:19:41.4646766+00:00"
66  }
67  ]
68 ],
69 "extensions": [
70  {
71    "name": "AzureDiskEncryptionForLinux",
72    "type": "Microsoft.Azure.Security.AzureDiskEncryptionForLinux",
73    "typeHandlerVersion": "0.1.0.999190",
74    "substatuses": [
75      {
76        "code": "ComponentStatus/Microsoft.Azure.Security.AzureDiskEncryptionForLinux
77        "level": "Info",
78        "displayStatus": "Provisioning succeeded",
79        "message": "{\"os\": \"NotEncrypted\", \"data\": \"EncryptionInProgress\"}"
80      }
81    ]
82  }
83 ]
```

- Look at [boot diagnostics](#). Messages from the ADE extension should be prefixed with `[AzureDiskEncryption]`.
- Sign in to the VM via SSH, and get the extension log from:

/var/log/azure/Microsoft.Azure.Security.AzureDiskEncryptionForLinux

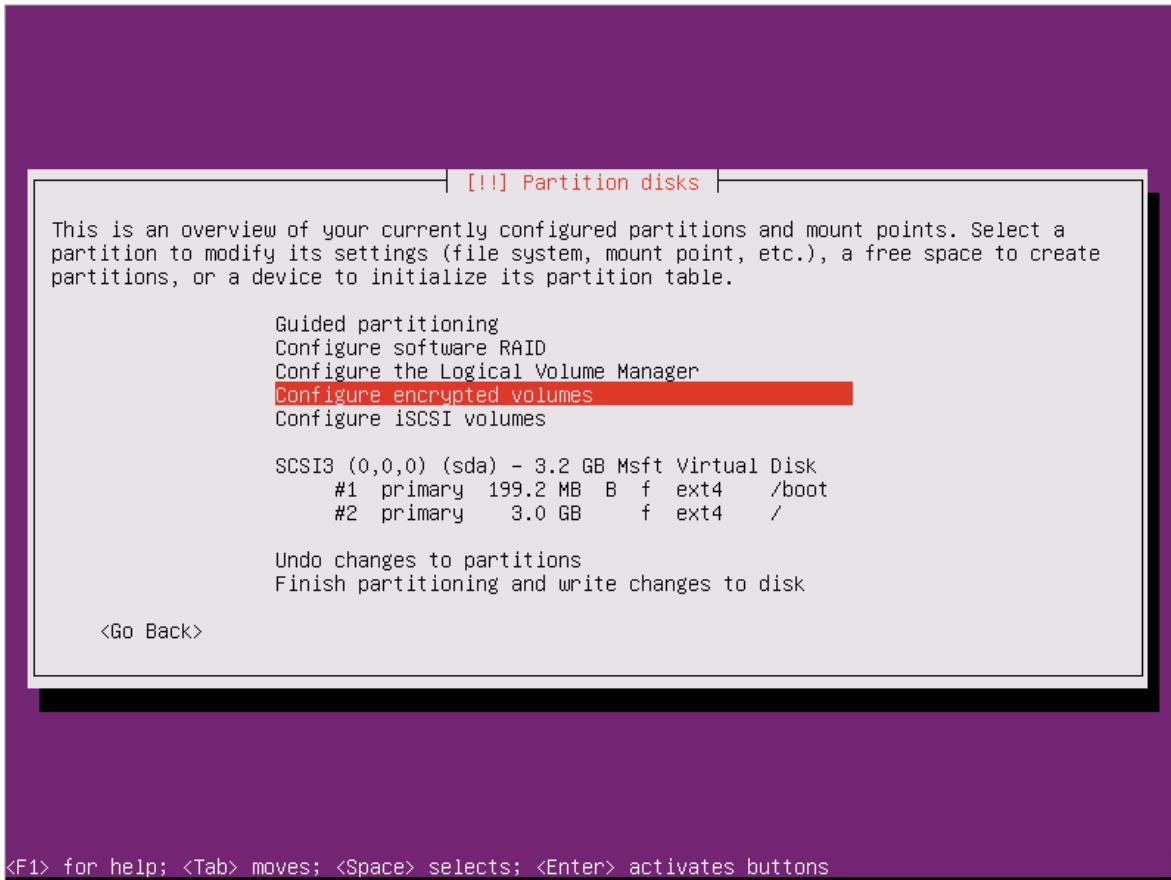
We recommend that you do not sign in to the VM while OS encryption is in progress. Copy the logs only when the other two methods have failed.

Prepare a pre-encrypted Linux VHD

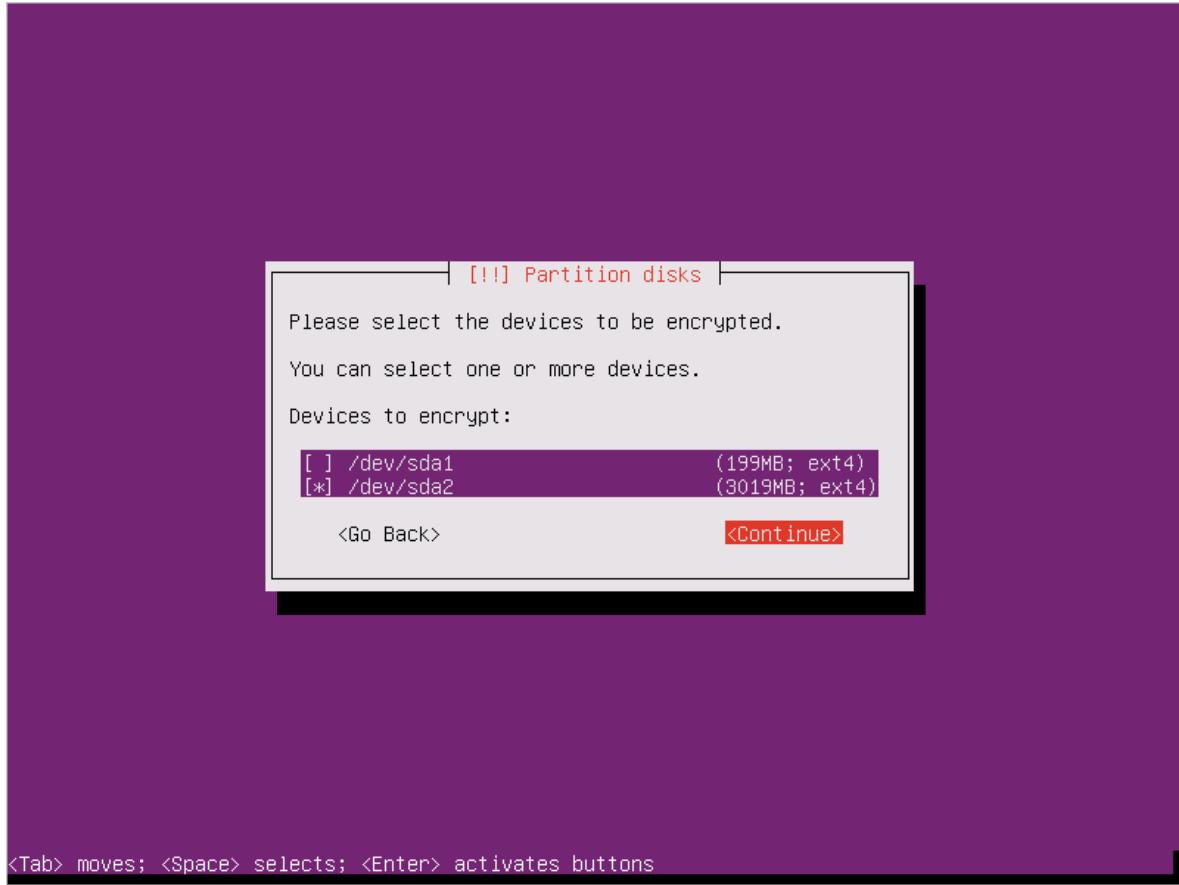
Ubuntu 16

Configure encryption during the distribution installation by doing the following:

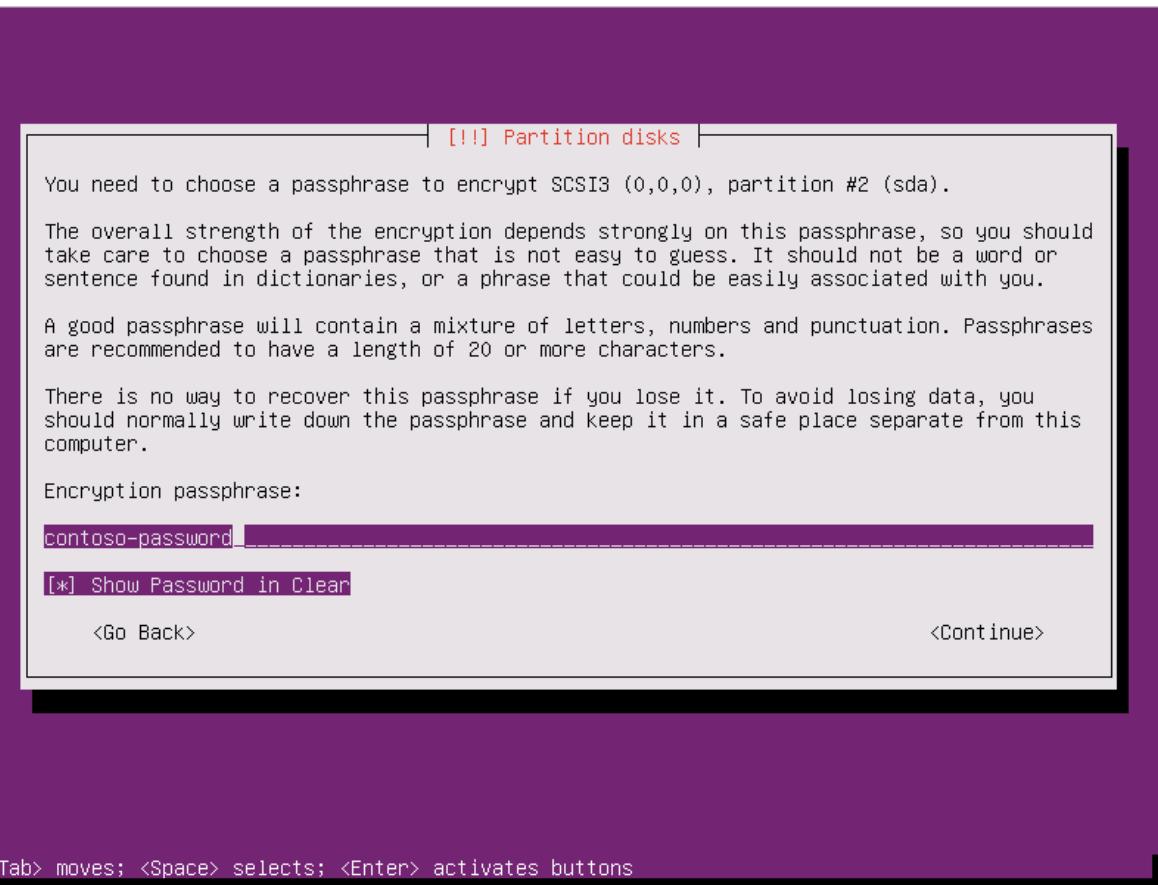
1. Select **Configure encrypted volumes** when you partition the disks.



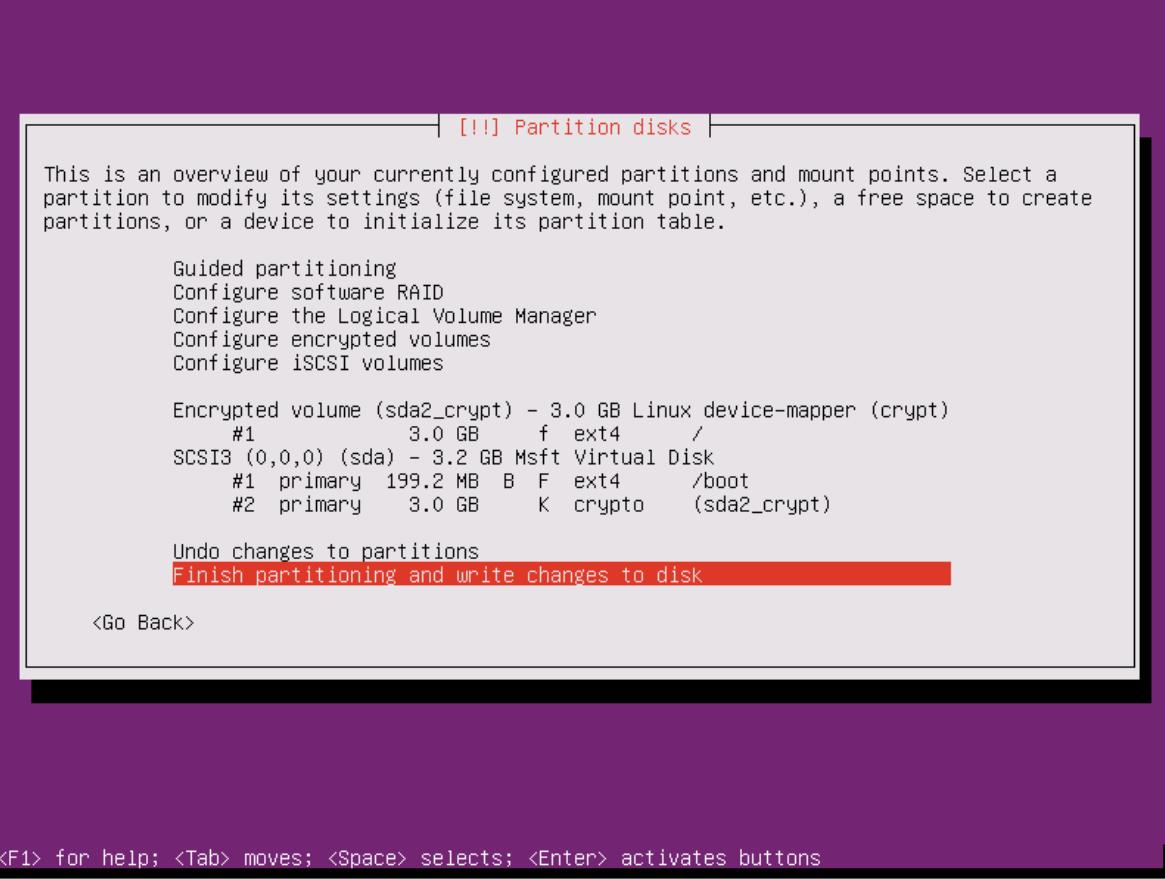
2. Create a separate boot drive, which must not be encrypted. Encrypt your root drive.



3. Provide a passphrase. This is the passphrase that you upload to the key vault.



4. Finish partitioning.



<F1> for help; <Tab> moves; <Space> selects; <Enter> activates buttons

5. When you boot the VM and are asked for a passphrase, use the passphrase you provided in step 3.

```
[ 1.129797] input: Microsoft Umbus HID-compliant Mouse as /devices/0006:045E:0621.0001/input/input4
[ 1.132206] sda: sda1 sda2
[ 1.133217] hid 0006:045E:0621.0001: input: <UNKNOWN> HID v0.01 Mouse [Microsoft Umbus HID-compliant Mouse] on
[ 1.134340] hv_netvsc: hv_netvsc channel opened successfully
[ 1.138418] sd 2:0:0:0: [sda] Attached SCSI disk
[ 1.265049] hv_netvsc umbus_15: Send section size: 6144, Section count:2560
[ 1.266137] hv_netvsc umbus_15: Device MAC 00:15:5d:05:34:01 link state up
[ 1.272596] scsi host3: storvsc_host_t
[ 1.436076] psmouse serio1: trackpoint: failed to get extended button data
Begin: Loading essential drivers ... [ 2.401782] md: linear personality registered for level -1
[ 2.404316] md: multipath personality registered for level -4
[ 2.407122] md: raid0 personality registered for level 0
[ 2.410610] md: raid1 personality registered for level 1
[ 2.480091] raid6: sse2x1 gen() 10995 MB/s
[ 2.548012] raid6: sse2x1 xor() 8467 MB/s
[ 2.616010] raid6: sse2x2 gen() 14312 MB/s
[ 2.684013] raid6: sse2x2 xor() 9555 MB/s
[ 2.752011] raid6: sse2x4 gen() 16205 MB/s
[ 2.820010] raid6: sse2x4 xor() 11594 MB/s
[ 2.888007] raid6: avx2x1 gen() 21995 MB/s
[ 2.956007] raid6: avx2x2 gen() 25959 MB/s
[ 3.024011] raid6: avx2x4 gen() 29505 MB/s
[ 3.024735] raid6: using algorithm avx2x4 gen() 29505 MB/s
[ 3.025038] raid6: using avx2x2 recovery algorithm
[ 3.027102] xor: automatically using best checksumming function:
[ 3.064003] avx : 35013.000 MB/sec
[ 3.065688] async_tx: api initialized (async)
[ 3.074685] md: raid6 personality registered for level 6
[ 3.075435] md: raid5 personality registered for level 5
[ 3.075746] md: raid4 personality registered for level 4
[ 3.079565] md: raid10 personality registered for level 10
done.
Begin: Running /scripts/init-premount ... done.
Begin: Mounting root file system ... Begin: Running /scripts/local-top ... Please unlock disk sda2_crypt: _
```

6. Prepare the VM for uploading into Azure using [these instructions](#). Do not run the last step (deprovisioning the VM) yet.

Configure encryption to work with Azure by doing the following:

1. Create a file under /usr/local/sbin/azure_crypt_key.sh, with the content in the following script. Pay attention to the KeyFileName, because it is the passphrase file name used by Azure.

```
#!/bin/sh
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint
modprobe vfat >/dev/null 2>&1
modprobe ntfs >/dev/null 2>&1
sleep 2
OPENED=0
cd /sys/block
for DEV in sd*; do

    echo "> Trying device: $DEV ..." >&2
    mount -t vfat -r /dev/${DEV}1 $MountPoint >/dev/null ||
    mount -t ntfs -r /dev/${DEV}1 $MountPoint >/dev/null
    if [ -f $MountPoint/$KeyFileName ]; then
        cat $MountPoint/$KeyFileName
        umount $MountPoint 2>/dev/null
        OPENED=1
        break
    fi
    umount $MountPoint 2>/dev/null
done

if [ $OPENED -eq 0 ]; then
    echo "FAILED to find suitable passphrase file ..." >&2
    echo -n "Try to enter your password: " >&2
    read -s -r A </dev/console
    echo -n "$A"
else
    echo "Success loading keyfile!" >&2
fi
```

2. Change the crypt config in /etc/crypttab. It should look like this:

```
xxx_crypt uuid=xxxxxxxxxxxxxxxxxxxxxx none luks,discard,keyscheme=/usr/local/sbin/azure_crypt_key.sh
```

3. If you are editing *azure_crypt_key.sh* in Windows and you copied it to Linux, run

```
dos2unix /usr/local/sbin/azure_crypt_key.sh .
```

4. Add executable permissions to the script:

```
chmod +x /usr/local/sbin/azure_crypt_key.sh
```

5. Edit /etc/initramfs-tools/modules by appending lines: vfat ntfs nls_cp437 nls_utf8 nls_iso8859-1

6. Run `update-initramfs -u -k all` to update the initramfs to make the `keyscheme` take effect.

7. Now you can deprovision the VM.

```

root@ubuntu-preencrypted:~# ls -l /usr/local/sbin/azure_crypt_key.sh
-rwxr-xr-x 1 root root 860 Sep 18 16:57 /usr/local/sbin/azure_crypt_key.sh
root@ubuntu-preencrypted:~# cat /etc/crypttab
sda2_crypt UUID=b0dee704-1f2a-4f02-9a13-289c6c99dbb8 none luks,discard,keyscheme=/usr/local/sbin/azure_crypt_key.sh
root@ubuntu-preencrypted:~# cat /etc/initramfs-tools/modules
# List of modules that you want to include in your initramfs.
# They will be loaded at boot time in the order below.
#
# Syntax: module_name [args ...]
#
# You must run update-initramfs(8) to effect this change.
#
# Examples:
#
# raid1
# sd_mod
# vfat
# ntfs
# nls_cp437
# nls_utf8
# nls_iso8859-1
root@ubuntu-preencrypted:~# update-initramfs -u -k all
update-initramfs: Generating /boot/initrd.img-4.4.0-36-generic
W: plymouth: The plugin label.so is missing, the selected theme might not work as expected.
W: plymouth: You might want to install the plymouth-themes and plymouth-label package to fix this.
W: mdadm: /etc/mdadm/mdadm.conf defines no arrays.
I 6289.9601731 blk_update_request: I/O error, dev fd0, sector 0
update-initramfs: Generating /boot/initrd.img-4.4.0-21-generic
W: plymouth: The plugin label.so is missing, the selected theme might not work as expected.
W: plymouth: You might want to install the plymouth-themes and plymouth-label package to fix this.
W: mdadm: /etc/mdadm/mdadm.conf defines no arrays.
I 6297.5922361 blk_update_request: I/O error, dev fd0, sector 0
root@ubuntu-preencrypted:~# waagent -force -deprovision
WARNING! The waagent service will be stopped.
WARNING! Cached DHCP leases will be deleted.
WARNING! root password will be disabled. You will not be able to login as root.
WARNING! Nameserver configuration in /etc/resolvconf/resolv.conf.d/{tail,original} will be deleted.
2016/09/18 17:06:38.572398 INFO resolvconf is enabled; leaving /etc/resolv.conf intact
2016/09/18 17:06:38.572398 INFO resolvconf is enabled; leaving /etc/resolv.conf intact
root@ubuntu-preencrypted:~# export HISTSIZE=0
root@ubuntu-preencrypted:~# logout

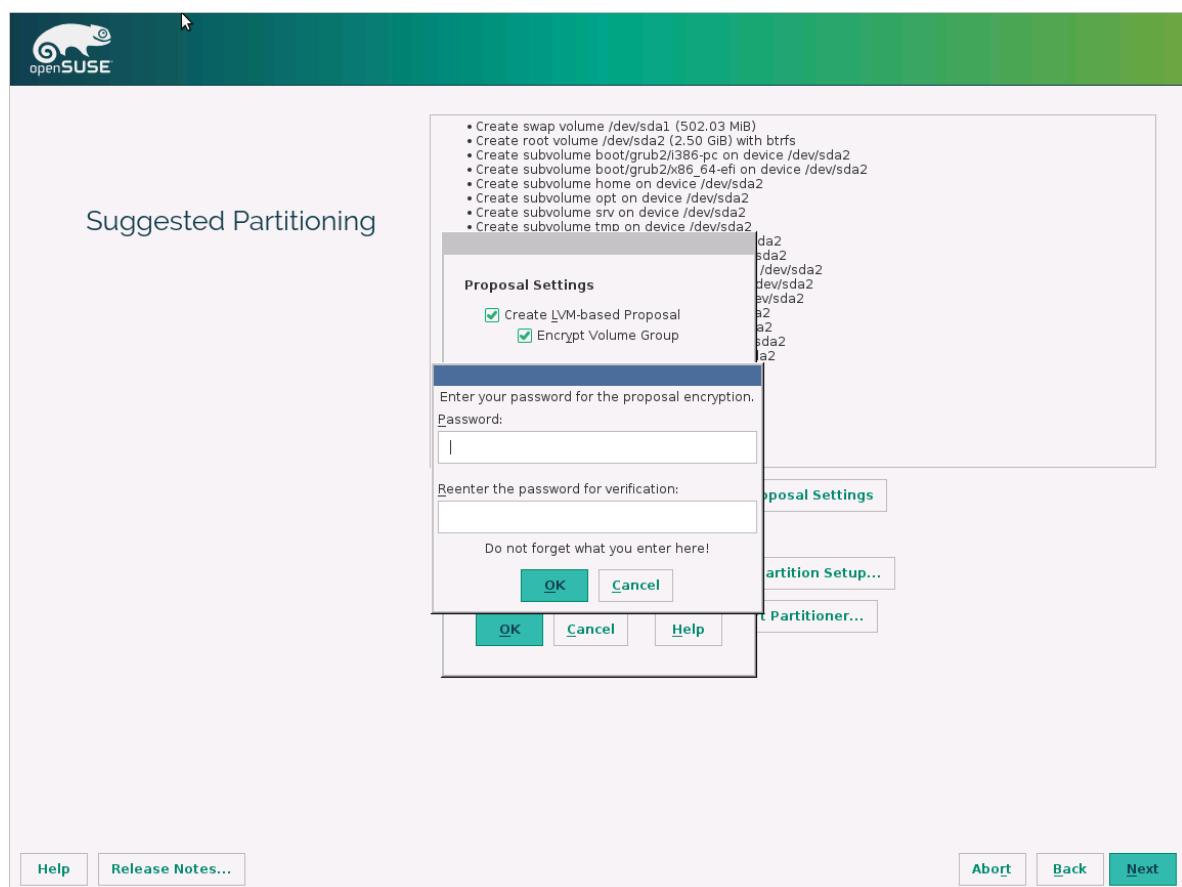
```

8. Continue to the next step and [upload your VHD](#) into Azure.

openSUSE 13.2

To configure encryption during the distribution installation, do the following:

- When you partition the disks, select **Encrypt Volume Group**, and then enter a password. This is the password that you will upload to your key vault.



2. Boot the VM using your password.

```
[ 0.000000] tsc: Fast TSC calibration failed
[ OK ] Found device Virtual_Disk.
[ OK ] Found device Virtual_Disk.
    Starting Cryptography Setup for cr_scsi-14d53465420202020fd10f64360...278fd6327ec-part2...
    Starting Setup Virtual Console...
[ OK ] Started Setup Virtual Console.
    Starting Dispatch Password Requests to Console...
[ OK ] Started Dispatch Password Requests to Console.
Please enter passphrase for disk Virtual_Disk (cr_scsi-14d53465420202020fd10f64360f5f14797052278fd6327ec-part2)! ****
```

3. Prepare the VM for uploading to Azure by following the instructions in [Prepare a SLES or openSUSE virtual machine for Azure](#). Do not run the last step (deprovisioning the VM) yet.

To configure encryption to work with Azure, do the following:

1. Edit the /etc/dracut.conf, and add the following line: `add_drivers+=" vfat ntfs nls_cp437 nls_iso8859-1"`
2. Comment out these lines by the end of the file /usr/lib/dracut/modules.d/90crypt/module-setup.sh:

```
#      inst_multiple -o \
#          $systemdutildir/system-generators/systemd-cryptsetup-generator \
#          $systemdutildir/systemd-cryptsetup \
#          $systemdsystemunitdir/systemd-ask-password-console.path \
#          $systemdsystemunitdir/systemd-ask-password-console.service \
#          $systemdsystemunitdir/cryptsetup.target \
#          $systemdsystemunitdir/sysinit.target.wants/cryptsetup.target \
#          systemd-ask-password systemd-tty-ask-password-agent
#      inst_script "$moddir"/crypt-run-generator.sh /sbin/crypt-run-generator
```

3. Append the following line at the beginning of the file /usr/lib/dracut/modules.d/90crypt/parse-crypt.sh:

```
DRACUT_SYSTEMD=0
```

And change all occurrences of:

```
if [ -z "$DRACUT_SYSTEMD" ]; then
```

to:

```
if [ 1 ]; then
```

4. Edit /usr/lib/dracut/modules.d/90crypt/cryptroot-ask.sh and append it to "# Open LUKS device":

```
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint >&2
modprobe vfat >/dev/null >&2
modprobe ntfs >/dev/null >&2
for SFS in /dev/sd*; do
echo "> Trying device:$SFS..." >&2
mount ${SFS}1 $MountPoint -t vfat -r >&2 ||
mount ${SFS}1 $MountPoint -t ntfs -r >&2
if [ -f $MountPoint/$KeyFileName ]; then
echo "> keyfile got..." >&2
cp $MountPoint/$KeyFileName /tmp-keyfile >&2
luksfile=/tmp-keyfile
umount $MountPoint >&2
break
fi
done
```

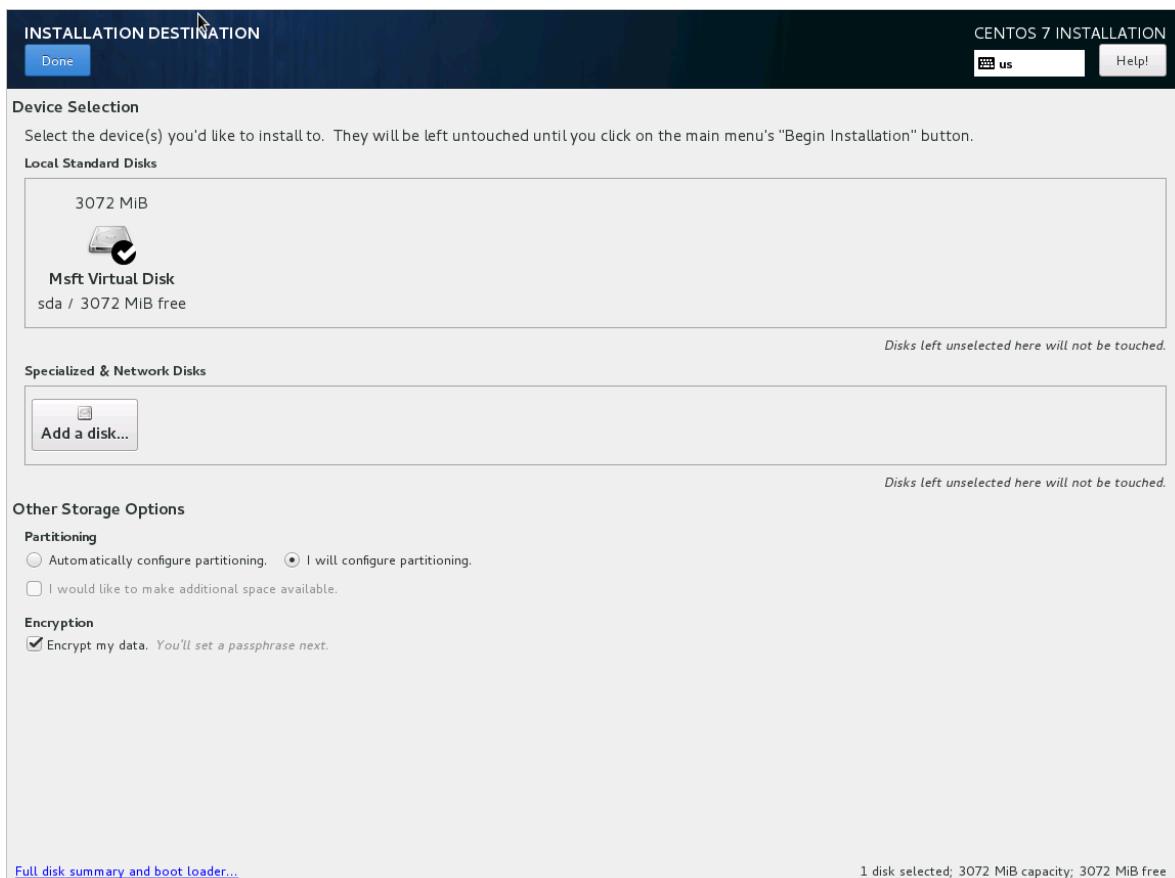
5. Run `/usr/sbin/dracut -f -v` to update the initrd.

6. Now you can deprovision the VM and [upload your VHD](#) into Azure.

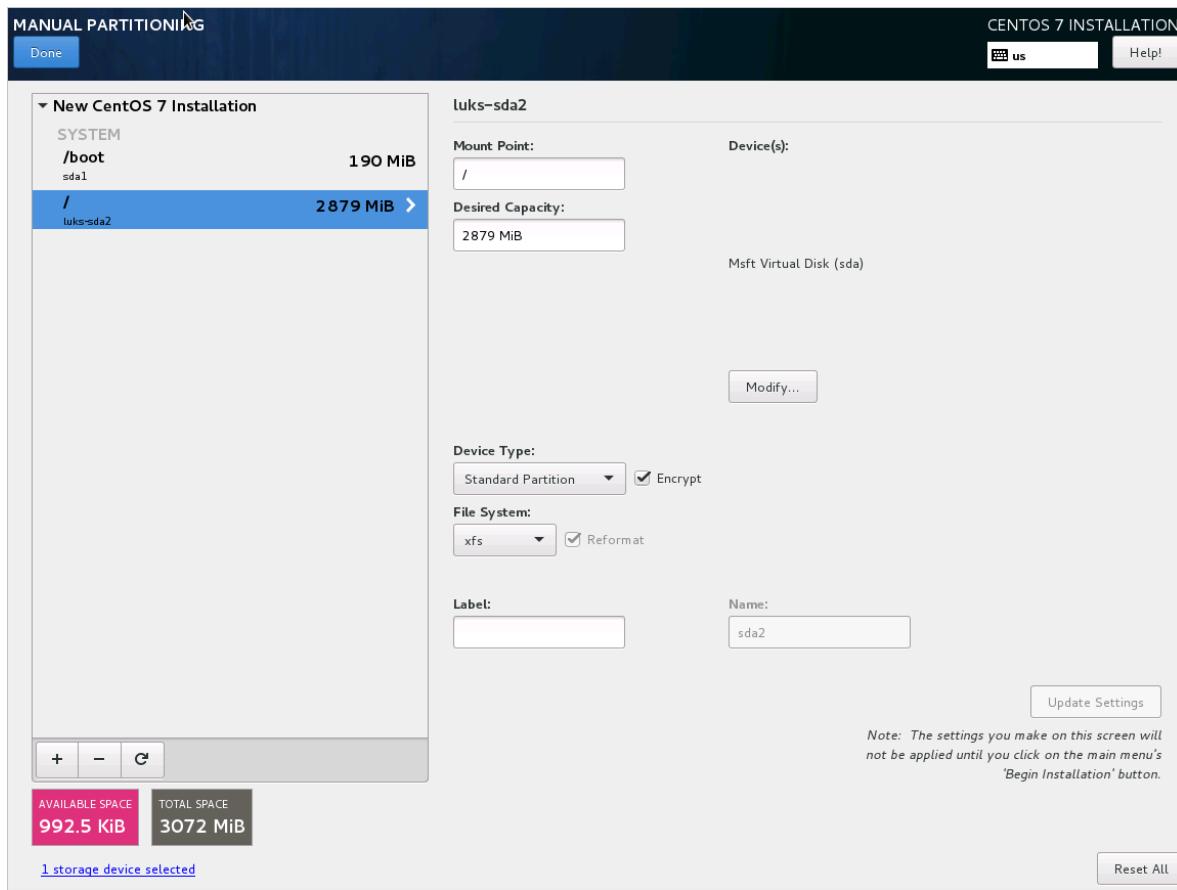
CentOS 7

To configure encryption during the distribution installation, do the following:

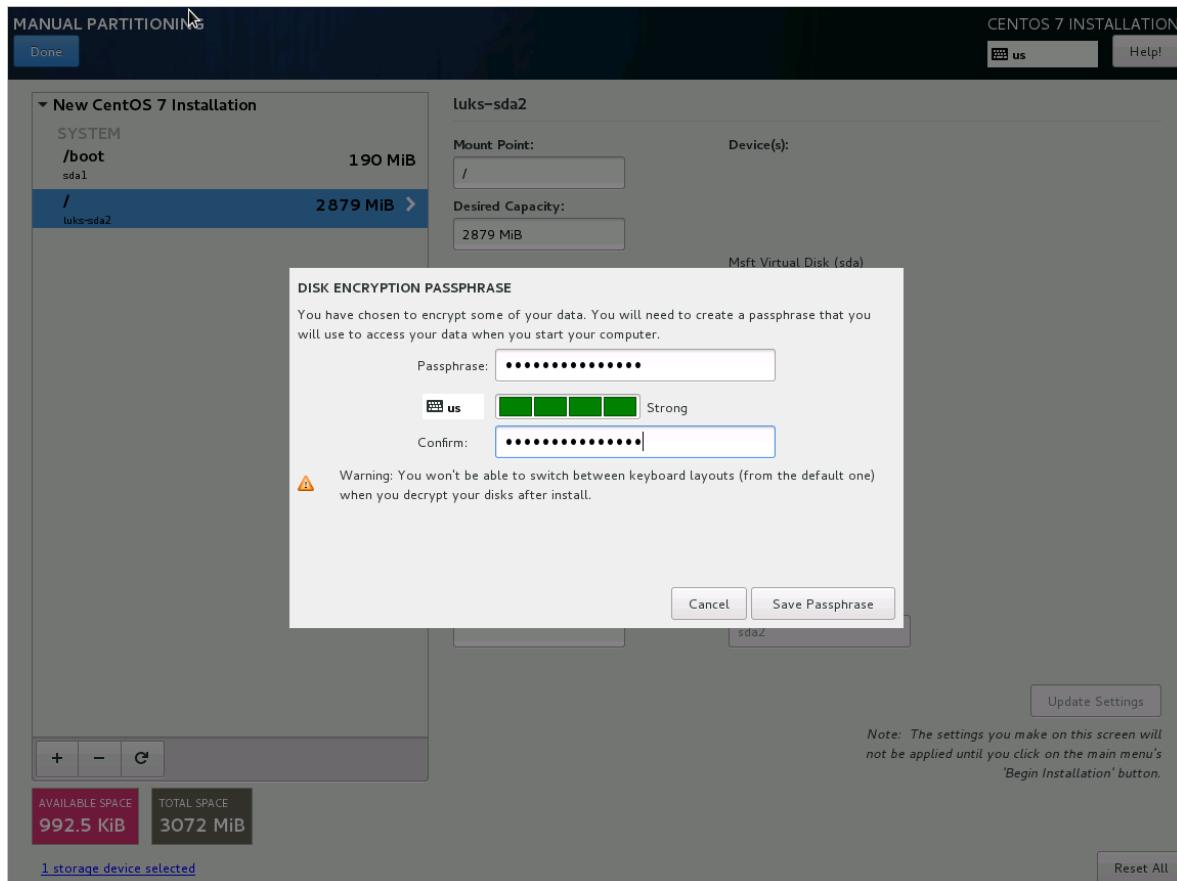
1. Select **Encrypt my data** when you partition disks.



2. Make sure **Encrypt** is selected for root partition.



- Provide a passphrase. This is the passphrase that you will upload to your key vault.



- When you boot the VM and are asked for a passphrase, use the passphrase you provided in step 3.



```
Please enter passphrase for disk Virtual_Disk (luks-4e88b050-991e-4283-9ec5-130ec245cadf)! :*****
```

5. Prepare the VM for uploading into Azure by using the "CentOS 7.0+" instructions in [Prepare a CentOS-based virtual machine for Azure](#). Do not run the last step (deprovisioning the VM) yet.
6. Now you can deprovision the VM and [upload your VHD](#) into Azure.

To configure encryption to work with Azure, do the following:

1. Edit the /etc/dracut.conf, and add the following line:

```
add_drivers+=" vfat ntfs nls_cp437 nls_iso8859-1"
```

2. Comment out these lines by the end of the file /usr/lib/dracut/modules.d/90crypt/module-setup.sh:

```
#      inst_multiple -o \
#      $systemdutildir/system-generators/systemd-cryptsetup-generator \
#      $systemdutildir/systemd-cryptsetup \
#      $systemdsystemunitdir/systemd-ask-password-console.path \
#      $systemdsystemunitdir/systemd-ask-password-console.service \
#      $systemdsystemunitdir/cryptsetup.target \
#      $systemdsystemunitdir/sysinit.target.wants/cryptsetup.target \
#      systemd-ask-password systemd-tty-ask-password-agent
#      inst_script "$moddir"/crypt-run-generator.sh /sbin/crypt-run-generator
```

3. Append the following line at the beginning of the file /usr/lib/dracut/modules.d/90crypt/parse-crypt.sh:

```
DRACUT_SYSTEMD=0
```

And change all occurrences of:

```
if [ -z "$DRACUT_SYSTEMD" ]; then
```

to

```
if [ 1 ]; then
```

4. Edit /usr/lib/dracut/modules.d/90crypt/cryptroot-ask.sh and append this after the "# Open LUKS device":

```
MountPoint=/tmp-keydisk-mount KeyFileName=LinuxPassPhraseFileName echo "Trying to get the key from disks ..." >&2 mkdir -p $MountPoint >&2 modprobe vfat >/dev/null >&2 modprobe ntfs >/dev/null >&2 for SFS in /dev/sd*; do echo "> Trying device:$SFS..." >&2 mount ${SFS}1 $MountPoint -t vfat -r >&2 || mount ${SFS}1 $MountPoint -t ntfs -r >&2 if [ -f $MountPoint/$KeyFileName ]; then echo "> keyfile got..." >&2 cp $MountPoint/$KeyFileName /tmp-keyfile >&2 luksfile=/tmp-keyfile umount $MountPoint >&2 break fi done
```

5. Run the "/usr/sbin/dracut -f -v" to update the initrd.

```
[root@centos-preencrypted ~]# cat /etc/dracut.conf | grep add_drivers
add_drivers+="vfat ntfs nls_cp437 nls_iso8859-1"
[root@centos-preencrypted ~]# cat /usr/lib/dracut/modules.d/90crypt/cryptroot-ask.sh | grep LinuxPassPhraseFileName -A 15 -B 1
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint >&2
modprobe vfat >/dev/null >&2
modprobe ntfs >/dev/null >&2
for SFS in /dev/sd*; do
echo "> Trying device:$SFS..." >&2
mount ${SFS}1 $MountPoint -t vfat -r >&2 ||
mount ${SFS}1 $MountPoint -t ntfs -r >&2
if [ -f $MountPoint/$KeyFileName ]; then
    echo "> keyfile got..." >&2
    cp $MountPoint/$KeyFileName /tmp-keyfile >&2
    luksfile=/tmp-keyfile
    umount $MountPoint >&2
    break
fi
[root@centos-preencrypted ~]# dracut -f -v_
```

Upload encrypted VHD to an Azure storage account

After BitLocker encryption or DM-Crypt encryption is enabled, the local encrypted VHD needs to be uploaded to your storage account.

```
Add-AzureRmVhd [-Destination] <Uri> [-LocalFilePath] <FileInfo> [[-NumberOfUploaderThreads] <Int32> ] [[-BaseImageUriToPatch] <Uri> ] [[-OverWrite]] [ <CommonParameters>]
```

Upload the disk-encryption secret for the pre-encrypted VM to your key vault

The disk-encryption secret that you obtained previously must be uploaded as a secret in your key vault. The key vault needs to have disk encryption and permissions enabled for your Azure AD client.

```

$AadClientId = "YourAADClientId"
$AadClientSecret = "YourAADClientSecret"

$keyVault = New-AzureRmKeyVault -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -Location $Location

Set-AzureRmKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -ServicePrincipalName $AadClientId -PermissionsToKeys all -PermissionsToSecrets all
Set-AzureRmKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -EnabledForDiskEncryption

```

Disk encryption secret not encrypted with a KEK

To set up the secret in your key vault, use [Set-AzureKeyVaultSecret](#). If you have a Windows virtual machine, the bek file is encoded as a base64 string and then uploaded to your key vault using the `Set-AzureKeyVaultSecret` cmdlet. For Linux, the passphrase is encoded as a base64 string and then uploaded to the key vault. In addition, make sure that the following tags are set when you create the secret in the key vault.

```

# This is the passphrase that was provided for encryption during the distribution installation
$passphrase = "contoso-password"

$tags = @{"DiskEncryptionKeyEncryptionAlgorithm" = "RSA-OAEP"; "DiskEncryptionKeyFileName" =
"LinuxPassPhraseFileName"}
$secretName = [guid]::NewGuid().ToString()
$secretValue = [Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($passphrase))
$secureSecretValue = ConvertTo-SecureString $secretValue -AsPlainText -Force

$secret = Set-AzureKeyVaultSecret -VaultName $KeyVaultName -Name $secretName -SecretValue $secureSecretValue
-tags $tags
$secretUrl = $secret.Id

```

Use the `$secretUrl` in the next step for [attaching the OS disk without using KEK](#).

Disk encryption secret encrypted with a KEK

Before you upload the secret to the key vault, you can optionally encrypt it by using a key encryption key. Use the [wrap API](#) to first encrypt the secret using the key encryption key. The output of this wrap operation is a base64 URL encoded string, which you can then upload as a secret by using the `Set-AzureKeyVaultSecret` cmdlet.

```

# This is the passphrase that was provided for encryption during the distribution installation
$passphrase = "contoso-password"

Add-AzureKeyVaultKey -VaultName $KeyVaultName -Name "keyencryptionkey" -Destination Software
$keyEncryptionKey = Get-AzureKeyVaultKey -VaultName $KeyVault.OriginalVault.Name -Name "keyencryptionkey"

$apiversion = "2015-06-01"

#####
# Get Auth URI
#####

$url = $KeyVault.VaultUri + "/keys"
$headers = @{}

$response = try { Invoke-RestMethod -Method GET -Uri $url -Headers $headers } catch { $_.Exception.Response }

$authHeader = $response.Headers["www-authenticate"]
$authUri = [regex]::match($authHeader, 'authorization="(.*)"').Groups[1].Value

Write-Host "Got Auth URI successfully"

#####
# Get Auth Token
#####

```

```

$uri = $authUri + "/oauth2/token"
$body = "grant_type=client_credentials"
$body += "&client_id=" + $AadClientId
$body += "&client_secret=" + [Uri]::EscapeDataString($AadClientSecret)
$body += "&resource=" + [Uri]::EscapeDataString("https://vault.azure.net")
$headers = @{}

$response = Invoke-RestMethod -Method POST -Uri $uri -Headers $headers -Body $body

$access_token = $response.access_token

Write-Host "Got Auth Token successfully"

#####
# Get KEK info
#####

$uri = $KeyEncryptionKey.Id + "?api-version=" + $apiversion
$headers = @{"Authorization" = "Bearer " + $access_token}

$response = Invoke-RestMethod -Method GET -Uri $uri -Headers $headers

$keyid = $response.key.kid

Write-Host "Got KEK info successfully"

#####
# Encrypt passphrase using KEK
#####

$passphraseB64 = [Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Passphrase))
$uri = $keyid + "/encrypt?api-version=" + $apiversion
$headers = @{"Authorization" = "Bearer " + $access_token; "Content-Type" = "application/json"}
$bodyObj = @{"alg" = "RSA-OAEP"; "value" = $passphraseB64}
$body = $bodyObj | ConvertTo-Json

$response = Invoke-RestMethod -Method POST -Uri $uri -Headers $headers -Body $body

$wrappedSecret = $response.value

Write-Host "Encrypted passphrase successfully"

#####
# Store secret
#####

$secretName = [guid]::NewGuid().ToString()
$uri = $KeyVault.VaultUri + "/secrets/" + $secretName + "?api-version=" + $apiversion
$secretAttributes = @{"enabled" = $true}
$secretTags = @{"DiskEncryptionKeyEncryptionAlgorithm" = "RSA-OAEP"; "DiskEncryptionKeyFileName" =
"LinuxPassPhraseFileName"}
$headers = @{"Authorization" = "Bearer " + $access_token; "Content-Type" = "application/json"}
$bodyObj = @{"value" = $wrappedSecret; "attributes" = $secretAttributes; "tags" = $secretTags}
$body = $bodyObj | ConvertTo-Json

$response = Invoke-RestMethod -Method PUT -Uri $uri -Headers $headers -Body $body

Write-Host "Stored secret successfully"

$secretUrl = $response.id

```

Use `$KeyEncryptionKey` and `$secretUrl` in the next step for [attaching the OS disk using KEK](#).

Specify a secret URL when you attach an OS disk

Without using a KEK

While you are attaching the OS disk, you need to pass `$secretUrl`. The URL was generated in the "Disk-

encryption secret not encrypted with a KEK" section.

```
Set-AzureRmVMOSDisk ` 
    -VM $VirtualMachine ` 
    -Name $OSDiskName ` 
    -SourceImageUri $VhdUri ` 
    -VhdUri $OSDiskUri ` 
    -Linux ` 
    -CreateOption FromImage ` 
    -DiskEncryptionKeyVaultId $KeyVault.ResourceId ` 
    -DiskEncryptionKeyUrl $SecretUrl
```

Using a KEK

When you attach the OS disk, pass `$KeyEncryptionKey` and `$secretUrl`. The URL was generated in the "Disk-encryption secret not encrypted with a KEK" section.

```
Set-AzureRmVMOSDisk ` 
    -VM $VirtualMachine ` 
    -Name $OSDiskName ` 
    -SourceImageUri $CopiedTemplateBlobUri ` 
    -VhdUri $OSDiskUri ` 
    -Linux ` 
    -CreateOption FromImage ` 
    -DiskEncryptionKeyVaultId $KeyVault.ResourceId ` 
    -DiskEncryptionKeyUrl $SecretUrl ` 
    -KeyEncryptionKeyVaultId $KeyVault.ResourceId ` 
    -KeyEncryptionKeyURL $KeyEncryptionKey.Id
```

Download this guide

You can download this guide from the [TechNet Gallery](#).

For more information

[Explore Azure Disk Encryption with Azure PowerShell - Part 1](#)

[Explore Azure Disk Encryption with Azure PowerShell - Part 2](#)

Azure Disk Encryption FAQ

9/1/2017 • 5 min to read • [Edit Online](#)

This article provides answers to frequently asked questions (FAQ) about Azure Disk Encryption for Windows and Linux IaaS VMs. For more information about this service, see [Azure Disk Encryption for Windows and Linux IaaS VMs](#).

General questions

Q: Where is Azure Disk Encryption in general availability (GA)?

A: Azure Disk Encryption for Windows and Linux IaaS VMs is in general availability in all Azure public regions.

Q: What user experiences are available with Azure Disk Encryption?

A: Azure Disk Encryption GA supports Azure Resource Manager templates, Azure PowerShell, and Azure CLI. This gives you a lot of flexibility. You have three different options for enabling disk encryption for your IaaS VMs. For more information on the user experience and step-by-step guidance available in Azure Disk Encryption, see Azure Disk Encryption deployment scenarios and experiences.

Q: How much does Azure Disk Encryption cost?

A: There is no charge for encrypting VM disks with Azure Disk Encryption.

Q: Which virtual machine tiers does Azure Disk Encryption support?

A: Azure Disk Encryption is available on standard tier VMs including [A](#), [D](#), [DS](#), [G](#), [GS](#), and [F](#) series IaaS VMs. It is also available for VMs with premium storage. It is not available on basic tier VMs.

Q: What Linux distributions does Azure Disk Encryption support?

A: Azure Disk Encryption is supported on the following Linux server distributions and versions:

LINUX DISTRIBUTION	VERSION	VOLUME TYPE SUPPORTED FOR ENCRYPTION
Ubuntu	16.04-DAILY-LTS	OS and data disk
Ubuntu	14.04.5-DAILY-LTS	OS and data disk
RHEL	7.3	OS and data disk
RHEL	7.2	OS and data disk
RHEL	6.8	OS and data disk
RHEL	6.7	Data disk
CentOS	7.3	OS and data disk
CentOS	7.2n	OS and data disk
CentOS	6.8	OS and data disk

LINUX DISTRIBUTION	VERSION	VOLUME TYPE SUPPORTED FOR ENCRYPTION
CentOS	7.1	Data disk
CentOS	7.0	Data disk
CentOS	6.7	Data disk
CentOS	6.6	Data disk
CentOS	6.5	Data disk
openSUSE	13.2	Data disk
SLES	12 SP1	Data disk
SLES	Priority:12-SP1	Data disk
SLES	HPC 12	Data disk
SLES	Priority:11-SP4	Data disk
SLES	11 SP4	Data disk

Q: How can I start using Azure Disk Encryption?

A: To get started, read the [Azure Disk Encryption for Windows and Linux IaaS VMs](#) white paper.

Q: Can I encrypt both boot and data volumes with Azure Disk Encryption?

A: Yes, you can encrypt boot and data volumes for Windows and Linux IaaS VMs. For Windows VMs, you cannot encrypt the data without first encrypting the OS volume. For Linux VMs, you can encrypt the data volume without having to encrypt the OS volume first. After you have encrypted the OS volume for Linux, disabling encryption on an OS volume for Linux IaaS VMs is not supported.

Q: Does Azure Disk Encryption allow you to bring your own key (BYOK) capability?

A: Yes, you can supply your own key encryption keys. These keys are safeguarded in Azure Key Vault, which is the key store for Azure Disk Encryption. For more information on the key encryption keys support scenarios, see Azure Disk Encryption deployment scenarios and experiences.

Q: Can I use an Azure-created key encryption key?

A: Yes, you can use Azure Key Vault to generate a key encryption key for Azure disk encryption use. These keys are safeguarded in Azure Key Vault, which is the key store for Azure Disk Encryption. For more information on the key encryption key support scenarios, see Azure Disk Encryption deployment scenarios and experiences.

Q: Can I use an on-premises key management service or HSM to safeguard the encryption keys?

A: You cannot use the on-premises key management service or HSM to safeguard the encryption keys with Azure Disk Encryption. You can only use the Azure Key Vault service to safeguard the encryption keys. For more information on the key encryption key support scenarios, see Azure Disk Encryption deployment scenarios and experiences.

Q: What are the prerequisites to configure Azure Disk Encryption?

A: There is a prerequisite PowerShell script. With this script, you can create an Azure Active Directory application, create a new key vault, or set up an existing key vault for disk encryption access to enable encryption and safeguard secrets and keys. For more information on the key encryption key support scenarios, see Azure Disk Encryption prerequisites and deployment scenarios and experiences.

Q: Where can I get more information on how to use PowerShell for configuring Azure Disk Encryption?

A: We have some great articles on how you can perform basic Azure Disk Encryption tasks, as well as more advanced scenarios. For the basic tasks, see [Explore Azure Disk Encryption with Azure PowerShell – Part 1](#). For more advanced scenarios, see [Explore Azure Disk Encryption with Azure PowerShell – Part 2](#).

Q: What version of Azure PowerShell does Azure Disk Encryption support?

A: Use the latest version of the Azure PowerShell SDK to configure Azure Disk Encryption. Download the latest version of [Azure PowerShell](#). Azure Disk Encryption is *not* supported by Azure SDK version 1.1.0.

NOTE

The Linux Azure disk encryption preview extension is deprecated. For details, see [Deprecating Azure disk encryption preview extension for Linux IaaS VMs](#).

Q: Can I apply Azure Disk Encryption on my custom Linux image?

A: You cannot apply Azure Disk Encryption on your custom Linux image. We support only the gallery Linux images for the supported distributions called out previously. We do not currently support custom Linux images.

Q: Can I apply updates to a Linux Red Hat VM that uses the yum update?

A: Yes, you can perform an update or patch a Red Hat Linux VM. For more information, see [Applying updates to an encrypted Azure IaaS Red Hat VM by using the yum update](#).

Q: What is the recommended Azure disk encryption workflow for Linux?

A: The following workflow is recommended to have the best results on Linux:

- Start from the unmodified stock gallery image corresponding to the desired OS distro and version
- Back up any mounted drives that will be encrypted. This permits recovery in case of failure, for example if the VM is rebooted before encryption has completed.
- Encrypt (can take multiple hours or even days depending on vm characteristics and size of any attached data disks)
- Customize, and add software to the image as needed.

If this workflow is not possible, relying on [Storage Service Encryption \(SSE\)](#) at the platform storage account layer may be an alternative to full disk encryption using dm-crypt.

Q: Where can I go to ask questions or provide feedback?

A: You can ask questions or provide feedback on the [Azure Disk Encryption forum](#).

Next steps

In this document, you learned more about the most frequent questions related to Azure Disk Encryption. For more information about this service and its capabilities, see the following articles:

- [Apply disk encryption in Azure Security Center](#)
- [Encrypt an Azure virtual machine](#)
- [Azure data encryption at rest](#)

Azure Disk Encryption troubleshooting guide

8/31/2017 • 5 min to read • [Edit Online](#)

This guide is for IT professionals, information security analysts, and cloud administrators whose organizations use Azure Disk Encryption and need guidance to troubleshoot disk-encryption-related problems.

Troubleshooting Linux OS disk encryption

Linux operating system (OS) disk encryption must unmount the OS drive before running it through the full disk encryption process. If it cannot unmount the drive, an error message of "failed to unmount after ..." is likely to occur.

This error is most likely to happen when OS disk encryption is attempted on a target VM environment that has been modified or changed from its supported stock gallery image. Examples of deviations from the supported image that can interfere with the extension's ability to unmount the OS drive include the following:

- Customized images no longer match a supported file system or partitioning scheme.
- Large applications such as SAP, MongoDB, or Apache Cassandra are installed and running in the OS prior to encryption. The extension cannot properly shut down these applications. If the applications maintain open file handles to the OS drive, the drive cannot be unmounted, causing failure.
- Custom scripts that run in close time proximity to the encryption being enabled, or if any other changes are being made on the VM during the encryption process. This conflict can happen when an Azure Resource Manager template defines multiple extensions to execute simultaneously, or when a custom script extension or other action runs simultaneously to disk encryption. Serializing and isolating such steps might resolve the issue.
- Security Enhanced Linux (SELinux) has not been disabled before enabling encryption, so the unmount step fails. SELinux can be reenabled after encryption is complete.
- The OS disk uses a Logical Volume Manager (LVM) scheme. Although limited LVM data disk support is available, an LVM OS disk is not.
- Minimum memory requirements are not met (7 GB is suggested for OS disk encryption).
- Data drives are recursively mounted under the /mnt/ directory, or each other (for example, /mnt/data1, /mnt/data2, /data3 + /data3/data4).
- Other Azure Disk Encryption [prerequisites](#) for Linux are not met.

Unable to encrypt

In some cases, the Linux disk encryption appears to be stuck at "OS disk encryption started" and SSH is disabled. The encryption process can take between 3-16 hours to finish on a stock gallery image. If multi-terabyte-sized data disks are added, the process might take days.

The Linux OS disk encryption sequence unmounts the OS drive temporarily. It then performs block-by-block encryption of the entire OS disk, before it remounts it in its encrypted state. Unlike Azure Disk Encryption on Windows, Linux Disk Encryption does not allow for concurrent use of the VM while the encryption is in progress. The performance characteristics of the VM can make a significant difference in the time required to complete encryption. These characteristics include the size of the disk and whether the storage account is standard or premium (SSD) storage.

To check the encryption status, poll the **ProgressMessage** field returned from the [Get-AzureRmVmDiskEncryptionStatus](#) command. While the OS drive is being encrypted, the VM enters a servicing state, and disables SSH to prevent any disruption to the ongoing process. The **EncryptionInProgress** message reports for the majority of the time while the encryption is in progress. Several hours later, a **VMRestartPending** message

prompts you to restart the VM. For example:

```
PS > Get-AzureRmVMDiskEncryptionStatus -ResourceGroupName $resourceGroupName -VMName $vmName
OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : EncryptionInProgress
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage        : OS disk encryption started

PS > Get-AzureRmVMDiskEncryptionStatus -ResourceGroupName $resourceGroupName -VMName $vmName
OsVolumeEncrypted      : VMRestartPending
DataVolumesEncrypted   : Encrypted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage        : OS disk successfully encrypted, please reboot the VM
```

After you are prompted to reboot the VM, and after the VM restarts, you must wait 2-3 minutes for the reboot and for the final steps to be performed on the target. The status message changes when the encryption is finally complete. After this message is available, the encrypted OS drive is expected to be ready for use and the VM is ready to be used again.

In the following cases, we recommend that you restore the VM back to the snapshot or backup taken immediately before encryption:

- If the reboot sequence described previously does not happen.
- If the boot information, progress message, or other error indicators report that OS encryption has failed in the middle of this process. An example of a message is the "failed to unmount" error that is described in this guide.

Prior to the next attempt, reevaluate the characteristics of the VM and ensure that all of the prerequisites are satisfied.

Troubleshooting Azure Disk Encryption behind a firewall

When connectivity is restricted by a firewall, proxy requirement, or network security group (NSG) settings, the ability of the extension to perform needed tasks might be disrupted. This disruption can result in status messages such as "Extension status not available on the VM." In expected scenarios, the encryption fails to finish. The sections that follow have some common firewall problems that you might investigate.

Network security groups

Any network security group settings that are applied must still allow the endpoint to meet the documented network configuration [prerequisites](#) for disk encryption.

Azure Key Vault behind a firewall

The VM must be able to access a key vault. Refer to guidance on access to the key vault from behind a firewall that the [Azure Key Vault](#) team maintains.

Linux package management behind a firewall

At runtime, Azure Disk Encryption for Linux relies on the target distribution's package management system to install needed prerequisite components prior to enabling encryption. If the firewall settings prevent the VM from being able to download and install these components, then subsequent failures are expected. The steps to configure this package management system can vary by distribution. On Red Hat, when a proxy is required, you must ensure that the subscription-manager and yum are set up properly. For more information, see [How to troubleshoot subscription-manager and yum problems](#).

Troubleshooting Windows Server 2016 Server Core

On Windows Server 2016 Server Core, the bdehdcfg component is not available by default. This component is required by Azure Disk Encryption. It is used to split the system volume from OS volume, which is done only once

for the life time of the VM. These binaries are not required during later encryption operations.

To workaround this issue, copy the following 4 files from a Windows Server 2016 Data Center VM to the same location on Server Core:

```
\windows\system32\bdehdcfg.exe  
\windows\system32\bdehdcfglib.dll  
\windows\system32\en-US\bdehdcfglib.dll.mui  
\windows\system32\en-US\bdehdcfg.exe.mui
```

1. Enter the following command:

```
bdehdcfg.exe -target default
```

2. This command creates a 550-MB system partition. Reboot the system.

3. Use DiskPart to check the volumes, and then proceed.

For example:

```
DISKPART> list vol
```

Volume #	Ltr	Label	Fs	Type	Size	Status	Info
Volume 0	C		NTFS	Partition	126 GB	Healthy	Boot
Volume 1			NTFS	Partition	550 MB	Healthy	System
Volume 2	D	Temporary S	NTFS	Partition	13 GB	Healthy	Pagefile

Next steps

In this document, you learned more about some common problems in Azure Disk Encryption and how to troubleshoot those problems. For more information about this service and its capabilities, see the following articles:

- [Apply disk encryption in Azure Security Center](#)
- [Encrypt an Azure virtual machine](#)
- [Azure data encryption at rest](#)

Encrypt an Azure Virtual Machine

6/27/2017 • 10 min to read • [Edit Online](#)

Azure Security Center will alert you if you have virtual machines that are not encrypted. These alerts will show as High Severity and the recommendation is to encrypt these virtual machines.

VIRTUAL MACHINES RECOMMENDATIONS		TOTAL			
Missing disk encryption	2 of 2 VMs	<div style="width: 100%; background-color: red; height: 10px;"></div>			
Virtual machines					
NAME	ONBOARDING	SYSTEM UPDATES	ANTIMALWARE	BASELINE	DISK ENCRYPTION
ASC-VM1	✓	✓	✓	✓	!
ASC-VM2	✓	✓	✓	✓	!

NOTE

The information in this document applies to encrypting virtual machines without using a Key Encryption Key (which is required for backing up virtual machines using Azure Backup). Please see the article [Azure Disk Encryption for Windows and Linux Azure Virtual Machines](#) for information on how to use a Key Encryption Key to support Azure Backup for encrypted Azure Virtual Machines.

To encrypt Azure Virtual Machines that have been identified by Azure Security Center as needing encryption, we recommend the following steps:

- Install and configure Azure PowerShell. This will enable you to run the PowerShell commands required to set up the prerequisites required to encrypt Azure Virtual Machines.
- Obtain and run the Azure Disk Encryption Prerequisites Azure PowerShell script
- Encrypt your virtual machines

The goal of this document is to enable you to encrypt your virtual machines, even if you have little or no background in Azure PowerShell. This document assumes you are using Windows 10 as the client machine from which you will configure Azure Disk Encryption.

There are many approaches that can be used to setup the prerequisites and to configure encryption for Azure Virtual Machines. If you are already well-versed in Azure PowerShell or Azure CLI, then you may prefer to use alternate approaches.

NOTE

To learn more about alternate approaches to configuring encryption for Azure virtual machines, please see [Azure Disk Encryption for Windows and Linux Azure Virtual Machines](#).

Install and configure Azure PowerShell

You need Azure PowerShell version 1.2.1 or above installed on your computer. The article [How to install and](#)

configure Azure PowerShell contains all the steps you need to provision your computer to work with Azure PowerShell. The most straightforward approach is to use the Web PI installation approach mentioned in that article. Even if you already have Azure PowerShell installed, install again using the Web PI approach so that you have the latest version of Azure PowerShell.

Obtain and run the Azure disk encryption prerequisites configuration script

The Azure Disk Encryption Prerequisites Configuration Script will set up all the prerequisites required for encrypting your Azure Virtual Machines.

1. Go to the GitHub page that has the [Azure Disk Encryption Prerequisite Setup Script](#).
2. On the GitHub page, click the **Raw** button.
3. Use **CTRL-A** to select all the text on the page and then use **CTRL-C** to copy all the text on the page to the clipboard.
4. Open **Notepad** and paste the copied text into Notepad.
5. Create a new folder on your C: drive named **AzureADEScript**.
6. Save the Notepad file – click **File**, then click **Save As**. In the File name textbox, enter “**ADEPrereqScript.ps1**” and click **Save**. (make sure you put the quotation marks around the name, otherwise it will save the file with a .txt file extension).

Now that the script content is saved, open the script in the PowerShell ISE:

1. In the Start Menu, click **Cortana**. Ask **Cortana** “PowerShell” by typing **PowerShell** in the Cortana search text box.
2. Right click **Windows PowerShell ISE** and click **Run as administrator**.
3. In the **Administrator: Windows PowerShell ISE** window, click **View** and then click **Show Script Pane**.
4. If you see the **Commands** pane on the right side of the window, click the “x” in the top right corner of the pane to close it. If the text is too small for you to see, use **CTRL+Add** (“Add” is the “+” sign). If the text is too large, use **CTRL+Subtract** (Subtract is the “-” sign).
5. Click **File** and then click **Open**. Navigate to the **C:\AzureADEScript** folder and the double-click on the **ADEPrereqScript**.
6. The **ADEPrereqScript** contents should now appear in the PowerShell ISE and is color-coded to help you see various components, such as commands, parameters and variables more easily.

You should now see something like the figure below.

Administrator: Windows PowerShell ISE

File Edit View Tools Debug Add-ons Help

Untitled1.ps1 2TomADEPrereqScript.ps1 ADEPrereqScript.ps1 X

```
11 [Parameter(Mandatory = $true,
12     HelpMessage="Location of the KeyVault. Important note: Make sure the Key
13     [ValidateNotNullOrEmpty()]
14     [string]$location,
15
16 [Parameter(Mandatory = $true,
17     HelpMessage="Name of the AAD application that will be used to write secr
18     [ValidateNotNullOrEmpty()]
19     [string]$aadAppName,
20
21 [Parameter(Mandatory = $false,
22     HelpMessage="Client secret of the AAD application that was created earli
23     [ValidateNotNullOrEmpty()]
24     [string]$aadClientSecret,
25
26 [Parameter(Mandatory = $false,
27     HelpMessage="Identifier of the Azure subscription to be used. Default su
28     [ValidateNotNullOrEmpty()]
29     [string]$subscriptionId,
30
31 [Parameter(Mandatory = $false,
32     HelpMessage="Name of optional key encryption key in KeyVault. A new key
33     [ValidateNotNullOrEmpty()]
34     [string]$keyEncryptionKeyName
35
36 )
37
38 #####
39 # Section1: Log-in to Azure and select appropriate subscription.
40 #####
41
42
```

PS C:\windows\System32>

Completed Ln 1 Col 25 130%

The top pane is referred to as the "script pane" and the bottom pane is referred to as the "console". We will use these terms later in this article.

Run the Azure disk encryption prerequisites PowerShell command

The Azure Disk Encryption Prerequisites script will ask you for the following information after you start the script:

- **Resource Group Name** - Name of the Resource Group that you want to put the Key Vault into. A new Resource Group with the name you enter will be created if there isn't already one with that name created. If you already have a Resource Group that you want to use in this subscription, then enter the name of that Resource Group.
 - **Key Vault Name** - Name of the Key Vault in which encryption keys are to be placed. A new Key Vault with this name will be created if you don't already have a Key Vault with this name. If you already have a Key Vault that you want to use, enter the name of the existing Key Vault.
 - **Location** - Location of the Key Vault. Make sure the Key Vault and VMs to be encrypted are in the same location. If you don't know the location, there are steps later in this article that will show you how to find out.
 - **Azure Active Directory Application Name** - Name of the Azure Active Directory application that will be used to write secrets to the Key Vault. A new application with this name will be created if one doesn't exist. If you already have an Azure Active Directory application that you want to use, enter the name of that Azure Active Directory application.

NOTE

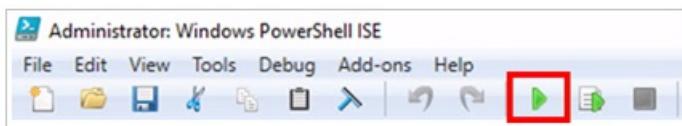
If you're curious as to why you need to create an Azure Active Directory application, please see *Register an application with Azure Active Directory* section in the article [Getting Started with Azure Key Vault](#).

Perform the following steps to encrypt an Azure Virtual Machine:

1. If you closed the PowerShell ISE, open an elevated instance of the PowerShell ISE. Follow the instructions earlier in this article if the PowerShell ISE is not already open. If you closed the script, then open the **ADEPrereqScript.ps1** clicking **File**, then **Open** and selecting the script from the **c:\AzureADEScript** folder. If you have followed this article from the start, then just move on to the next step.
2. In the console of the PowerShell ISE (the bottom pane of the PowerShell ISE), change the focus to the local of the script by typing **cd c:\AzureADEScript** and press **ENTER**.
3. Set the execution policy on your machine so that you can run the script. Type **Set-ExecutionPolicy Unrestricted** in the console and then press **ENTER**. If you see a dialog box telling about the effects of the change to execution policy, click either **Yes to all** or **Yes** (if you see **Yes to all**, select that option – if you do not see **Yes to all**, then click **Yes**).
4. Log into your Azure account. In the console, type **Login-AzureRmAccount** and press **ENTER**. A dialog box will appear in which you enter your credentials (make sure you have rights to change the virtual machines – if you do not have rights, you will not be able to encrypt them. If you are not sure, ask your subscription owner or administrator). You should see information about your **Environment**, **Account**, **TenantId**, **SubscriptionId** and **CurrentStorageAccount**. Copy the **SubscriptionId** to Notepad. You will need to use this in step #6.
5. Find what subscription your virtual machine belongs to and its location. Go to <https://portal.azure.com> and log in. On the left side of the page, click **Virtual Machines**. You will see a list of your virtual machines and the subscriptions they belong to.

NAME	STATUS	RESOURCE GROUP	LOCATION	SUBSCRIPTION
ASC-VM1	Running	ASC-ResourceGroup	Central US	Microsoft Azure Internal Consumption
ASC-VM2	Running	ASC-ResourceGroup	Central US	Microsoft Azure Internal Consumption
TomVM7	Running	TomRG7	Central US	Microsoft Azure Internal Consumption

6. Return to the PowerShell ISE. Set the subscription context in which the script will be run. In the console, type **Select-AzureRmSubscription -SubscriptionId** (replace < **your_subscription_Id** > with your actual Subscription ID) and press **ENTER**. You will see information about the Environment, **Account**, **TenantId**, **SubscriptionId** and **CurrentStorageAccount**.
7. You are now ready to run the script. Click the **Run Script** button or press **F5** on the keyboard.



8. The script asks for **resourceGroupName**: - enter the name of *Resource Group* you want to use, then press **ENTER**. If you don't have one, enter a name you want to use for a new one. If you already have a *Resource Group* that you want to use (such as the one that your virtual machine is in), enter the name of the existing *Resource Group*.
9. The script asks for **keyVaultName**: - enter the name of the *Key Vault* you want to use, then press **ENTER**. If you don't have one, enter a name you want to use for a new one. If you already have a *Key Vault* that you want to use, enter the name of the existing *Key Vault*.
10. The script asks for **location**: - enter the name of the location in which the VM you want to encrypt is located, then press **ENTER**. If you don't remember the location, go back to step #5.
11. The script asks for **aadAppName**: - enter the name of the *Azure Active Directory* application you want to use, then press **ENTER**. If you don't have one, enter a name you want to use for a new one. If you already have an *Azure Active Directory application* that you want to use, enter the name of the existing *Azure Active Directory application*.
12. A log in dialog box will appear. Provide your credentials (yes, you have logged in once, but now you need to do

it again).

13. The script runs and when complete it will ask you to copy the values of the **aadClientId**, **aadClientSecret**, **diskEncryptionKeyVaultUrl**, and **keyVaultResourceId**. Copy each of these values to the clipboard and paste them into Notepad.
14. Return to the PowerShell ISE and place the cursor at the end of the last line, and press **ENTER**.

The output of the script should look something like the screen below:

```
PS C:\AzureADEScript> C:\AzureADEScript\ADEPrereqScript.ps1
cmdlet ADEPrereqScript.ps1 at command pipeline position 1
Supply values for the following parameters:
(Type !? for Help.)
resourceGroupName: ASC-ResourceGroup
keyVaultName: ASC-KV3
location: Central US
aadAppName: ASC-AADapp3
Please log into Azure now
Creating new AAD application (ASC-AADapp3)
Created a new AAD Application (ASC-AADapp3) with ID: 8919cda0-f00b-402b-b94b-
b6faf8716fa3
Creating new key vault: (ASC-KV3)
Created a new KeyVault named ASC-KV3 to store encryption keys
Please note down below aadClientId, aadClientSecret, diskEncryptionKeyVaultUrl,
keyVaultResourceId values that will be needed to enable encryption on your VMs
  aadClientId: 8919cda0-f00b-402b-b94b-b6faf8716fa3
  aadClientSecret: 52ac8e6f-3f1d-4791-96ec-692efccc1301
  diskEncryptionKeyVaultUrl: https://ASC-KV3.vault.azure.net/
  keyVaultResourceId: /subscriptions/ad961f94-471b-43a3-aebd-
86dc84709961/resourceGroups/ASC-ResourceGroup/providers/Microsoft.KeyVault/vaults/ASC-
KV3
Please Press [Enter] after saving values displayed above. They are needed to enable
encryption using Set-AzureRmVmDiskEncryptionExtension cmdlet
```

Encrypt the Azure virtual machine

You are now ready to encrypt your virtual machine. If your virtual machine is located in the same Resource Group as your Key Vault, you can move on to the encryption steps section. However, if your virtual machine is not in the same Resource Group as your Key Vault, you will need to enter the following in the console in the PowerShell ISE:

\$resourceGroupName = <'Virtual_Machine_RG'>

Replace **< Virtual_Machine_RG >** with the name of the Resource Group in which your virtual machines are contained, surrounded by a single quote. Then press **ENTER**. To confirm that the correct Resource Group name was entered, type the following in the PowerShell ISE console:

\$resourceGroupName

Press **ENTER**. You should see the name of Resource Group that your virtual machines are located in. For example:

```
PS C:\AzureADEScript> $resourceGroupName = 'ASC-ResourceGroup'
PS C:\AzureADEScript> $resourceGroupName
ASC-ResourceGroup
```

Encryption steps

First, you need to tell PowerShell the name of the virtual machine you want to encrypt. In the console, type:

\$vmName = <'your_vm_name'>

Replace **<'your_vm_name'>** with the name of your VM (make sure the name is surrounded by a single quote) and then press **ENTER**.

To confirm that the correct VM name was entered, type:

\$vmName

Press **ENTER**. You should see the name of the virtual machine you want to encrypt. For example:

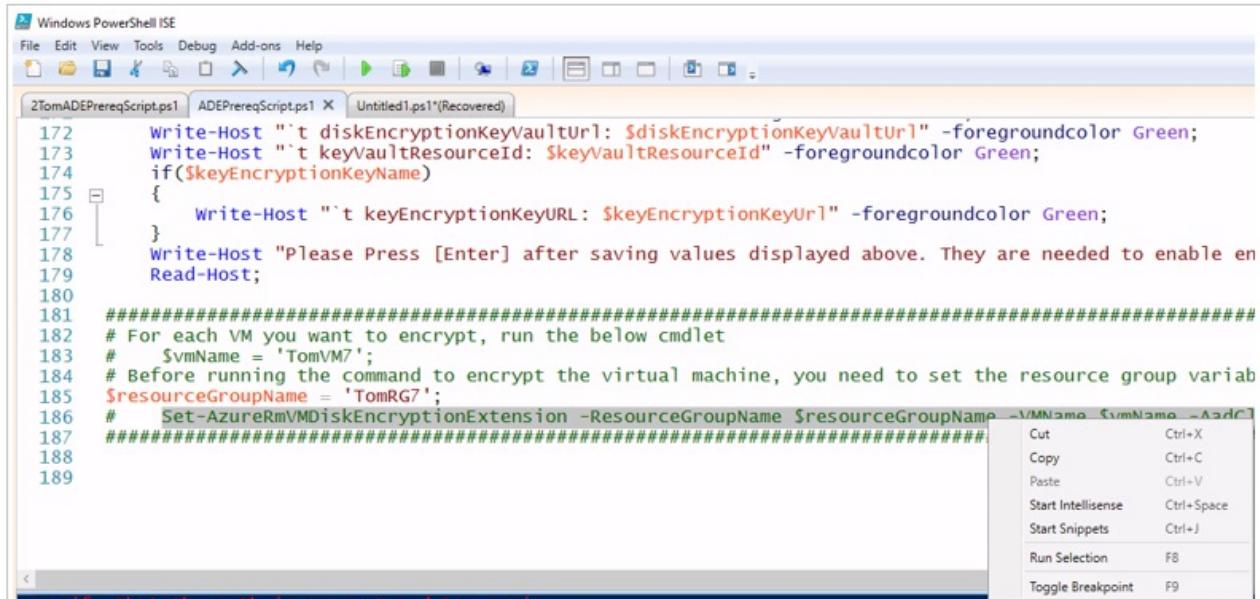
```
PS C:\AzureADEScript> $vmName = 'ASC-VM3'  
PS C:\AzureADEScript> $vmName  
ASC-VM3
```

There are two methods to run the encryption command to encrypt all drives on the virtual machine. The first method is to type the following command in the PowerShell ISE console:

```
Set-AzureRmVMDiskEncryptionExtension -ResourceGroupName $resourceGroupName -VMName $vmName -AadClientID  
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -  
DiskEncryptionKeyVaultId $keyVaultResourceId -VolumeType All
```

After typing this command press **ENTER**.

The second method is to click in the script pane (the top pane of the PowerShell ISE) and scroll down to the bottom of the script. Highlight the command listed above, and then right click it and then click **Run Selection** or press **F8** on the keyboard.



Regardless of the method you use, a dialog box will appear informing you that it will take 10-15 minutes for the operation to complete. Click **Yes**.

While the encryption process is taking place, you can return to the Azure Portal and see the status of the virtual machine. On the left side of the page, click **Virtual Machines**, then in the **Virtual Machines** blade, click the name of the virtual machine you're encrypting. In the blade that appears, you'll notice that the **Status** says that it's **Updating**. This demonstrates that encryption is in process.

Updating

Essentials ^

Resource group: **ASC-ResourceGroup**

Status: **Updating**

Location: **Central US**

Subscription name: **Microsoft Azure Internal Consumption**

Subscription ID: **ad961f94-471b-43a3-aebd-86dc84709961**

Computer name: -

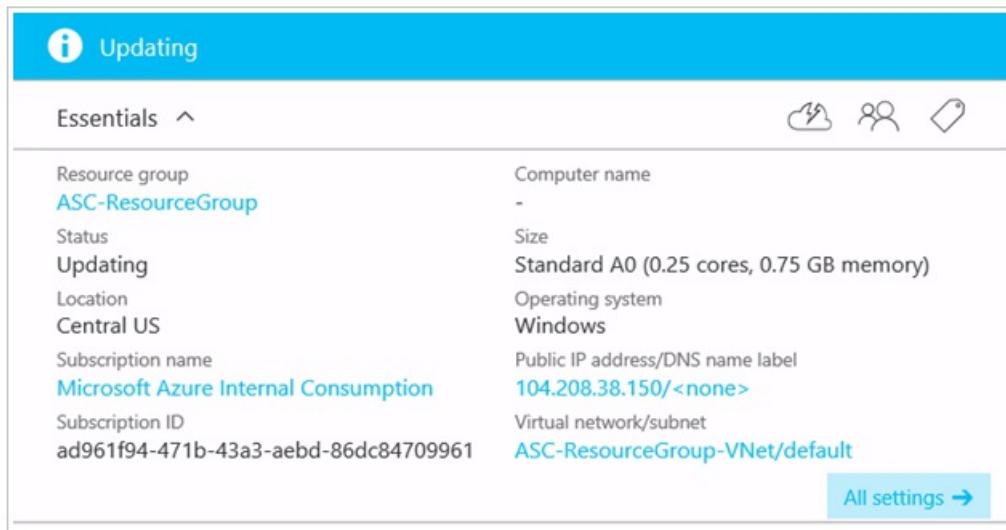
Size: **Standard A0 (0.25 cores, 0.75 GB memory)**

Operating system: **Windows**

Public IP address/DNS name label: **104.208.38.150/<none>**

Virtual network/subnet: **ASC-ResourceGroup-VNet/default**

[All settings →](#)



Return to the PowerShell ISE. When the script completes, you'll see what appears in the figure below.

RequestId	IsSuccess	Status	StatusCode	ReasonPhrase
	True	OK	OK	OK

To demonstrate that the virtual machine is now encrypted, return to the Azure Portal and click **Virtual Machines** on the left side of the page. Click the name of the virtual machine you encrypted. In the **Settings** blade, click **Disks**.

Settings
ASC-VM3

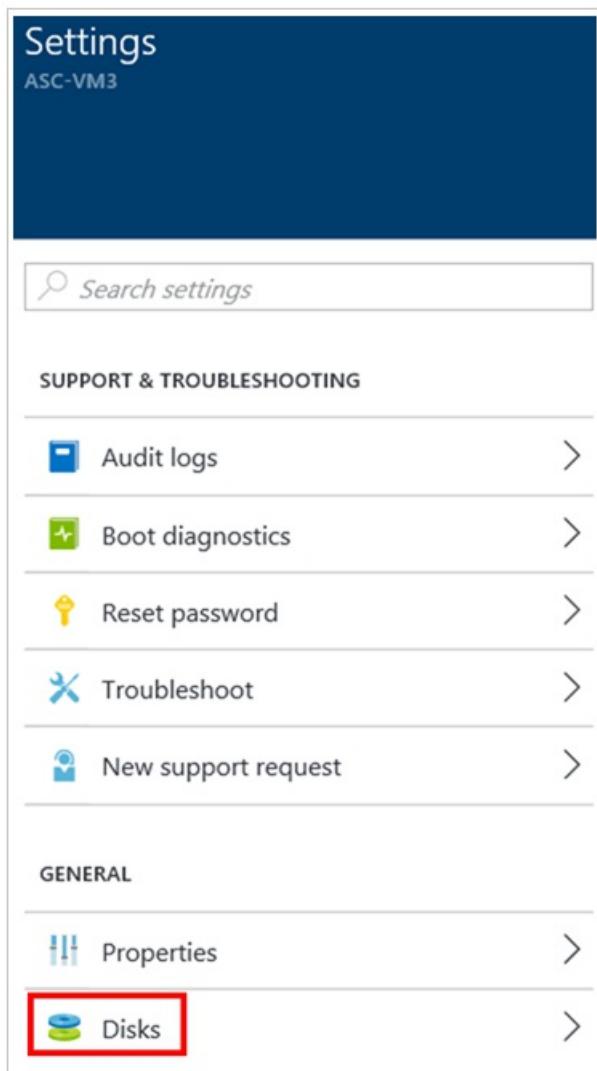
Search settings

SUPPORT & TROUBLESHOOTING

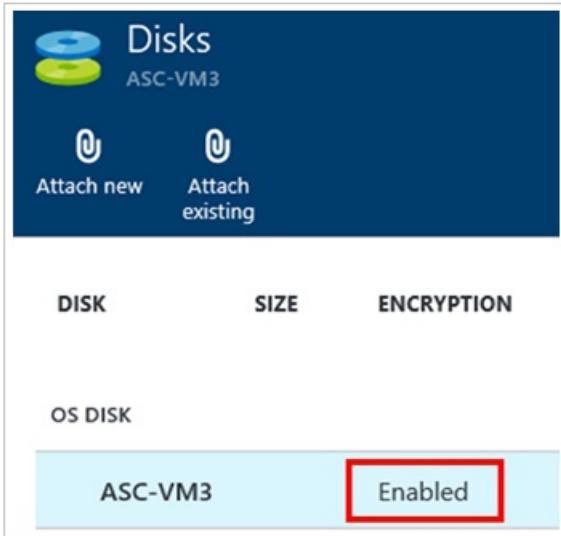
- Audit logs >
- Boot diagnostics >
- Reset password >
- Troubleshoot >
- New support request >

GENERAL

- Properties >
- Disks > **Disks**



On the **Disks** blade, you will see that **Encryption** is **Enabled**.



Next steps

In this document, you learned how to encrypt an Azure Virtual Machine. To learn more about Azure Security Center, see the following:

- [Security health monitoring in Azure Security Center](#) – Learn how to monitor the health of your Azure resources
- [Managing and responding to security alerts in Azure Security Center](#) - Learn how to manage and respond to security alerts
- [Azure Security Center FAQ](#) – Find frequently asked questions about using the service
- [Azure Security Blog](#) – Find blog posts about Azure security and compliance

Azure Operational Security best practices

8/9/2017 • 11 min to read • [Edit Online](#)

Azure Operational Security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure. Azure Operational Security is built on a framework that incorporates the knowledge gained through various capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Center program, and deep awareness of the cybersecurity threat landscape.

In this article, we discuss a collection of Azure database security best practices. These best practices are derived from our experience with Azure database security and the experiences of customers like yourself.

For each best practice, we explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- How you can learn to enable the best practice

This Azure Operational Security Best Practices article is based on a consensus opinion, and Azure platform capabilities and feature sets, as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure Operational Security best practices discussed in this article include:

- Monitor, manage, and protect cloud infrastructure
- Manage identity and implement single sign-on (SSO)
- Trace requests, analyze usage trends, and diagnose issues
- Monitoring services with a centralized monitoring solution
- Prevent, detect, and respond to threats
- End-to-end scenario-based network monitoring
- Secure deployment using proven DevOps tools

Monitor, manage, and protect cloud infrastructure

IT Operations is responsible for managing datacenter infrastructure, applications, and data, including the stability and security of these systems. However, gaining security insights across increasing complex IT environments often requires organizations to cobble together data from multiple security and management systems.

[Microsoft Operations Management Suite \(OMS\)](#) is Microsoft's cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure.

[OMS Security and Audit solution](#) enables IT to actively monitor all resources, which can help minimize the impact of security incidents. OMS Security and Audit have security domains that can be used for monitoring resources.

For more information about OMS, read the article [Operations Management Suite](#).

To help you prevent, detect, and respond to threats, [Operations Management Suite \(OMS\) Security and Audit Solution](#) collects and processes data about your resources, which includes:

- Security event log
- Event Tracing for Windows (ETW) events

- AppLocker auditing events
- Windows Firewall log
- Advanced Threat Analytics events
- Results of baseline assessment
- Results of antimalware assessment
- Results of update/patch assessment
- Syslog streams that are explicitly enabled on the agent

Manage identity and implement single sign-on

Azure Active Directory ([Azure AD](#)) is Microsoft's multi-tenant cloud based directory and identity management service.

Azure AD also includes a full suite of [identity management](#) capabilities including [multi-factor authentication](#), device registration, self-service password management, self-service group management, privileged account management, role-based access control, application usage monitoring, rich auditing and, security monitoring and alerting.

The following capabilities can help secure cloud-based applications, streamline IT processes, cut costs and help ensure that corporate compliance goals are met:

- Identity and access management for the cloud
- Simplify user access to any cloud app
- Protect sensitive data and applications
- Enable self-service for your employees
- Integrate with Azure Active Directory

Identity and access management for the cloud

Azure Active Directory (Azure AD) is a comprehensive [identity and access management cloud solution](#), which gives you a robust set of capabilities to manage users and groups. It helps secure access to on-premises and cloud applications, including Microsoft web services like Office 365 and much non-Microsoft software as a service (SaaS) applications. To learn more how to enable identity protection in Azure AD, see [Enabling Azure Active Directory Identity Protection](#).

Simplify user access to any cloud app

Enable [single sign-on](#) to simplify user access to thousands of cloud applications from Windows, Mac, Android, and iOS devices. Users can launch applications from a personalized web-based access panel or mobile app using their company credentials. Use the Azure AD Application Proxy module to go beyond SaaS applications and publish on-premises web applications to provide highly secure remote access and single sign-on.

Protect sensitive data and applications

Enable [Azure Multi-Factor Authentication](#) to prevent unauthorized access to on-premises and cloud applications by providing an additional level of authentication. Protect your business and mitigate potential threats with security monitoring, alerts, and machine learning-based reports that identify inconsistent access patterns.

Enable self-service for your employees

Delegate important tasks to your employees, such as resetting passwords and creating and managing groups. Enable [self-service password change](#), reset, and self-service group management with Azure AD.

Integrate with Azure Active Directory

Extend [Active Directory](#) and any other on-premises directories to Azure AD to enable single sign-on for all cloud-based applications. User attributes can be automatically synchronized to your cloud directory from all kinds of on-premises directories.

To learn more about integration of Azure Active Directory and how to enable it, please read the article [Integrate](#)

your on-premises directories with Azure Active Directory.

Trace requests, analyze usage trends, and diagnose issues

Azure Storage Analytics performs logging and provides metrics data for a storage account. You can use this data to trace requests, analyze usage trends, and diagnose issues with your storage account.

Storage Analytics metrics are enabled by default for new storage accounts. You can enable logging and configure both metrics and logging in the Azure portal; for details, see [Monitor a storage account in the Azure portal](#). You can also enable Storage Analytics programmatically via the REST API or the client library. Use the Set Service Properties operation to enable Storage Analytics individually for each service.

For an in-depth guide on using Storage Analytics and other tools to identify, diagnose, and troubleshoot Azure Storage-related issues, see [Monitor, diagnose, and troubleshoot Microsoft Azure Storage](#).

To learn more about integration of Azure Active Directory and how to enable it, read the article [Enabling and Configuring Storage Analytics](#).

Monitoring services

Cloud applications are complex with many moving parts. Monitoring provides data to ensure that your application stays up and running in a healthy state. It also helps you to stave off potential problems or troubleshoot past ones. In addition, you can use monitoring data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention.

Monitor Azure resources

Azure Monitor is the platform service that provides a single source for monitoring Azure resources. With Azure Monitor, you can visualize, query, route, archive, and take action on the metrics and logs coming from resources in Azure. You can work with this data using the Monitor portal blade, [Monitor PowerShell Cmdlets](#), [Cross-Platform CLI](#), or [Azure Monitor REST APIs](#).

Enable Autoscale with Azure monitor

Enable [Azure Monitor Autoscale](#) applies only to virtual machine scale sets (VMSS), cloud services, app service plans, and app service environments.

Manage Roles Permissions and Security

Many teams need to strictly [regulate access to monitoring](#) data and settings. For example, if you have team members who work exclusively on monitoring (support engineers, devops engineers) or if you use a managed service provider, you may want to grant them access to only monitoring data while restricting their ability to create, modify, or delete resources.

This shows how to quickly apply a built-in monitoring RBAC role to a user in Azure or build your own custom role for a user who needs limited monitoring permissions. It then discusses security considerations for your Azure Monitor-related resources and how you can limit access to the data they contain.

Prevent, detect, and respond to threats

Security Center threat detection works by automatically collecting security information from Azure resources, the network, and connected partner solutions. It analyses this information, often correlating information from multiple sources, to identify threats. Security alerts are prioritized in Security Center along with recommendations on how to remediate the threat.

- [Configure a security policy](#) for your Azure subscription.
- Use the [recommendations in Security Center](#) to help you protect your Azure resources.

- Review and manage your current [security alerts](#).

[Azure Security Center](#) helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Security Center delivers easy-to-use and effective threat prevention, detection, and response capabilities that are built in to Azure. Key capabilities are:

- Understand cloud security state
- Take control of cloud security
- Easily deploy integrated cloud security solutions
- Detect threats and respond fast

Understand cloud security state

Use Azure Security Center to get a central view of the security state of all of your Azure resources. At a glance, verify that the appropriate security controls are in place and configured correctly and quickly identify any resources, which require attention.

Take control of cloud security

Define [security policies](#) for your Azure subscriptions according to your company's cloud security needs, tailored to the type of applications or sensitivity of the data in each subscription. Use policy-driven recommendations to guide resource owners through the process of implementing required controls—take the guesswork out of cloud security.

Easily deploy integrated cloud security solutions

Enable [security solutions](#) from Microsoft and its partners, including industry-leading firewalls and antimalware. Use streamlined provisioning to deploy security solutions—even networking changes are configured for you. Your security events from partner solutions are automatically collected for analysis and alerting.

Detect threats and respond fast

Stay ahead of current and emerging cloud threats with an integrated, analytics-driven approach. By combining Microsoft global [threat intelligence](#) and expertise, with insights into cloud security-related events across your Azure deployments, Security Center helps you detect actual threats early and reduce false positives. Cloud security alerts give you insights into the attack campaign, including related events and impacted resources and suggest ways to remediate issues and recover quickly.

End-to-end scenario-based network monitoring

Customers build an end-to-end network in Azure by orchestrating and composing various individual network resources such as VNet, ExpressRoute, Application Gateway, Load balancers, and more. Monitoring is available on each of the network resources.

[Network Watcher](#) is a regional service that enables you to monitor and diagnose conditions at a network scenario level in, to, and from Azure. Network diagnostic and visualization tools available with Network Watcher help you understand, diagnose, and gain insights to your network in Azure.

Automate remote network monitoring with packet capture

Monitor and diagnose networking issues without logging in to your virtual machines (VMs) using Network Watcher. Trigger [packet capture](#) by setting alerts and gain access to real-time performance information at the packet level. When you see an issue, you can investigate in detail for better diagnoses.

Gain insight into your network traffic using flow logs

Build a deeper understanding of your network traffic pattern using [Network Security Group flow logs](#). Information

provided by flow logs helps you gather data for compliance, auditing and monitoring your network security profile.

Diagnose VPN connectivity issues

Network Watcher provides you the ability to [diagnose your most common VPN Gateway and Connections issues](#). Allowing you not only to identify the issue but also to use the detailed logs created to help further investigate.

To learn more about how to configure Network watcher and how to enable it, please read the article [configure Network watcher](#).

Secure deployment using proven DevOps tools

These are some of the List of Azure DevOps Practices in this Microsoft Cloud space, which makes enterprises and teams productive and efficient.

- **Infrastructure as Code (IaC):** Infrastructure as Code is a set of techniques and practices, which help IT Pros remove the burden associated with the day to day build and management of modular infrastructure. It allows IT Pros to build and maintain their modern server environment in a way that is like how software developers build and maintain application code. For Azure, we have [Azure Resource Manager](#) allows you to provision your applications using a declarative template. In a single template, you can deploy multiple services along with their dependencies. You use the same template to repeatedly deploy your application during every stage of the application lifecycle.
- **Continuous Integration and Deployment:** You can configure your Visual Studio Online team projects to [automatically build and deploy](#) to Azure web apps or cloud services. VSO automatically deploys the binaries after doing a build to Azure after every code check-in. The package build process described here is equivalent to the Package command in Visual Studio, and the publishing steps are equivalent to the Publish command in Visual Studio.
- **Release Management:** Visual Studio [Release Management](#) is a great solution for automating multi-stage deployment and managing the release process. Create managed continuous deployment pipelines to release quickly, easily, and often. With Release Management, we can much automate our release process, and we can have predefined approval workflows. Deploy on-premises and to the cloud, extend, and customize as required.
- **App Performance Monitoring:** Detect issues, solve problems, and continuously improve your applications. Quickly diagnose any problems in your live application. Understand what your users do with it. Configuration is easy matter of adding JS code and a webconfig entry, and you see results within minutes in the portal with all the details.[App insights](#) helps enterprises for faster detection of issues & remediation.
- **Load Testing & Autoscale:** We can find performance problems in our app to improve deployment quality and to make sure our app is always up or available to cater to the business needs. Make sure your app can handle traffic for your next launch or marketing campaign. Start running cloud-based [load tests](#) in almost no time with Visual Studio Online.

Next steps

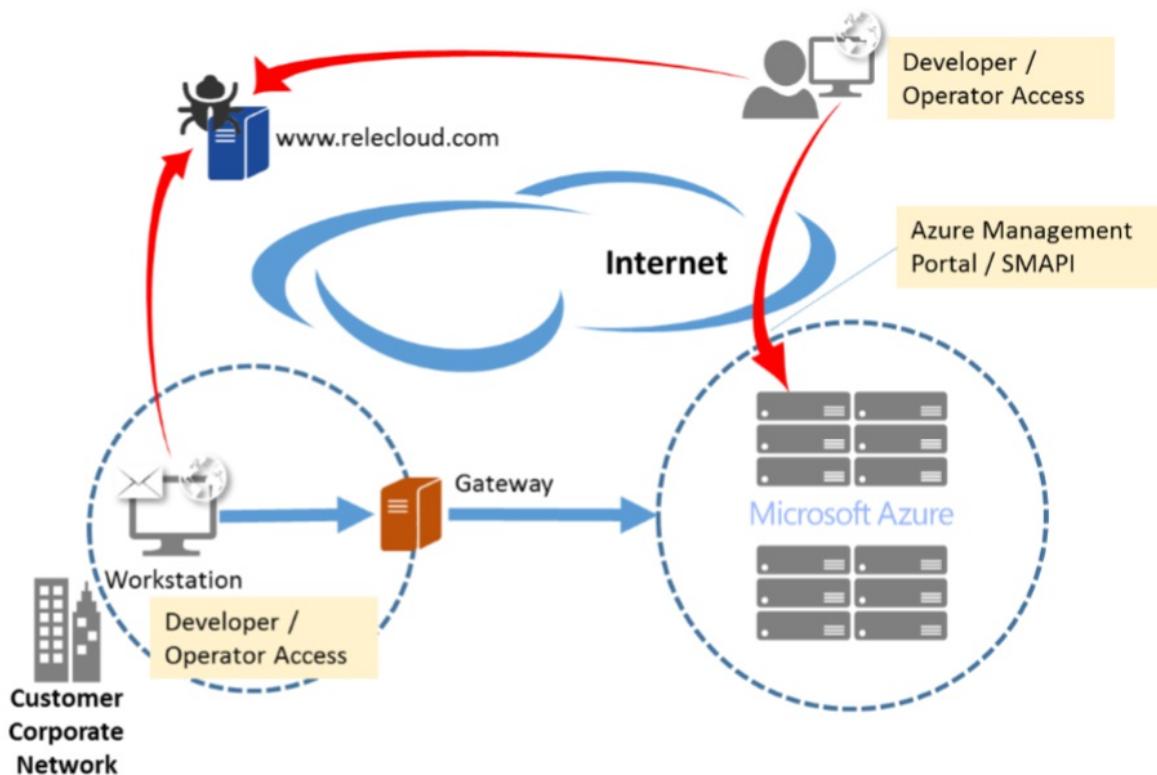
- Learn more about [Azure Operational Security](#).
- To Learn more [Operations Management Suite | Security & Compliance](#).
- [Getting started with Operations Management Suite Security and Audit Solution](#).

Security management in Azure

6/27/2017 • 20 min to read • [Edit Online](#)

Azure subscribers may manage their cloud environments from multiple devices, including management workstations, developer PCs, and even privileged end-user devices that have task-specific permissions. In some cases, administrative functions are performed through web-based consoles such as the [Azure portal](#). In other cases, there may be direct connections to Azure from on-premises systems over Virtual Private Networks (VPNs), Terminal Services, client application protocols, or (programmatically) the Azure Service Management API (SMAPI). Additionally, client endpoints can be either domain joined or isolated and unmanaged, such as tablets or smartphones.

Although multiple access and management capabilities provide a rich set of options, this variability can add significant risk to a cloud deployment. It can be difficult to manage, track, and audit administrative actions. This variability may also introduce security threats through unregulated access to client endpoints that are used for managing cloud services. Using general or personal workstations for developing and managing infrastructure opens unpredictable threat vectors such as web browsing (for example, watering hole attacks) or email (for example, social engineering and phishing).



The potential for attacks increases in this type of environment because it is challenging to construct security policies and mechanisms to appropriately manage access to Azure interfaces (such as SMAPI) from widely varied endpoints.

Remote management threats

Attackers often attempt to gain privileged access by compromising account credentials (for example, through password brute forcing, phishing, and credential harvesting), or by tricking users into running harmful code (for example, from harmful websites with drive-by downloads or from harmful email attachments). In a remotely managed cloud environment, account breaches can lead to an increased risk due to anywhere, anytime access.

Even with tight controls on primary administrator accounts, lower-level user accounts can be used to exploit weaknesses in one's security strategy. Lack of appropriate security training can also lead to breaches through accidental disclosure or exposure of account information.

When a user workstation is also used for administrative tasks, it can be compromised at many different points. Whether a user is browsing the web, using 3rd-party and open-source tools, or opening a harmful document file that contains a trojan.

In general, most targeted attacks that result in data breaches can be traced to browser exploits, plug-ins (such as Flash, PDF, Java), and spear phishing (email) on desktop machines. These machines may have administrative-level or service-level permissions to access live servers or network devices for operations when used for development or management of other assets.

Operational security fundamentals

For more secure management and operations, you can minimize a client's attack surface by reducing the number of possible entry points. This can be done through security principles: "separation of duties" and "segregation of environments."

Isolate sensitive functions from one another to decrease the likelihood that a mistake at one level leads to a breach in another. Examples:

- Administrative tasks should not be combined with activities that might lead to a compromise (for example, malware in an administrator's email that then infects an infrastructure server).
- A workstation used for high-sensitivity operations should not be the same system used for high-risk purposes such as browsing the Internet.

Reduce the system's attack surface by removing unnecessary software. Example:

- Standard administrative, support, or development workstation should not require installation of an email client or other productivity applications if the device's main purpose is to manage cloud services.

Client systems that have administrator access to infrastructure components should be subjected to the strictest possible policy to reduce security risks. Examples:

- Security policies can include Group Policy settings that deny open Internet access from the device and use of a restrictive firewall configuration.
- Use Internet Protocol security (IPsec) VPNs if direct access is needed.
- Configure separate management and development Active Directory domains.
- Isolate and filter management workstation network traffic.
- Use antimalware software.
- Implement multi-factor authentication to reduce the risk of stolen credentials.

Consolidating access resources and eliminating unmanaged endpoints also simplifies management tasks.

Providing security for Azure remote management

Azure provides security mechanisms to aid administrators who manage Azure cloud services and virtual machines. These mechanisms include:

- Authentication and [role-based access control](#).
- Monitoring, logging, and auditing.
- Certificates and encrypted communications.
- A web management portal.
- Network packet filtering.

With client-side security configuration and datacenter deployment of a management gateway, it is possible to restrict and monitor administrator access to cloud applications and data.

NOTE

Certain recommendations in this article may result in increased data, network, or compute resource usage, and may increase your license or subscription costs.

Hardened workstation for management

The goal of hardening a workstation is to eliminate all but the most critical functions required for it to operate, making the potential attack surface as small as possible. System hardening includes minimizing the number of installed services and applications, limiting application execution, restricting network access to only what is needed, and always keeping the system up to date. Furthermore, using a hardened workstation for management segregates administrative tools and activities from other end-user tasks.

Within an on-premises enterprise environment, you can limit the attack surface of your physical infrastructure through dedicated management networks, server rooms that have card access, and workstations that run on protected areas of the network. In a cloud or hybrid IT model, being diligent about secure management services can be more complex because of the lack of physical access to IT resources. Implementing protection solutions requires careful software configuration, security-focused processes, and comprehensive policies.

Using a least-privilege minimized software footprint in a locked-down workstation for cloud management—and for application development—can reduce the risk of security incidents by standardizing the remote management and development environments. A hardened workstation configuration can help prevent the compromise of accounts that are used to manage critical cloud resources by closing many common avenues used by malware and exploits. Specifically, you can use [Windows AppLocker](#) and Hyper-V technology to control and isolate client system behavior and mitigate threats, including email or Internet browsing.

On a hardened workstation, the administrator runs a standard user account (which blocks administrative-level execution) and associated applications are controlled by an allow list. The basic elements of a hardened workstation are as follows:

- Active scanning and patching. Deploy antimalware software, perform regular vulnerability scans, and update all workstations by using the latest security update in a timely fashion.
- Limited functionality. Uninstall any applications that are not needed and disable unnecessary (startup) services.
- Network hardening. Use Windows Firewall rules to allow only valid IP addresses, ports, and URLs related to Azure management. Ensure that inbound remote connections to the workstation are also blocked.
- Execution restriction. Allow only a set of predefined executable files that are needed for management to run (referred to as "default-deny"). By default, users should be denied permission to run any program unless it is explicitly defined in the allow list.
- Least privilege. Management workstation users should not have any administrative privileges on the local machine itself. This way, they cannot change the system configuration or the system files, either intentionally or unintentionally.

You can enforce all this by using [Group Policy Objects](#) (GPOs) in Active Directory Domain Services (AD DS) and applying them through your (local) management domain to all management accounts.

Managing services, applications, and data

Azure cloud services configuration is performed through either the Azure portal or SAPI, via the Windows PowerShell command-line interface or a custom-built application that takes advantage of these RESTful interfaces. Services using these mechanisms include Azure Active Directory (Azure AD), Azure Storage, Azure Websites, and Azure Virtual Network, and others.

Virtual Machine-deployed applications provide their own client tools and interfaces as needed, such as the Microsoft Management Console (MMC), an enterprise management console (such as Microsoft System Center or Windows Intune), or another management application—Microsoft SQL Server Management Studio, for example.

These tools typically reside in an enterprise environment or client network. They may depend on specific network protocols, such as Remote Desktop Protocol (RDP), that require direct, stateful connections. Some may have web-enabled interfaces that should not be openly published or accessible via the Internet.

You can restrict access to infrastructure and platform services management in Azure by using [multi-factor authentication](#), [X.509 management certificates](#), and firewall rules. The Azure portal and SAPI require Transport Layer Security (TLS). However, services and applications that you deploy into Azure require you to take protection measures that are appropriate based on your application. These mechanisms can frequently be enabled more easily through a standardized hardened workstation configuration.

Management gateway

To centralize all administrative access and simplify monitoring and logging, you can deploy a dedicated [Remote Desktop Gateway](#) (RD Gateway) server in your on-premises network, connected to your Azure environment.

A Remote Desktop Gateway is a policy-based RDP proxy service that enforces security requirements. Implementing RD Gateway together with Windows Server Network Access Protection (NAP) helps ensure that only clients that meet specific security health criteria established by Active Directory Domain Services (AD DS) Group Policy objects (GPOs) can connect. In addition:

- Provision an [Azure management certificate](#) on the RD Gateway so that it is the only host allowed to access the Azure portal.
- Join the RD Gateway to the same [management domain](#) as the administrator workstations. This is necessary when you are using a site-to-site IPsec VPN or ExpressRoute within a domain that has a one-way trust to Azure AD, or if you are federating credentials between your on-premises AD DS instance and Azure AD.
- Configure a [client connection authorization policy](#) to let the RD Gateway verify that the client machine name is valid (domain joined) and allowed to access the Azure portal.
- Use IPsec for [Azure VPN](#) to further protect management traffic from eavesdropping and token theft, or consider an isolated Internet link via [Azure ExpressRoute](#).
- Enable multi-factor authentication (via [Azure Multi-Factor Authentication](#)) or smart-card authentication for administrators who log on through RD Gateway.
- Configure source [IP address restrictions](#) or [Network Security Groups](#) in Azure to minimize the number of permitted management endpoints.

Security guidelines

In general, helping to secure administrator workstations for use with the cloud is similar to the practices used for any workstation on-premises—for example, minimized build and restrictive permissions. Some unique aspects of cloud management are more akin to remote or out-of-band enterprise management. These include the use and auditing of credentials, security-enhanced remote access, and threat detection and response.

Authentication

You can use Azure logon restrictions to constrain source IP addresses for accessing administrative tools and audit access requests. To help Azure identify management clients (workstations and/or applications), you can configure both SAPI (via customer-developed tools such as Windows PowerShell cmdlets) and the Azure portal to require client-side management certificates to be installed, in addition to SSL certificates. We also recommend that administrator access require multi-factor authentication.

Some applications or services that you deploy into Azure may have their own authentication mechanisms for both end-user and administrator access, whereas others take full advantage of Azure AD. Depending on whether you are federating credentials via Active Directory Federation Services (AD FS), using directory synchronization or maintaining user accounts solely in the cloud, using [Microsoft Identity Manager](#) (part of Azure AD Premium) helps you manage identity lifecycles between the resources.

Connectivity

Several mechanisms are available to help secure client connections to your Azure virtual networks. Two of these mechanisms, [site-to-site VPN](#) (S2S) and [point-to-site VPN](#) (P2S), enable the use of industry standard IPsec (S2S) or the [Secure Socket Tunneling Protocol](#) (SSTP) (P2S) for encryption and tunneling. When Azure is connecting to public-facing Azure services management such as the Azure portal, Azure requires Hypertext Transfer Protocol Secure (HTTPS).

A stand-alone hardened workstation that does not connect to Azure through an RD Gateway should use the SSTP-based point-to-site VPN to create the initial connection to the Azure Virtual Network, and then establish RDP connection to individual virtual machines from with the VPN tunnel.

Management auditing vs. policy enforcement

Typically, there are two approaches for helping to secure management processes: auditing and policy enforcement. Doing both provides comprehensive controls, but may not be possible in all situations. In addition, each approach has different levels of risk, cost, and effort associated with managing security, particularly as it relates to the level of trust placed in both individuals and system architectures.

Monitoring, logging, and auditing provide a basis for tracking and understanding administrative activities, but it may not always be feasible to audit all actions in complete detail due to the amount of data generated. Auditing the effectiveness of the management policies is a best practice, however.

Policy enforcement that includes strict access controls puts programmatic mechanisms in place that can govern administrator actions, and it helps ensure that all possible protection measures are being used. Logging provides proof of enforcement, in addition to a record of who did what, from where, and when. Logging also enables you to audit and crosscheck information about how administrators follow policies, and it provides evidence of activities

Client configuration

We recommend three primary configurations for a hardened workstation. The biggest differentiators between them are cost, usability, and accessibility, while maintaining a similar security profile across all options. The following table provides a short analysis of the benefits and risks to each. (Note that "corporate PC" refers to a standard desktop PC configuration that would be deployed for all domain users, regardless of roles.)

CONFIGURATION	BENEFITS	CONS
Stand-alone hardened workstation	Tightly controlled workstation	higher cost for dedicated desktops
-	Reduced risk of application exploits	Increased management effort
-	Clear separation of duties	-
Corporate PC as virtual machine	Reduced hardware costs	-
-	Segregation of role and applications	-
Windows to go with BitLocker drive encryption	Compatibility with most PCs	Asset tracking
-	Cost-effectiveness and portability	-
-	Isolated management environment	-

It is important that the hardened workstation is the host and not the guest, with nothing between the host operating system and the hardware. Following the "clean source principle" (also known as "secure origin") means that the host should be the most hardened. Otherwise, the hardened workstation (guest) is subject to attacks on the

system on which it is hosted.

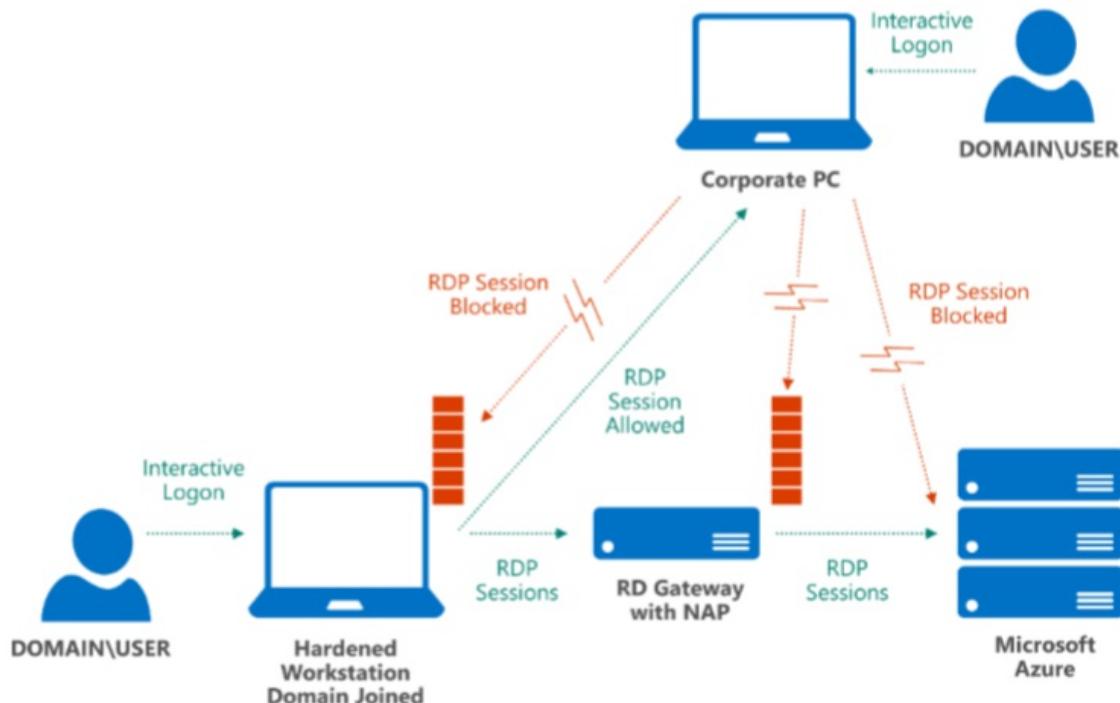
You can further segregate administrative functions through dedicated system images for each hardened workstation that have only the tools and permissions needed for managing select Azure and cloud applications, with specific local AD DS GPOs for the necessary tasks.

For IT environments that have no on-premises infrastructure (for example, no access to a local AD DS instance for GPOs because all servers are in the cloud), a service such as [Microsoft Intune](#) can simplify deploying and maintaining workstation configurations.

Stand-alone hardened workstation for management

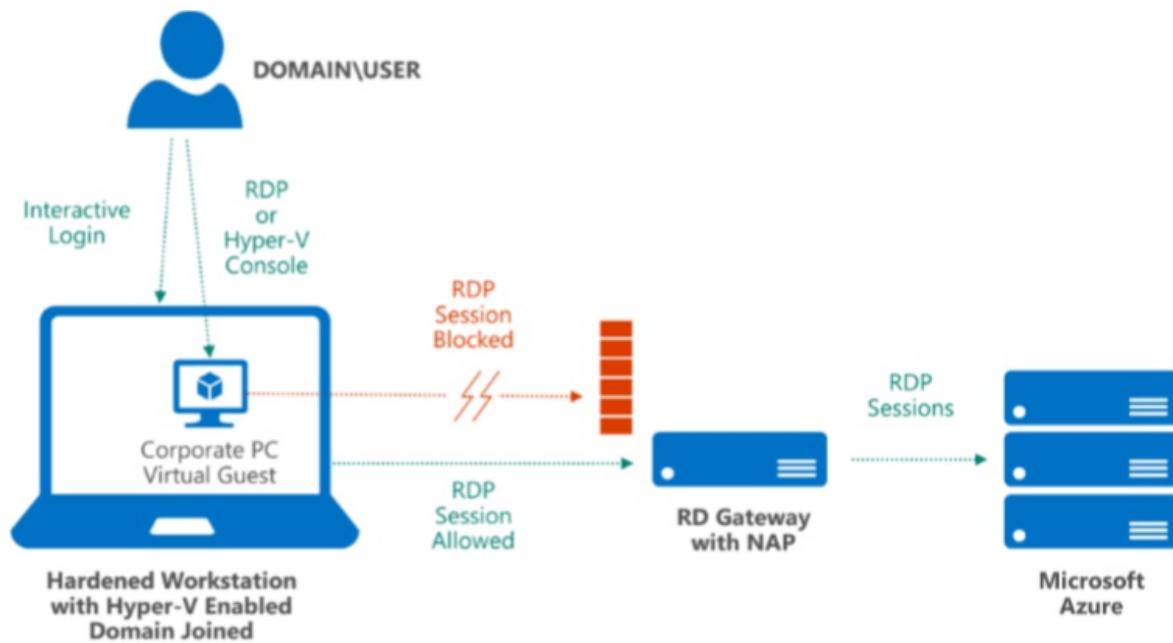
With a stand-alone hardened workstation, administrators have a PC or laptop that they use for administrative tasks and another, separate PC or laptop for non-administrative tasks. A workstation dedicated to managing your Azure services does not need other applications installed. Additionally, using workstations that support a [Trusted Platform Module \(TPM\)](#) or similar hardware-level cryptography technology aids in device authentication and prevention of certain attacks. TPM can also support full volume protection of the system drive by using [BitLocker Drive Encryption](#).

In the stand-alone hardened workstation scenario (shown below), the local instance of Windows Firewall (or a non-Microsoft client firewall) is configured to block inbound connections, such as RDP. The administrator can log on to the hardened workstation and start an RDP session that connects to Azure after establishing a VPN connect with an Azure Virtual Network, but cannot log on to a corporate PC and use RDP to connect to the hardened workstation itself.



Corporate PC as virtual machine

In cases where a separate stand-alone hardened workstation is cost prohibitive or inconvenient, the hardened workstation can host a virtual machine to perform non-administrative tasks.



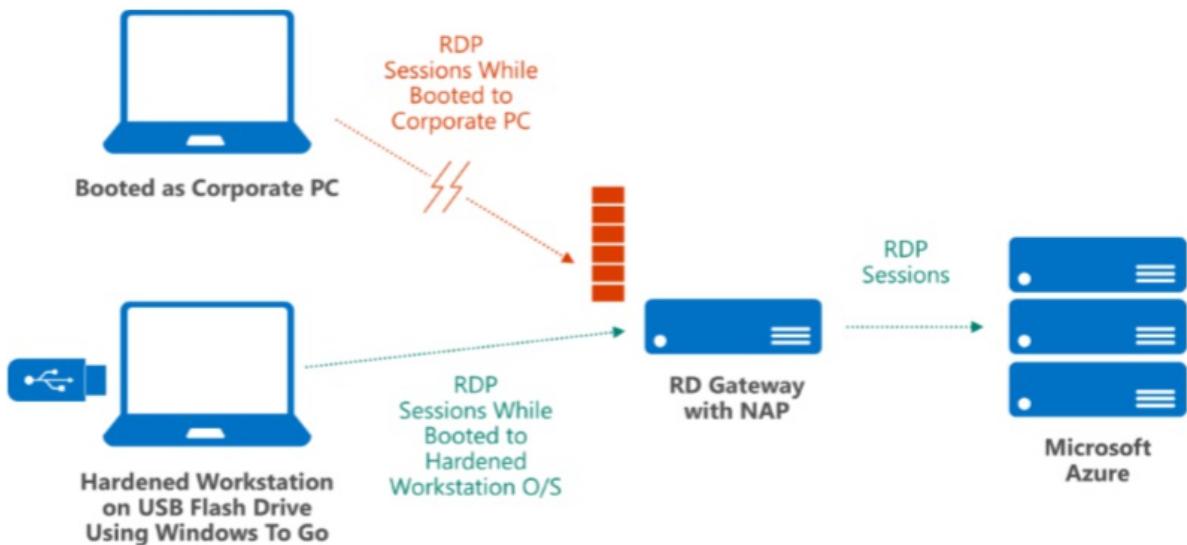
To avoid several security risks that can arise from using one workstation for systems management and other daily work tasks, you can deploy a Windows Hyper-V virtual machine to the hardened workstation. This virtual machine can be used as the corporate PC. The corporate PC environment can remain isolated from the Host, which reduces its attack surface and removes the user's daily activities (such as email) from coexisting with sensitive administrative tasks.

The corporate PC virtual machine runs in a protected space and provides user applications. The host remains a "clean source" and enforces strict network policies in the root operating system (for example, blocking RDP access from the virtual machine).

Windows To Go

Another alternative to requiring a stand-alone hardened workstation is to use a [Windows To Go](#) drive, a feature that supports a client-side USB-boot capability. Windows To Go enables users to boot a compatible PC to an isolated system image running from an encrypted USB flash drive. It provides additional controls for remote-administration endpoints because the image can be fully managed by a corporate IT group, with strict security policies, a minimal OS build, and TPM support.

In the figure below, the portable image is a domain-joined system that is preconfigured to connect only to Azure, requires multi-factor authentication, and blocks all non-management traffic. If a user boots the same PC to the standard corporate image and tries accessing RD Gateway for Azure management tools, the session is blocked. Windows To Go becomes the root-level operating system, and no additional layers are required (host operating system, hypervisor, virtual machine) that may be more vulnerable to outside attacks.



It is important to note that USB flash drives are more easily lost than an average desktop PC. Use of BitLocker to encrypt the entire volume, together with a strong password, makes it less likely that an attacker can use the drive image for harmful purposes. Additionally, if the USB flash drive is lost, revoking and [issuing a new management certificate](#) along with a quick password reset can reduce exposure. Administrative audit logs reside within Azure, not on the client, further reducing potential data loss.

Best practices

Consider the following additional guidelines when you are managing applications and data in Azure.

Dos and don'ts

Don't assume that because a workstation has been locked down that other common security requirements do not need to be met. The potential risk is higher because of elevated access levels that administrator accounts generally possess. Examples of risks and their alternate safe practices are shown in the table below.

DON'T	DO
Don't email credentials for administrator access or other secrets (for example, SSL or management certificates)	Maintain confidentiality by delivering account names and passwords by voice (but not storing them in voice mail), perform a remote installation of client/server certificates (via an encrypted session), download from a protected network share, or distribute by hand via removable media.
-	Proactively manage your management certificate life cycles.
Don't store account passwords unencrypted or un-hashed in application storage (such as in spreadsheets, SharePoint sites, or file shares).	Establish security management principles and system hardening policies, and apply them to your development environment.
-	Use Enhanced Mitigation Experience Toolkit 5.5 certificate pinning rules to ensure proper access to Azure SSL/TLS sites.
Don't share accounts and passwords between administrators, or reuse passwords across multiple user accounts or services, particularly those for social media or other nonadministrative activities.	Create a dedicated Microsoft account to manage your Azure subscription—an account that is not used for personal email.

DON'T	DO
Don't email configuration files.	Configuration files and profiles should be installed from a trusted source (for example, an encrypted USB flash drive), not from a mechanism that can be easily compromised, such as email.
Don't use weak or simple logon passwords.	Enforce strong password policies, expiration cycles (change-on-first-use), console timeouts, and automatic account lockouts. Use a client password management system with multi-factor authentication for password vault access.
Don't expose management ports to the Internet.	Lock down Azure ports and IP addresses to restrict management access. For more information, see the Azure Network Security white paper.
-	Use firewalls, VPNs, and NAP for all management connections.

Azure operations

Within Microsoft's operation of Azure, operations engineers and support personnel who access Azure's production systems use [hardened workstation PCs with VMs](#) provisioned on them for internal corporate network access and applications (such as e-mail, intranet, etc.). All management workstation computers have TPMs, the host boot drive is encrypted with BitLocker, and they are joined to a special organizational unit (OU) in Microsoft's primary corporate domain.

System hardening is enforced through Group Policy, with centralized software updating. For auditing and analysis, event logs (such as security and AppLocker) are collected from management workstations and saved to a central location.

In addition, dedicated jump-boxes on Microsoft's network that require two-factor authentication are used to connect to Azure's production network.

Azure security checklist

Minimizing the number of tasks that administrators can perform on a hardened workstation helps minimize the attack surface in your development and management environment. Use the following technologies to help protect your hardened workstation:

- IE hardening. The Internet Explorer browser (or any web browser, for that matter) is a key entry point for harmful code due to its extensive interactions with external servers. Review your client policies and enforce running in protected mode, disabling add-ons, disabling file downloads, and using [Microsoft SmartScreen](#) filtering. Ensure that security warnings are displayed. Take advantage of Internet zones and create a list of trusted sites for which you have configured reasonable hardening. Block all other sites and in-browser code, such as ActiveX and Java.
- Standard user. Running as a standard user brings a number of benefits, the biggest of which is that stealing administrator credentials via malware becomes more difficult. In addition, a standard user account does not have elevated privileges on the root operating system, and many configuration options and APIs are locked out by default.
- AppLocker. You can use [AppLocker](#) to restrict the programs and scripts that users can run. You can run AppLocker in audit or enforcement mode. By default, AppLocker has an allow rule that enables users who have an admin token to run all code on the client. This rule exists to prevent administrators from locking themselves out, and it applies only to elevated tokens. See also [Code Integrity](#) as part of Windows Server [core security](#).
- Code signing. Code signing all tools and scripts used by administrators provides a manageable mechanism for

deploying application lockdown policies. Hashes do not scale with rapid changes to the code, and file paths do not provide a high level of security. You should combine AppLocker rules with a PowerShell [execution policy](#) that only allows specific signed code and scripts to be [executed](#).

- Group Policy. Create a global administrative policy that is applied to any domain workstation that is used for management (and block access from all others), and to user accounts authenticated on those workstations.
- Security-enhanced provisioning. Safeguard your baseline hardened workstation image to help protect against tampering. Use security measures like encryption and isolation to store images, virtual machines, and scripts, and restrict access (perhaps use an auditable check-in/check-out process).
- Patching. Maintain a consistent build (or have separate images for development, operations, and other administrative tasks), scan for changes and malware routinely, keep the build up to date, and only activate machines when they are needed.
- Encryption. Make sure that management workstations have a TPM to more securely enable [Encrypting File System](#) (EFS) and BitLocker. If you are using Windows To Go, use only encrypted USB keys together with BitLocker.
- Governance. Use AD DS GPOs to control all the administrators' Windows interfaces, such as file sharing. Include management workstations in auditing, monitoring, and logging processes. Track all administrator and developer access and usage.

Summary

Using a hardened workstation configuration for administering your Azure cloud services, Virtual Machines, and applications can help you avoid numerous risks and threats that can come from remotely managing critical IT infrastructure. Both Azure and Windows provide mechanisms that you can employ to help protect and control communications, authentication, and client behavior.

Next steps

The following resources are available to provide more general information about Azure and related Microsoft services, in addition to specific items referenced in this paper:

- [Securing Privileged Access](#) – get the technical details for designing and building a secure administrative workstation for Azure management
- [Microsoft Trust Center](#) - learn about Azure platform capabilities that protect the Azure fabric and the workloads that run on Azure
- [Microsoft Security Response Center](#) -- where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com
- [Azure Security Blog](#) – keep up to date on the latest in Azure Security

Introduction to Azure Security Center

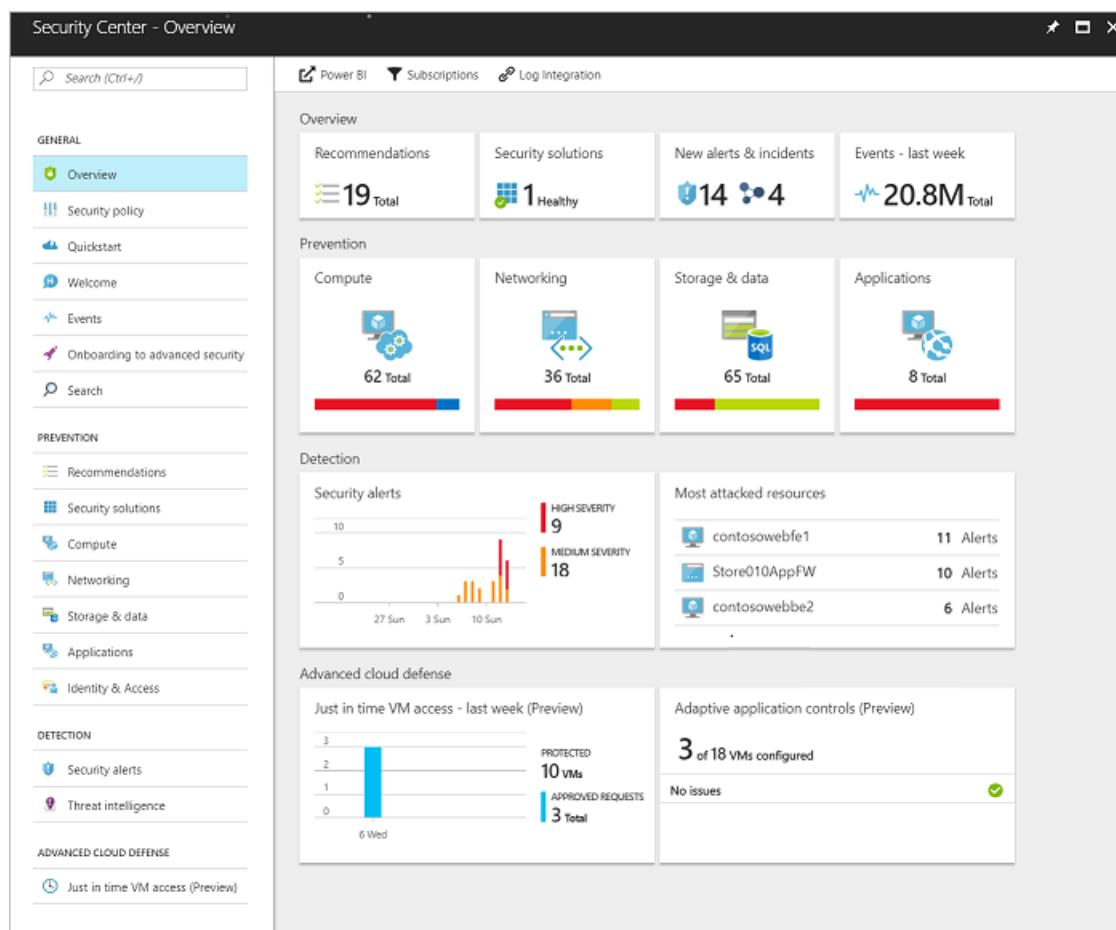
9/13/2017 • 3 min to read • [Edit Online](#)

Learn about Azure Security Center, its key capabilities, and how to get started.

What is Azure Security Center?

Azure Security Center provides unified security management and advanced threat protection for workloads running in Azure, on-premises, and in other clouds. It delivers visibility and control over hybrid cloud workloads, active defenses that reduce your exposure to threats, and intelligent detection to help you keep pace with rapidly evolving cyber attacks.

The Security Center **Overview** provides a quick view into the security posture of your Azure and non-Azure workloads, enabling you to discover and assess the security of your workloads and to identify and mitigate risk.



The screenshot shows the Azure Security Center - Overview page. The left sidebar contains navigation links for General, Prevention, Detection, and Advanced Cloud Defense. The main area has sections for Overview, Prevention, Detection, and Advanced cloud defense, each with various metrics and charts.

- Overview:** Includes recommendations (19 Total), security solutions (1 Healthy), new alerts & incidents (14 Total), and events from the last week (20.8M Total).
- Prevention:** Shows counts for Compute (62 Total), Networking (36 Total), Storage & data (65 Total), and Applications (8 Total).
- Detection:** A chart for security alerts shows counts for High Severity (9) and Medium Severity (18) over time periods 27 Sun, 3 Sun, and 10 Sun.
- Advanced cloud defense:** Shows Just in time VM access - last week (Preview) with 3 PROTECTED VMs and 3 APPROVED REQUESTS, and Adaptive application controls (Preview) with 3 of 18 VMs configured and No issues.

Why use Security Center?

Unified security management

- Reduced management complexity.** Manage security across all your hybrid cloud workloads – on-premises, Azure, and other cloud platforms – in one console. Built-in dashboards provide instant insights into security issues that require attention.
- Centralized policy management.** Ensure compliance with company or regulatory security requirements by centrally managing security policies across all your hybrid cloud workloads.
- Security data from many sources.** Collect, search, and analyze security data from a variety of sources,

including connected partner solutions like network firewalls and other Microsoft services.

- **Integration with existing security workflows.** Access, integrate, and analyze security information using REST APIs to connect existing tools and processes.
- **Compliance reporting.** Use security data and insights to demonstrate compliance and easily generate evidence for auditors.

Multi-layer cyber defense

- **Continuous security assessment.** Monitor the security of machines, networks, and Azure services using hundreds of built-in security assessments or create your own. Identify software and configurations that are vulnerable to attack.
- **Actionable recommendations.** Remediate security vulnerabilities before they can be exploited by attackers with prioritized, actionable security recommendations and built-in automation playbooks.
- **Adaptive application controls.** Block malware and other unwanted applications by applying whitelisting recommendations adapted to your specific Azure workloads and powered by machine learning.
- **Network access security.** Reduce the network attack surface with just-in-time, controlled access to management ports on Azure VMs, drastically reducing exposure to brute force and other network attacks.

Intelligent threat detection and response

- **Industry's most extensive threat intelligence.** Tap into the Microsoft Intelligent Security Graph, which uses trillions of signals from Microsoft services and systems around the globe to identify new and evolving threats.
- **Advanced threat detection.** Use built-in behavioral analytics and machine learning to identify attacks and zero-day exploits. Monitor networks, machines, and cloud services for incoming attacks and post-breach activity.
- **Alerts and Incidents.** Focus on the most critical threats first with prioritized security alerts and incidents that map alerts of different types into a single attack campaign. Create your own custom security alerts as well.
- **Streamlined investigation.** Quickly assess the scope and impact of an attack with a visual, interactive experience. Use predefined or ad hoc queries for deeper exploration of security data.
- **Contextual threat intelligence.** Visualize the source of attacks on an interactive world map. Use built-in threat intelligence reports to gain valuable insight into the techniques and objectives of known malicious actors.

Get started

To get started with Security Center, you need a subscription to Microsoft Azure. Security Center is enabled with your Azure subscription. If you do not have a subscription, you can sign up for a [free trial](#).

You access Security Center from the [Azure portal](#). See the [portal documentation](#) to learn more.

[Getting started with Azure Security Center](#) quickly guides you through the security-monitoring and policy-management components of Security Center.

Next steps

In this document, you were introduced to Security Center, its key capabilities, and how to get started. To learn more, see the following resources:

- [Planning and operations guide](#) - Learn how to optimize your use of Security Center based on your organization's security requirements and cloud management model.
- [Setting security policies](#) — Learn how to configure security policies for your Azure subscriptions and resource groups.
- [Managing security recommendations](#) — Learn how recommendations help you protect your Azure resources.
- [Security health monitoring](#) — Learn how to monitor the health of your Azure resources.
- [Managing and responding to security alerts](#) — Learn how to manage and respond to security alerts.
- [Monitoring and processing security events](#) - Learn how to monitor and process security events collected over

time.

- [Monitoring partner solutions](#) — Learn how to monitor the health status of your partner solutions.
- [Azure Security Center FAQ](#) — Find frequently asked questions about using the service.
- [Azure Security blog](#) — Get the latest Azure security news and information.

Introduction to Microsoft Azure log integration

8/10/2017 • 3 min to read • [Edit Online](#)

Learn about Azure log integration, its key capabilities, and how it works.

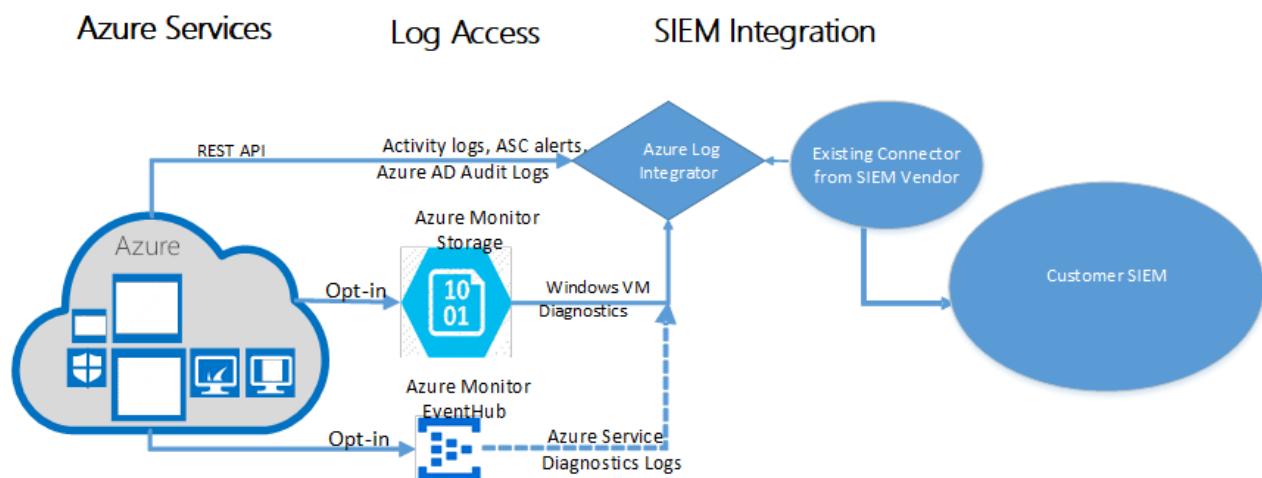
Overview

Azure log integration is a free solution that enables you to integrate raw logs from your Azure resources in to your on-premises Security Information and Event Management (SIEM) systems.

Azure log integration collects Windows events from Windows Event Viewer logs, [Azure Activity Logs](#), [Azure Security Center alerts](#), and [Azure Diagnostic logs](#) from Azure resources. This integration helps your SIEM solution provide a unified dashboard for all your assets, on-premises or in the cloud, so that you can aggregate, correlate, analyze, and alert for security events.

NOTE

At this time, the only supported clouds are Azure commercial and Azure Government. Other clouds are not supported.



What logs can I integrate?

Azure produces extensive logging for every Azure service. These logs represent three types of logs:

- **Control/management logs** provide visibility into the [Azure Resource Manager](#) CREATE, UPDATE, and DELETE operations. Azure Activity Logs is an example of this type of log.
- **Data plane logs** provide visibility into the events raised as part of the usage of an Azure resource. An example of this type of log is the Windows Event Viewer's **System**, **Security**, and **Application** channels in a Windows virtual machine. Another example is Diagnostics Logging configured through Azure Monitor
- **Processed events** provide analyzed event and alert information processed on your behalf. An example of this type of event is Azure Security Center Alerts, where Azure Security Center has processed and analyzed your subscription to provide alerts relevant to your current security posture.

Azure Log Integration supports ArcSight, QRadar, and Splunk. In all circumstances, please check with your SIEM vendor to assess whether they have a native connector. In some cases, you will not need to use Azure Log Integration when native connectors are available. For additional information on supported log types please visit the FAQ.

NOTE

While Azure Log Integration is a free solution, there are Azure storage costs resulting from the log file information storage.

Community assistance is available through the [Azure Log Integration MSDN Forum](#). The forum provides the AzLog community the ability to support each other with questions, answers, tips, and tricks on how to get the most out of Azure Log Integration. In addition, the Azure Log Integration team monitors this forum and will help whenever we can.

You can also open a [support request](#). To do this, select **Log Integration** as the service for which you are requesting support.

Next steps

In this document, you were introduced to Azure log integration. To learn more about Azure log integration and the types of logs supported, see the following:

- [Microsoft Azure Log Integration](#) – Download Center for details, system requirements, and install instructions on Azure log integration.
- [Get started with Azure log integration](#) - This tutorial walks you through installation of Azure log integration and integrating logs from Azure WAD storage, Azure Activity Logs, Azure Security Center alerts and Azure Active Directory audit logs.
- [Partner configuration steps](#) – This blog post shows you how to configure Azure log integration to work with partner solutions Splunk, HP ArcSight, and IBM QRadar. This blog represents our current position on configuring the partner solutions. In all cases, please refer to partner solution documentation first.
- [Activity and ASC alerts over syslog to QRadar](#) – This blog post provides the steps to send Activity and Azure Security Center alerts over syslog to QRadar
- [Azure log Integration frequently asked questions \(FAQ\)](#) - This FAQ answers questions about Azure log integration.
- [Integrating Security Center alerts with Azure log Integration](#) – This document shows you how to sync Azure Security Center alerts with Azure Log Integration.

Azure log integration with Azure Diagnostics Logging and Windows Event Forwarding

7/31/2017 • 10 min to read • [Edit Online](#)

Azure Log Integration (AzLog) enables you to integrate raw logs from your Azure resources into your on-premises Security Information and Event Management (SIEM) systems. This integration makes it possible to have a unified security dashboard for all your assets, on-premises or in the cloud, so that you can aggregate, correlate, analyze, and alert for security events associated with your applications.

NOTE

For more information on Azure Log Integration, you can review the [Azure Log Integration overview](#).

This article will help you get started with Azure Log Integration by focusing on the installation of the Azlog service and integrating the service with Azure Diagnostics. The Azure Log Integration service will then be able to collect Windows Event Log information from the Windows Security Event Channel from virtual machines deployed in Azure IaaS. This is very similar to "Event Forwarding" that you may have used on-premises.

NOTE

The ability to bring the output of Azure log integration in to the SIEM is provided by the SIEM itself. Please see the article [Integrating Azure Log Integration with your On-premises SIEM](#) for more information.

To be very clear - the Azure Log Integration service runs on a physical or virtual computer that is using the Windows Server 2008 R2 or above operating system (Windows Server 2012 R2 or Windows Server 2016 are preferred).

The physical computer can run on-premises (or on a hoster site). If you choose to run the Azure Log Integration service on a virtual machine, that virtual machine can be located on-premises or in a public cloud, such as Microsoft Azure.

The physical or virtual machine running the Azure Log Integration service requires network connectivity to the Azure public cloud. Steps in this article provide details on the configuration.

Prerequisites

At a minimum, the installation of AzLog requires the following items:

- An **Azure subscription**. If you do not have one, you can sign up for a [free account](#).
- A **storage account** that can be used for Windows Azure diagnostic logging (you can use a pre-configured storage account, or create a new one – will we demonstrate how to configure the storage account later in this article) >[!NOTE] Depending on your scenario a storage account may not be required. For the Azure diagnostics scenario covered in this article one is needed.
- **Two systems:** a machine that will run the Azure Log Integration service, and a machine that will be monitored and have its logging information sent to the Azlog service machine.
 - A machine you want to monitor – this is a VM running as an [Azure Virtual Machine](#)
 - A machine that will run the Azure log integration service; this machine will collect all the log information that will later be imported into your SIEM.
 - This system can be on-premises or in Microsoft Azure.

- This system can be on premises or in Microsoft Azure.
- o It needs to be running an x64 version of Windows server 2008 R2 SP1 or higher and have .NET 4.5.1 installed. You can determine the .NET version installed by following the article titled [How to: Determine Which .NET Framework Versions Are Installed](#)
- It must have connectivity to the Azure storage account used for Azure diagnostic logging. We will provide instructions later in this article on how you can confirm this connectivity

For a quick demonstration of the process of creating a virtual machine using the Azure portal take a look at the video below.

Deployment considerations

While you are testing Azure Log Integration, you can use any system that meets the minimum operating system requirements. However, for a production environment the load may require you to plan for scaling up or out.

You can run multiple instances of the Azure Log Integration service (one instance per physical or virtual machine) if event volume is high. In addition, you can load balance Azure Diagnostics storage accounts for Windows (WAD) and the number of subscriptions to provide to the instances should be based on your capacity.

NOTE

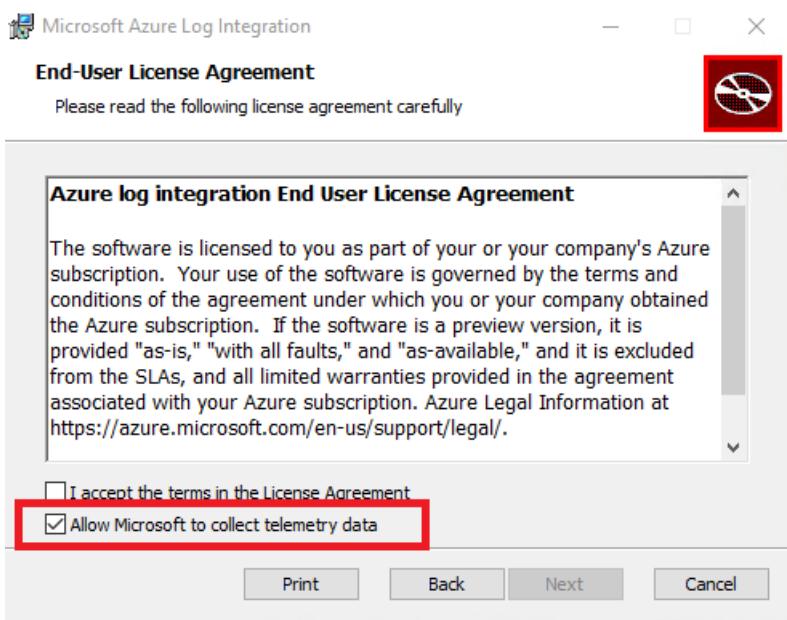
At this time we do not have specific recommendations for when to scale out instances of Azure log integration machines (i.e., machines that are running the Azure log integration service), or for storage accounts or subscriptions. Scaling decisions should be based on your performance observations in each of these areas.

You also have the option to scale up the Azure Log Integration service to help improve performance. The following performance metrics can help you in sizing the machines that you choose to run the Azure log integration service:

- On an 8-processor (core) machine, a single instance of Azlog Integrator can process about 24 million events per day (~1M/hour).
- On a 4-processor (core) machine, a single instance of Azlog Integrator can process about 1.5 million events per day (~62.5K/hour).

Install Azure log integration

To install Azure Log Integration, you need to download the [Azure log integration](#) installation file. Run through the setup routine and decide if you want to provide telemetry information to Microsoft.



NOTE

We recommend that you allow Microsoft to collect telemetry data. You can turn off collection of telemetry data by unchecking this option.

The Azure Log Integration service collects telemetry data from the machine on which it is installed.

Telemetry data collected is:

- Exceptions that occur during execution of Azure log integration
- Metrics about the number of queries and events processed
- Statistics about which Azlog.exe command-line options are being used

The installation process is covered in the video below.

Post installation and validation steps

After completing the basic setup routine, you're ready step to perform post installation and validation steps:

1. Open an elevated PowerShell window and navigate to **c:\Program Files\Microsoft Azure Log Integration**
2. The first step you need to take is to get the AzLog Cmdlets imported. You can do that by running the script **LoadAzlogModule.ps1** (notice the "." in the following command). Type **.\\LoadAzlogModule.ps1** and press **ENTER**.

You should see something like what appears in the figure below.

```
PS C:\Program Files\Microsoft Azure Log Integration> .\LoadAzLogModule.ps1
List of AzLog Cmdlets. To View again run "Get-Command -Module AzLogDll"

 CommandType Name          Version      Source
----- ----          -----      -----
 Cmdlet   Add-AzLogEventDestination 1.0.6269.5008 AzLogDll
 Cmdlet   Add-AzLogEventRoute       1.0.6269.5008 AzLogDll
 Cmdlet   Add-AzLogEventSource     1.0.6269.5008 AzLogDll
 Cmdlet   Get-AzLogEventDestination 1.0.6269.5008 AzLogDll
 Cmdlet   Get-AzLogEventRoute       1.0.6269.5008 AzLogDll
 Cmdlet   Get-AzLogEventSource     1.0.6269.5008 AzLogDll
 Cmdlet   Remove-AzLogEventDestination 1.0.6269.5008 AzLogDll
 Cmdlet   Remove-AzLogEventRoute     1.0.6269.5008 AzLogDll
 Cmdlet   Remove-AzLogEventSource   1.0.6269.5008 AzLogDll
 Cmdlet   Set-AzLogAzureEnvironment 1.0.6269.5008 AzLogDll
```

3. Now you need to configure AzLog to use a specific Azure environment. An “Azure environment” is the “type” of Azure cloud data center you want to work with. While there are several Azure environments at this time, the currently relevant options are either **AzureCloud** or **AzureUSGovernment**. In your elevated PowerShell environment, make sure that you are in **c:\program files\Microsoft Azure Log Integration**. Once there, run the command:

```
Set-AzlogAzureEnvironment -Name AzureCloud (for Azure commercial)
```

NOTE

When the command succeeds, you will not receive any feedback. If you want to use the US Government Azure cloud, you would use **AzureUSGovernment** (for the -Name variable) for the USA government cloud. Other Azure clouds are not supported at this time.

4. Before you can monitor a system you will need the name of the storage account in use for Azure Diagnostics. In the Azure portal navigate to **Virtual machines** and look for the virtual machine that you will monitor. In the **Properties** section, choose **Diagnostic Settings**. Click on **Agent** and make note of the storage account name specified. You will need this account name for a later step.

[Overview](#) [Performance counters](#) [Logs](#) [Crash dumps](#) [Sinks](#) [Agent](#) [Boot diagnostics](#)

Configure additional options for the Azure Diagnostics agent.

* Storage account ⓘ
vrm01diag967 >

Disk quota (MB): ⓘ

5120 ✓

Diagnostic infrastructure logs: ⓘ

Disabled Enabled

Log level: ⓘ

Error ▾



Azure Monitoring collects host-level metrics – like CPU utilization, disk and network usage – for all virtual machines without any additional software. For more insight into this virtual machine, you can collect guest-level metrics, logs, and other diagnostic data using the Azure Diagnostics agent. You can also send diagnostic data to other services like Application Insights. [Learn more](#)

To get started now, click the button below:

[Enable guest-level monitoring](#)

Already know what you're doing? You can customize the diagnostic data you want to collect by visiting each of the tabs above. You can add or remove data types to collect at any time.

NOTE

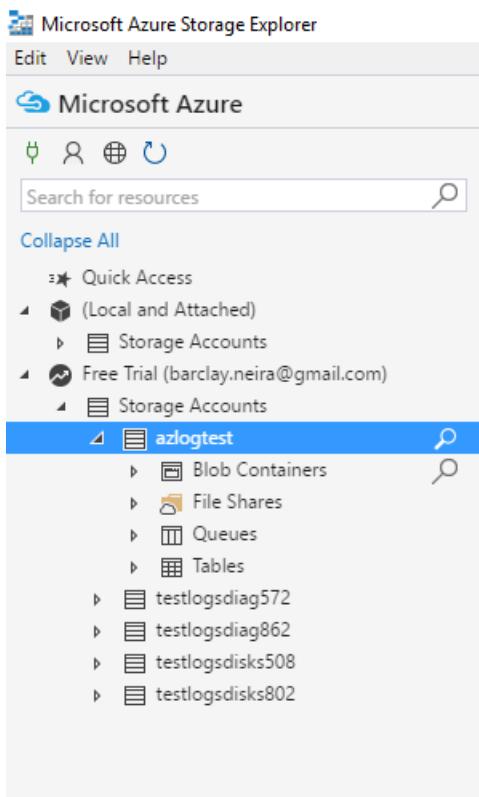
If Monitoring was not enabled during virtual machine creation you will be given the option to enable it as shown above.

5. Now we'll switch our attention back to the Azure log integration machine. We need to verify that you have connectivity to the Storage Account from the system where you installed Azure Log Integration. The physical computer or virtual machine running the Azure Log Integration service needs access to the storage account to retrieve information logged by Azure Diagnostics as configured on each of the monitored systems.
 - a. You can download Azure Storage Explorer [here](#).
 - b. Run through the setup routine
 - c. Once the installation completes click **Next** and leave the check box **Launch Microsoft Azure Storage Explorer** checked.
 - d. Log in to Azure.
 - e. Verify that you can see the storage account that you configured for Azure Diagnostics.

```

        ▲ Microsoft Azure Internal Consumption (tomsh@microsoft.com)
          ▲ Storage Accounts
            ▶ ad2041centralus
            ▶ ad2041southcentralus
            ▲ alirgdiag523
              ▶ Blob Containers
              ▶ File Shares
              ▶ Queues
              ▲ Tables
                ▶ $MetricsCapacityBlob
                ▶ $MetricsHourPrimaryTransactionsBlob
                ▶ $MetricsHourPrimaryTransactionsQueue
                ▶ $MetricsHourPrimaryTransactionsTable
                ▶ SchemasTable
                ▶ WADWindowsEventLogsTable
    
```

- f. Notice that there are a few options under storage accounts. One of them is **Tables**. Under **Tables** you should see one called **WADWindowsEventLogsTable**.



Integrate Azure Diagnostic logging

In this step, you will configure the machine running the Azure Log Integration service to connect to the storage account that contains the log files. To complete this step we will need a few things up front.

- **FriendlyNameForSource:** This is a friendly name that you can apply to the storage account that you've configured the virtual machine to store information from Azure Diagnostics
- **StorageAccountName:** This is the name of the storage account that you specified when you configured Azure diagnostics.
- **StorageKey:** This is the storage key for the storage account where the Azure Diagnostics information is stored for this virtual machine.

Perform the following steps to obtain the storage key:

1. Browse to the [Azure portal](#).
2. In the navigation pane of the Azure console, scroll to the bottom and click **More services**.

Microsoft Azure



New

SQL databases

Virtual machines (classic)

Virtual machines

Cloud services (classic)

Subscriptions

Application Insights

Virtual networks (classic)

Network interfaces

Public IP addresses

Azure Active Directory

Monitor

Billing

Help + support

Advisor

More services >

3. Enter **Storage** in the **Filter** text box. Click **Storage accounts** (this will appear after you enter **Storage**)

Shift+Space to toggle favorites

| Filter

By category ▾

GENERAL

-  Dashboard ★
-  Resource groups ★
-  All resources ★
-  Subscriptions ★
-  Billing PREVIEW ★
-  Help + support ★

COMPUTE

-  Virtual machines ★
-  Virtual machines (classic) ★
-  Virtual machine scale sets ★
-  Container services ★
-  Batch accounts ★

4. A list of storage accounts will appear, double click on the account that you assigned to Log storage.

Storage accounts

Microsoft

 Add  Columns  Refresh

Subscriptions: 1 of 6 selected – Don't see a subscription? [Switch directories](#)

Microsoft Azure Internal C

5 items

NAME ▾

 ad2041centralus

 ad2041southcentralus

 alirgdiag523

 alirgdisks684

 ascrgjuly20164543

5. Click on **Access keys** in the **Settings** section.

 alirgdiag523
Storage account

 Search (Ctrl+ /)

-  Overview
 -  Activity log
 -  Access control (IAM)
 -  Tags
 -  Diagnose and solve problems
-
- SETTINGS**
-  Access keys
 -  Configuration
 -  Shared access signature
 -  Properties
 -  Locks
 -  Automation script
-
- BLOB SERVICE**
-  Containers
 -  CORS

6. Copy **key1** and put it in a secure location that you can access for the next step.

The screenshot shows the Azure Storage account settings for 'alirgdiag523'. The left sidebar lists various options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, SETTINGS, BLOB SERVICE, Containers, and CORS. Under SETTINGS, 'Access keys' is highlighted with a red box. Other options include Configuration, Shared access signature, Properties, Locks, and Automation script. The BLOB SERVICE section contains Container and CORS settings.

7. On the server that you installed Azure Log Integration, open an elevated Command Prompt (note that we're using an elevated Command Prompt window here, not an elevated PowerShell console).
8. Navigate to **c:\Program Files\Microsoft Azure Log Integration**
9. Run `Azlog source add <FriendlyNameForTheSource> WAD <StorageAccountName> <StorageKey>`

For example

```
Azlog source add Azlogtest WAD Azlog9414
fxxxFxxxxxxxxywoEJK2xxxxxxxxxxxxJ+xVJx6m/X5SQDYc4Wpjpli9S9Mm+vXS2RVYtp1mes0t9H5cuqXEw==
```

If you would like the subscription ID to show up in the event XML, append the subscription ID to the friendly name: `Azlog source add <FriendlyNameForTheSource>.<SubscriptionID> WAD <StorageAccountName> <StorageKey>` or for example,

```
Azlog source add Azlogtest.YourSubscriptionID WAD Azlog9414
fxxxFxxxxxxxxywoEJK2xxxxxxxxxxxxJ+xVJx6m/X5SQDYc4Wpjpli9S9Mm+vXS2RVYtp1mes0t9H5cuqXEw==
```

NOTE

Wait up to 60 minutes, then view the events that are pulled from the storage account. To view, open **Event Viewer > Windows Logs > Forwarded Events** on the Azlog Integrator.

Here you can see a video going over the steps covered above.

What if data is not showing up in the Forwarded Events folder?

If after an hour data is not showing up in the **Forwarded Events** folder, then:

1. Check the machine running the Azure Log Integration service and confirm that it can access Azure. To test connectivity, try to open the [Azure portal](#) from the browser.
2. Make sure the user account **Azlog** has write permission on the folder **users\Azlog**.
 - a. Open **Windows Explorer**
 - b. Navigate to **c:\users**
 - c. Right click on **c:\users\Azlog**
 - d. Click on **Security**
 - e. Click on **NT Service\Azlog** and check the permissions for the account. If the account is missing from this tab or if the appropriate permissions are not currently showing you can grant the account rights in this tab.
3. Make sure the storage account added in the command **Azlog source add** is listed when you run the command **Azlog source list**.
4. Go to **Event Viewer > Windows Logs > Application** to see if there are any errors reported from the Azure log integration.

If you run into any issues during the installation and configuration, please open a [support request](#), select **Log Integration** as the service for which you are requesting support.

Another support option is the [Azure Log Integration MSDN Forum](#). Here the community can support each other with questions, answers, tips, and tricks on how to get the most out of Azure Log Integration. In addition, the Azure Log Integration team monitors this forum and will help whenever we can.

Next steps

To learn more about Azure Log Integration, see the following documents:

- [Microsoft Azure Log Integration for Azure logs](#) – Download Center for details, system requirements, and install instructions on Azure log integration.
- [Introduction to Azure log integration](#) – This document introduces you to Azure log integration, its key capabilities, and how it works.
- [Partner configuration steps](#) – This blog post shows you how to configure Azure log integration to work with partner solutions Splunk, HP ArcSight, and IBM QRadar. This is our current guidance on how to configure the SIEM components. Please check with your SIEM vendor first for additional details.
- [Azure log Integration frequently asked questions \(FAQ\)](#) - This FAQ answers questions about Azure log integration.
- [Integrating Security Center alerts with Azure log Integration](#) – This document shows you how to sync Security Center alerts, along with virtual machine security events collected by Azure Diagnostics and Azure Activity Logs, with your log analytics or SIEM solution.
- [New features for Azure diagnostics and Azure Audit Logs](#) – This blog post introduces you to Azure Audit Logs and other features that help you gain insights into the operations of your Azure resources.

Integrate Azure Active Directory audit logs

8/22/2017 • 2 min to read • [Edit Online](#)

Azure Active Directory (Azure AD) audit events help you identify privileged actions that occurred in Azure Active Directory. You can see the types of events that you can track by reviewing [Azure Active Directory audit report events](#).

NOTE

Before you attempt the steps in this article, you must review the [Get started](#) article and complete the steps there.

Steps to integrate Azure Active directory audit logs

1. Open the command prompt and run this command:

```
cd c:\Program Files\Microsoft Azure Log Integration
```

2. Run this command:

```
azlog createazureid
```

This command prompts you for your Azure login. The command then creates an Azure Active Directory service principal in the Azure AD tenants that host the Azure subscriptions in which the logged-in user is an administrator, a co-administrator, or an owner. The command will fail if the logged-in user is only a guest user in the Azure AD tenant. Authentication to Azure is done through Azure AD. Creating a service principal for Azure Log Integration creates the Azure AD identity that is given access to read from Azure subscriptions.

3. Run the following command to provide your tenant ID. You need to be member of the tenant admin role to run the command.

```
Azlog.exe authorizedirectoryreader tenantId
```

Example:

```
AZLOG.exe authorizedirectoryreader ba2c0000-d24b-4f4e-92b1-48c4469999
```

4. Check the following folders to confirm that the Azure Active Directory audit log JSON files are created in them:

- **C:\Users\azlog\AzureActiveDirectoryJson**
- **C:\Users\azlog\AzureActiveDirectoryJsonLD**

The following video demonstrates the steps covered in this article:

NOTE

For specific instructions on bringing the information in the JSON files into your security information and event management (SIEM) system, contact your SIEM vendor.

Community assistance is available through the [Azure Log Integration MSDN Forum](#). This forum enables people in the Azure Log Integration community to support each other with questions, answers, tips, and tricks. In addition, the Azure Log Integration team monitors this forum and helps whenever it can.

You can also open a [support request](#). Select **Log Integration** as the service for which you are requesting support.

Next steps

To learn more about Azure Log Integration, see:

- [Microsoft Azure Log Integration for Azure logs](#): This Download Center page gives details, system requirements, and installation instructions for Azure Log Integration.
- [Introduction to Azure Log Integration](#): This article introduces you to Azure Log Integration, its key capabilities, and how it works.
- [Partner configuration steps](#): This blog post shows you how to configure Azure Log Integration to work with partner solutions Splunk, HP ArcSight, and IBM QRadar.
- [Azure Log Integration FAQ](#): This article answers questions about Azure Log Integration.
- [Integrating Security Center alerts with Azure Log Integration](#): This article shows you how to sync Security Center alerts, along with virtual machine security events collected by Azure Diagnostics and Azure audit logs, with your log analytics or SIEM solution.
- [New features for Azure Diagnostics and Azure audit logs](#): This blog post introduces you to Azure audit logs and other features that help you gain insights into the operations of your Azure resources.

How to get your Security Center alerts in Azure log integration

8/30/2017 • 2 min to read • [Edit Online](#)

This article provides the steps required to enable the Azure Log Integration service to pull Security Alert information generated by Azure Security Center. You must have successfully completed the steps in the [Get started](#) article before performing the steps in this article.

Detailed steps

The following steps will create the required Azure Active Directory Service Principal and assign the Service Principle read permissions to the subscription:

1. Open the command prompt and navigate to **c:\Program Files\Microsoft Azure Log Integration**
2. Run the command `azlog createazureid`

This command prompts you for your Azure login. The command then creates an [Azure Active Directory Service Principal](#) in the Azure AD Tenants that host the Azure subscriptions in which the logged in user is an Administrator, a Co-Administrator, or an Owner. The command will fail if the logged in user is only a Guest user in the Azure AD Tenant. Authentication to Azure is done through Azure Active Directory (AD). Creating a service principal for Azlog Integration creates the Azure AD identity that will be given access to read from Azure subscriptions.

3. Next you will run a command that assigns reader access on the subscription to the service principal you just created. If you don't specify a SubscriptionID, then the command will attempt to assign the service principal reader role to all subscriptions to which you have any access.

```
azlog authorize <SubscriptionID>
```

for example

```
azlog authorize 0ee55555-0bc4-4a32-a4e8-c29980000000
```

NOTE

You may see warnings if you run the authorize command immediately after the `createazureid` command. There is some latency between when the Azure AD account is created and when the account is available for use. If you wait about 60 seconds after running the `createazureid` command to run the authorize command, then you should not see these warnings.

4. Check the following folders to confirm that the Audit log JSON files are there:

- **c:\Users\azlog\AzureResourceManagerJson**
- **c:\Users\azlog\AzureResourceManagerJsonLD**

5. Check the following folders to confirm that Security Center alerts exist in them:

- **c:\Users\azlog\AzureSecurityCenterJson**
- **c:\Users\azlog\AzureSecurityCenterJsonLD**

If you run into any issues during the installation and configuration, please open a [support request](#), select **Log Integration** as the service for which you are requesting support.

Next steps

To learn more about Azure Log Integration, see the following documents:

- [Microsoft Azure Log Integration for Azure logs](#) – Download Center for details, system requirements, and install instructions on Azure log integration.
- [Introduction to Azure log integration](#) – This document introduces you to Azure log integration, its key capabilities, and how it works.
- [Partner configuration steps](#) – This blog post shows you how to configure Azure log integration to work with partner solutions Splunk, HP ArcSight, and IBM QRadar.
- [Azure log Integration frequently asked questions \(FAQ\)](#) - This FAQ answers questions about Azure log integration.
- [New features for Azure diagnostics and Azure Audit Logs](#) – This blog post introduces you to Azure Audit Logs and other features that help you gain insights into the operations of your Azure resources.

Azure Log Integration tutorial: Process Azure Key Vault events by using Event Hubs

8/22/2017 • 6 min to read • [Edit Online](#)

You can use Azure Log Integration to retrieve logged events and make them available to your security information and event management (SIEM) system. This tutorial shows an example of how Azure Log Integration can be used to process logs that are acquired through Azure Event Hubs.

Use this tutorial to get acquainted with how Azure Log Integration and Event Hubs work together by following the example steps and understanding how each step supports the solution. Then you can take what you've learned here to create your own steps to support your company's unique requirements.

WARNING

The steps and commands in this tutorial are not intended to be copied and pasted. They're examples only. Do not use the PowerShell commands "as is" in your live environment. You must customize them based on your unique environment.

This tutorial walks you through the process of taking Azure Key Vault activity logged to an event hub and making it available as JSON files to your SIEM system. You can then configure your SIEM system to process the JSON files.

NOTE

Most of the steps in this tutorial involve configuring key vaults, storage accounts, and event hubs. The specific Azure Log Integration steps are at the end of this tutorial. Do not perform these steps in a production environment. They are intended for a lab environment only. You must customize the steps before using them in production.

Information provided along the way helps you understand the reasons behind each step. Links to other articles give you more detail on certain topics.

For more information about the services that this tutorial mentions, see:

- [Azure Key Vault](#)
- [Azure Event Hubs](#)
- [Azure Log Integration](#)

Initial setup

Before you can complete the steps in this article, you need the following:

1. An Azure subscription and account on that subscription with administrator rights. If you don't have a subscription, you can create a [free account](#).
2. A system with access to the internet that meets the requirements for installing Azure Log Integration. The system can be on a cloud service or hosted on-premises.
3. [Azure Log Integration](#) installed. To install it:
 - a. Use Remote Desktop to connect to the system mentioned in step 2.
 - b. Copy the Azure Log Integration installer to the system. You can [download the installation files](#).
 - c. Start the installer and accept the Microsoft Software License Terms.
 - d. If you will provide telemetry information, leave the check box selected. If you'd rather not send usage

information to Microsoft, clear the check box.

For more information about Azure Log Integration and how to install it, see [Azure Log Integration with Azure Diagnostics logging and Windows Event Forwarding](#).

4. The latest PowerShell version.

If you have Windows Server 2016 installed, then you have at least PowerShell 5.0. If you're using any other version of Windows Server, you might have an earlier version of PowerShell installed. You can check the version by entering `get-host` in a PowerShell window. If you don't have PowerShell 5.0 installed, you can [download it](#).

After you have at least PowerShell 5.0, you can proceed to install the latest version:

- a. In a PowerShell window, enter the `Install-Module Azure` command. Complete the installation steps.
- b. Enter the `Install-Module AzureRM` command. Complete the installation steps.

For more information, see [Install Azure PowerShell](#).

Create supporting infrastructure elements

1. Open an elevated PowerShell window and go to **C:\Program Files\Microsoft Azure Log Integration**.
2. Import the AzLog cmdlets by running the script LoadAzLogModule.ps1. Enter the `.\LoadAzLogModule.ps1` command. (Notice the "\." in that command.) You should see something like this:

```
PS C:\Program Files\Microsoft Azure Log Integration> .\LoadAzLogModule.ps1
List of AzLog Cmdlets. To View again run "Get-Command -Module AzLogD1l"

 CommandType Name            Version      Source
----- ----
 Cmdlet     Add-AzLogEventDestination 1.0.6269.5008 AzLogD1l
 Cmdlet     Add-AzLogEventRoute       1.0.6269.5008 AzLogD1l
 Cmdlet     Add-AzLogEventSource     1.0.6269.5008 AzLogD1l
 Cmdlet     Get-AzLogEventDestination 1.0.6269.5008 AzLogD1l
 Cmdlet     Get-AzLogEventRoute       1.0.6269.5008 AzLogD1l
 Cmdlet     Get-AzLogEventSource     1.0.6269.5008 AzLogD1l
 Cmdlet     Remove-AzLogEventDestination 1.0.6269.5008 AzLogD1l
 Cmdlet     Remove-AzLogEventRoute       1.0.6269.5008 AzLogD1l
 Cmdlet     Remove-AzLogEventSource     1.0.6269.5008 AzLogD1l
 Cmdlet     Set-AzLogAzureEnvironment 1.0.6269.5008 AzLogD1l
```

3. Enter the `Login-AzureRmAccount` command. In the login window, enter the credential information for the subscription that you will use for this tutorial.

NOTE

If this is the first time that you're logging in to Azure from this machine, you will see a message about allowing Microsoft to collect PowerShell usage data. We recommend that you enable this data collection because it will be used to improve Azure PowerShell.

4. After successful authentication, you're logged in and you see the information in the following screenshot. Take note of the subscription ID and subscription name, because you'll need them to complete later steps.

The screenshot shows a Windows PowerShell window titled "Administrator: Windows PowerShell". The command PS C:\WINDOWS\system32> login-azurermaccount is run, followed by a warning about data collection. It then provides instructions for enabling or disabling data collection, including the Disable-AzureDataCollection and Enable-AzureDataCollection cmdlets. A "Select Y to enable data collection [Y/N]:" prompt is shown, with the user's input "Y" highlighted. The output concludes with a warning about the setting being saved to a specific JSON file. Below the command line, a portion of the PowerShell interface is visible, showing environment variables like Account, TenantId, SubscriptionId, SubscriptionName, and CurrentStorageAccount.

5. Create variables to store values that will be used later. Enter each of the following PowerShell lines. You might need to adjust the values to match your environment.

- \$subscriptionName = 'Visual Studio Ultimate with MSDN' (Your subscription name might be different. You can see it as part of the output of the previous command.)
- \$location = 'West US' (This variable will be used to pass the location where resources should be created. You can change this variable to be any location of your choosing.)
- \$random = Get-Random
- \$name = 'azlogtest' + \$random (The name can be anything, but it should include only lowercase letters and numbers.)
- \$storageName = \$name (This variable will be used for the storage account name.)
- \$rgname = \$name (This variable will be used for the resource group name.)
- \$eventHubNameSpaceName = \$name (This is the name of the event hub namespace.)

6. Specify the subscription that you will be working with:

```
Select-AzureRmSubscription -SubscriptionName $subscriptionName
```

7. Create a resource group:

```
$rg = New-AzureRmResourceGroup -Name $rgname -Location $location
```

If you enter \$rg at this point, you should see output similar to this screenshot:

The screenshot shows the creation of a new Azure Resource Group named "azlogtest76" in the "westus" location. The command \$rg = New-AzureRmResourceGroup -Name \$rgname -Location \$location is run, followed by the output showing the ResourceGroupName, Location, ProvisioningState, Tags, and ResourceId of the newly created group.

ResourceGroupName	:	azlogtest76
Location	:	westus
ProvisioningState	:	Succeeded
Tags	:	
ResourceId	:	/subscriptions/[REDACTED]

8. Create a storage account that will be used to keep track of state information:

```
$storage = New-AzureRmStorageAccount -ResourceGroupName $rgname -Name $storagename -Location $location -SkuName Standard_LRS
```

9. Create the event hub namespace. This is required to create an event hub.

```
$eventHubNameSpace = New-AzureRmEventHubNamespace -ResourceGroupName $rgname -NamespaceName $eventHubNamespaceName -Location $location
```

10. Get the rule ID that will be used with the insights provider:

```
$sbruleid = $eventHubNameSpace.Id +'/authorizationrules/RootManageSharedAccessKey'
```

11. Get all possible Azure locations and add the names to a variable that can be used in a later step:

a. `$locationObjects = Get-AzureRmLocation`
b. `$locations = @('global') + $locationobjects.location`

If you enter `$locations` at this point, you see the location names without the additional information returned by `Get-AzureRmLocation`.

12. Create an Azure Resource Manager log profile:

```
Add-AzureRmLogProfile -Name $name -ServiceBusRuleId $sbruleid -Locations $locations
```

For more information about the Azure log profile, see [Overview of the Azure Activity Log](#).

NOTE

You might get an error message when you try to create a log profile. You can then review the documentation for `Get-AzureRmLogProfile` and `Remove-AzureRmLogProfile`. If you run `Get-AzureRmLogProfile`, you see information about the log profile. You can delete the existing log profile by entering the `Remove-AzureRmLogProfile -name 'Log Profile Name'` command.

```
Add-AzureRmLogProfile : Exception type: CloudException, Message: The limit of 1 log profiles was reached. To create new log profile 'azlogtest2102253579', delete an existing one., Code: Conflict, Status code:Conflict, Reason phrase: Conflict
At line:1 char:1
+ Add-AzureRmLogProfile -Name $name -ServiceBusRuleId $sbruleid -Locati ...
+ ~~~~~
+ CategoryInfo          : CloseError: (:) [Add-AzureRmLogProfile], PSInvalidOperationException
+ FullyQualifiedErrorId : Microsoft.Azure.Commands.Insights.LogProfiles.AddAzureRmLogProfileCommand
```

Create a key vault

1. Create the key vault:

```
$kv = New-AzureRmKeyVault -VaultName $name -ResourceGroupName $rgname -Location $location
```

2. Configure logging for the key vault:

```
Set-AzureRmDiagnosticSetting -ResourceId $kv.ResourceId -ServiceBusRuleId $sbruleid -Enabled $true
```

Generate log activity

Requests need to be sent to Key Vault to generate log activity. Actions like key generation, storing secrets, or reading secrets from Key Vault will create log entries.

1. Display the current storage keys:

```
Get-AzureRmStorageAccountKey -Name $storagename -ResourceGroupName $rgname | ft -a
```

2. Generate a new **key2**:

```
New-AzureRmStorageAccountKey -Name $storagename -ResourceGroupName $rgname -KeyName key2
```

3. Display the keys again and see that **key2** holds a different value:

```
Get-AzureRmStorageAccountKey -Name $storagename -ResourceGroupName $rgname | ft -a
```

4. Set and read a secret to generate additional log entries:

a.

```
Set-AzureKeyVaultSecret -VaultName $name -Name TestSecret -SecretValue (ConvertTo-SecureString -String 'Hi There!' -AsPlainText -Force)
```

b. `(Get-AzureKeyVaultSecret -VaultName $name -Name TestSecret).SecretValueText`

```
PS C:\Program Files\Microsoft Azure Log Integration> Set-AzureKeyVaultSecret -VaultName $name -Name TestSecret -SecretValue (ConvertTo-SecureString -String 'Hi There!' -AsPlainText -Force)

Vault Name   : azlogtest76
Name         : TestSecret
Version      : c43159516c2445c485361638fb6d6c16
Id           : https://azlogtest76.vault.azure.net:443/secrets/TestSecret/c43159516c2445c485361638fb6d6c16
Enabled       : True
Expires      :
Not Before   :
Created      : 5/22/2017 8:36:35 PM
Updated      : 5/22/2017 8:36:35 PM
Content Type :
Tags         :
```

Configure Azure Log Integration

Now that you have configured all the required elements to have Key Vault logging to an event hub, you need to configure Azure Log Integration:

1. `$storage = Get-AzureRmStorageAccount -ResourceGroupName $rgname -Name $storagename`
2. `$eventHubKey = Get-AzureRmEventHubNamespaceKey -ResourceGroupName $rgname -NamespaceName $eventHubNamespace.name -AuthorizationRuleName RootManageSharedAccessKey`
3. `$storagekeys = Get-AzureRmStorageAccountKey -ResourceGroupName $rgname -Name $storagename`
4. `$storagekey = $storagekeys[0].Value`

Run the AzLog command for each event hub:

1. `$eventhubs = Get-AzureRmEventHub -ResourceGroupName $rgname -NamespaceName $eventHubNamespaceName`
2. `$eventhubs.Name | %{$eventhubs | Add-AzLogEventSource -Name $sub' - '$_ -StorageAccount $storage.StorageAccountName - StorageKey $storageKey -EventHubConnectionString $eventHubKey.PrimaryConnectionString -EventHubName $_}`

After a minute or so of running the last two commands, you should see JSON files being generated. You can confirm that by monitoring the directory **C:\users\AzLog\EventHubJson**.

Next steps

- [Azure Log Integration FAQ](#)
- [Get started with Azure Log Integration](#)
- [Integrate logs from Azure resources into your SIEM systems](#)

Azure Log Integration FAQ

8/22/2017 • 4 min to read • [Edit Online](#)

This article answers frequently asked questions (FAQ) about Azure Log Integration.

Azure Log Integration is a Windows operating system service that you can use to integrate raw logs from your Azure resources into your on-premises security information and event management (SIEM) systems. This integration provides a unified dashboard for all your assets, on-premises or in the cloud. You can then aggregate, correlate, analyze, and alert for security events associated with your applications.

Is the Azure Log Integration software free?

Yes. There is no charge for the Azure Log Integration software.

Where is Azure Log Integration available?

It is currently available in Azure Commercial and Azure Government and is not available in China or Germany.

How can I see the storage accounts from which Azure Log Integration is pulling Azure VM logs?

Run the command **azlog source list**.

How can I tell which subscription the Azure Log Integration logs are from?

In the case of audit logs that are placed in the **AzureResourcemanagerJson** directories, the subscription ID is in the log file name. This is also true for logs in the **AzureSecurityCenterJson** folder. For example:

20170407T070805_2768037.0000000023.**1111e5ee-1111-111b-a11e-1e111e1111dc.json**

Azure Active Directory audit logs include the tenant ID as part of the name.

Diagnostic logs that are read from an event hub do not include the subscription ID as part of the name. Instead, they include the friendly name specified as part of the creation of the event hub source.

How can I update the proxy configuration?

If your proxy setting does not allow Azure storage access directly, open the **AZLOG.EXE.CONFIG** file in **c:\Program Files\Microsoft Azure Log Integration**. Update the file to include the **defaultProxy** section with the proxy address of your organization. After the update is done, stop and start the service by using the commands **net stop azlog** and **net start azlog**.

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <system.net>
    <connectionManagement>
      <add address="*" maxconnection="400" />
    </connectionManagement>
    <defaultProxy>
      <proxy usesystemdefault="true"
        proxyaddress=http://127.0.0.1:8888
        bypassonlocal="true" />
    </defaultProxy>
  </system.net>
  <system.diagnostics>
    <performanceCounters filemappingsize="20971520" />
  </system.diagnostics>
```

How can I see the subscription information in Windows events?

Append the subscription ID to the friendly name while adding the source:

```
Azlog source add <sourcefriendlyname>.<subscription id> <StorageName> <StorageKey>
```

The event XML has the following metadata, including the subscription ID:

```

SubjectDomainName WORKGROUP
SubjectLogonId 0x3e7
TargetUserId S-1-5-18
TargetUserName SYSTEM
TargetDomainName NT AUTHORITY
TargetLogonId 0x3e7
LogonType 5
LogonProcessName Advapi
AuthenticationPackageName Negotiate
WorkstationName
LogonGuid {00000000-0000-0000-0000-000000000000}
TransmittedServices -
LmPackageName -
KeyLength 0
ProcessId 0x234
ProcessName C:\Windows\System32\services.exe
IpAddress -
IpPort -
ImpersonationLevel %%%1833
- UserData
  - AzureSielIntegration
    SubscriptionId 00000000-0000-0000-0000-000000000000
    RoleName IaaS
    RoleInstanceId _azsiemdemo
    SourceStorageAccount azsiem9414
    SourceFriendlyName azsiem9414.SLAMDataAnalysis

```

Error messages

When I run the command `azlog createazureid`, why do I get the following error?

Error:

Failed to create AAD Application - Tenant 72f988bf-86f1-41af-91ab-2d7cd011db37 - Reason = 'Forbidden' - Message = 'Insufficient privileges to complete the operation.'

The **azlog createazureid** command attempts to create a service principal in all the Azure AD tenants for the subscriptions that the Azure login has access to. If your Azure login is only a guest user in that Azure AD tenant, the command fails with "Insufficient privileges to complete the operation." Ask the tenant admin to add your account as a user in the tenant.

When I run the command `azlog authorize`, why do I get the following error?

Error:

Warning creating Role Assignment - AuthorizationFailed: The client janedo@microsoft.com' with object id 'fe9e03e4-4dad-4328-910f-fd24a9660bd2' does not have authorization to perform action 'Microsoft.Authorization/roleAssignments/write' over scope '/subscriptions/70d95299-d689-4c97-b971-0d8ff0000000'.

The **azlog authorize** command assigns the role of reader to the Azure AD service principal (created with **azlog**

createazureid) to the provided subscriptions. If the Azure login is not a co-administrator or an owner of the subscription, it fails with an "Authorization Failed" error message. Azure Role-Based Access Control (RBAC) of co-administrator or owner is needed to complete this action.

Where can I find the definition of the properties in the audit log?

See:

- [Audit operations with Azure Resource Manager](#)
- [List the management events in a subscription in the Azure Monitor REST API](#)

Where can I find details on Azure Security Center alerts?

See [Managing and responding to security alerts in Azure Security Center](#).

How can I modify what is collected with VM diagnostics?

For details on how to get, modify, and set the Azure Diagnostics configuration, see [Use PowerShell to enable Azure Diagnostics in a virtual machine running Windows](#).

The following example gets the Azure Diagnostics configuration:

```
-AzureRmVMDiagnosticsExtension -ResourceGroupName AzLog-Integration -VMName AzlogClient
$publicsettings = (Get-AzureRmVMDiagnosticsExtension -ResourceGroupName AzLog-Integration -VMName
AzlogClient).PublicSettings
$encodedconfig = (ConvertFrom-Json -InputObject $publicsettings).xmlCfg
$xmlconfig = [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($encodedconfig))
Write-Host $xmlconfig

$xmlconfig | Out-File -Encoding utf8 -FilePath "d:\WADConfig.xml"
```

The following example modifies the Azure Diagnostics configuration. In this configuration, only event ID 4624 and event ID 4625 are collected from the security event log. Microsoft Antimalware for Azure events are collected from the system event log. For details on the use of XPath expressions, see [Consuming Events](#).

```
<WindowsEventLog scheduledTransferPeriod="PT1M">
  <DataSource name="Security!*[System[(EventID=4624 or EventID=4625)]]" />
  <DataSource name="System!*[System[Provider[@Name='Microsoft Antimalware']]]" />
</WindowsEventLog>
```

The following example sets the Azure Diagnostics configuration:

```
$diagnosticsconfig_path = "d:\WADConfig.xml"
Set-AzureRmVMDiagnosticsExtension -ResourceGroupName AzLog-Integration -VMName AzlogClient -
DiagnosticsConfigurationPath $diagnosticsconfig_path -StorageAccountName log3121 -StorageAccountKey <storage
key>
```

After you make changes, check the storage account to ensure that the correct events are collected.

If you have any issues during the installation and configuration, please open a [support request](#). Select **Log Integration** as the service for which you are requesting support.

Can I use Azure Log Integration to integrate Network Watcher logs into my SIEM?

Azure Network Watcher generates large quantities of logging information. These logs are not meant to be sent to

a SIEM. The only supported destination for Network Watcher logs is a storage account. Azure Log Integration does not support reading these logs and making them available to a SIEM.

Azure Service Fabric security best practices

9/8/2017 • 10 min to read • [Edit Online](#)

Deploying an application on Azure is fast, easy, and cost-effective. Before you deploy your cloud application into production, review our list of essential and recommended best practices for implementing secure clusters in your application.

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices. Service Fabric also addresses the significant challenges in developing and managing cloud applications. Developers and administrators can avoid complex infrastructure problems and focus on implementing mission-critical, demanding workloads that are scalable, reliable, and manageable.

For each best practice, we explain:

- What the best practice is.
- Why you should implement the best practice.
- What might happen if you don't implement the best practice.
- How you can learn to implement the best practice.

We recommend the following Azure Service Fabric security best practices:

- Use Azure Resource Manager templates and the Service Fabric PowerShell module to create secure clusters.
- Use X.509 certificates.
- Configure security policies.
- Implement the Reliable Actors security configuration.
- Configure SSL for Azure Service Fabric.
- Use network isolation and security with Azure Service Fabric.
- Configure Azure Key Vault for security.
- Assign users to roles.

Best practices for securing your clusters

Always use a secure cluster:

- Implement cluster security by using certificates.
- Provide client access (admin and read-only) by using Azure Active Directory (Azure AD).

Use automated deployments:

- Use scripts to generate, deploy, and roll over the secrets.
- Store the secrets in Azure Key Vault and use Azure AD for all other client access.
- Require authentication for human access to the secrets.

Additionally, consider the following configuration options:

- Create perimeter networks (also known as demilitarized zones, DMZs, and screened subnets) by using Azure Network Security Groups (NSGs).
- Access cluster virtual machines (VMs) or manage your cluster by using jump servers with Remote Desktop Connection.

Your clusters must be secured to prevent unauthorized users from connecting, especially when a cluster is running

in production. Although it's possible to create an unsecured cluster, anonymous users can connect to your cluster if the cluster exposes management endpoints to the public internet.

There are three [scenarios](#) for implementing cluster security by using various technologies:

- Node-to-node security: This scenario secures communication between the VMs and the computers in the cluster. This form of security ensures that only those computers that are authorized to join the cluster can host applications and services in the cluster. In this scenario, the clusters that run on Azure, or standalone clusters that run on Windows, can use either [certificate security](#) or [Windows security](#) for Windows Server machines.
- Client-to-node security: This scenario secures communication between a Service Fabric client and the individual nodes in the cluster.
- Role-Based Access Control (RBAC): This scenario uses separate identities (certificates, Azure AD, and so on) for each administrator and user client role that accesses the cluster. You specify the role identities when you create the cluster.

NOTE

Security recommendation for Azure clusters: Use Azure AD security to authenticate clients and certificates for node-to-node security.

To configure a standalone Windows cluster, see [Configure settings for a standalone Windows cluster](#).

Use Azure Resource Manager templates and the Service Fabric PowerShell module to create a secure cluster. For step-by-step instructions to create a secure Service Fabric cluster by using Azure Resource Manager templates, see [Creating a Service Fabric cluster](#).

Use the Azure Resource Manager template:

- Customize your cluster by using the template to configure managed storage for VM virtual hard disks (VHDs).
- Drive changes to your resource group by using the template for easy configuration management and auditing.

Treat your cluster configuration as code:

- Be thorough when checking your deployment configurations.
- Avoid using implicit commands to directly modify your resources.

Many aspects of the [Service Fabric application lifecycle](#) can be automated. The [Service Fabric PowerShell module](#) automates common tasks for deploying, upgrading, removing, and testing Azure Service Fabric applications. Managed APIs and HTTP APIs for application management are also available.

Use X.509 certificates

Always secure your clusters by using X.509 certificates or Windows security. Security is only configured at cluster creation time. It's not possible to turn on security after the cluster is created.

To specify a [cluster certificate](#), set the value of the **ClusterCredentialType** property to X509. To specify a server certificate for outside connections, set the **ServerCredentialType** property to X509.

In addition, follow these practices:

- Create the certificates for production clusters by using a correctly configured Windows Server certificate service. You can also obtain the certificates from an approved certificate authority (CA).
- Never use a temporary or test certificate for production clusters if the certificate was created by using MakeCert.exe or a similar tool.
- Use a self-signed certificate for test clusters, but not for production clusters.

If the cluster is unsecure, anyone can connect to the cluster anonymously and perform management operations. For this reason, always secure production clusters by using X.509 certificates or Windows security.

To learn more about using X.509 certificates, see [Add or remove certificates for a Service Fabric cluster](#).

Configure security policies

Service Fabric also secures the resources that are used by applications. Resources like files, directories, and certificates are stored under the user accounts when the application is deployed. This feature makes running applications more secure from one another, even in a shared hosted environment.

- Use an Active Directory domain group or user: Run the service under the credentials for an Active Directory user or group account. Be sure to use Active Directory on-premises within your domain and not Azure Active Directory. Access other resources in the domain that have been granted permissions by using a domain user or group. For example, resources such as file shares.
- Assign a security access policy for HTTP and HTTPS endpoints: Specify the **SecurityAccessPolicy** property to apply a **RunAs** policy to a service when the service manifest declares endpoint resources with HTTP. Ports allocated to the HTTP endpoints are correctly access-controlled lists for the RunAs user account that the service runs under. When the policy isn't set, http.sys doesn't have access to the service and you can get failures with calls from the client.

To learn how to use security policies in a Service Fabric cluster, see [Configure security policies for your application](#).

Implement the Reliable Actors security configuration

Service Fabric Reliable Actors is an implementation of the actor design pattern. As with any software design pattern, the decision to use a specific pattern is based on whether a software problem fits the pattern.

In general, use the actor design pattern to help model solutions for the following software problems or security scenarios:

- Your problem space involves a large number (thousands or more) of small, independent, and isolated units of state and logic.
- You're working with single-threaded objects that don't require significant interaction from external components, including querying state across a set of actors.
- Your actor instances don't block callers with unpredictable delays by issuing I/O operations.

In Service Fabric, actors are implemented in the Reliable Actors application framework. This framework is based on the actor pattern and built on top of [Service Fabric Reliable Services](#). Each reliable actor service that you write is a partitioned stateful reliable service.

Every actor is defined as an instance of an actor type, identical to the way a .NET object is an instance of a .NET type. For example, an **actor type** that implements the functionality of a calculator can have many actors of that type that are distributed on various nodes across a cluster. Each of the distributed actors is uniquely characterized by an actor identifier.

[Replicator security configurations](#) are used to secure the communication channel that is used during replication. This configuration prevents services from seeing each other's replication traffic and ensures that highly available data is secure. By default, an empty security configuration section prevents replication security. Replicator configurations configure the replicator that is responsible for making the Actor State Provider state highly reliable.

Configure SSL for Azure Service Fabric

The server authentication process [authenticates](#) the cluster management endpoints to a management client. The management client then recognizes that it's talking to the real cluster. This certificate also provides an [SSL](#) for the

HTTPS management API and for Service Fabric Explorer over HTTPS. You must obtain a custom domain name for your cluster. When you request a certificate from a certificate authority, the certificate's subject name must match the custom domain name that you use for your cluster.

To configure SSL for an application, you first need to obtain an SSL certificate that has been signed by a CA. The CA is a trusted third party that issues certificates for SSL security purposes. If you don't already have an SSL certificate, you need to obtain one from a company that sells SSL certificates.

The certificate must meet the following requirements for SSL certificates in Azure:

- The certificate must contain a private key.
- The certificate must be created for key exchange and be exportable to a personal information exchange (.pfx) file.
- The certificate's subject name must match the domain name that is used to access your cloud service.
 - Acquire a custom domain name to use for accessing your cloud service.
 - Request a certificate from a CA with a subject name that matches your service's custom domain name.
For example, if your custom domain name is **contoso.com**, the certificate from your CA should have the subject name **.contoso.com** or **www.contoso.com**.

NOTE

You cannot obtain an SSL certificate from a CA for the **cloudapp.net** domain.

- The certificate must use a minimum of 2,048-bit encryption.

The HTTP protocol is unsecure and subject to eavesdropping attacks. Data that is transmitted over HTTP is sent as plain text from the web browser to the web server or between other endpoints. Attackers can intercept and view sensitive data that is sent via HTTP, such as credit card details and account logins. When data is sent or posted through a browser via HTTPS, SSL ensures that sensitive information is encrypted and secure from interception.

To learn more about using SSL certificates, see [Configure SSL for Azure applications](#).

Use network isolation and security with Azure Service Fabric

Set up a 3 nodetype secure cluster by using the [Azure Resource Manager template](#) as a sample. Control the inbound and outbound network traffic by using the template and Network Security Groups.

The template has an NSG for each of the virtual machine scale sets and is used to control the traffic in and out of the set. The rules are configured by default to allow all traffic necessary for the system services and the application ports specified in the template. Review these rules and make any changes to fit your needs, including adding new rules for your applications.

For more information, see [Common networking scenarios for Azure Service Fabric](#).

Set up Azure Key Vault for security

Service Fabric uses certificates to provide authentication and encryption for securing a cluster and its applications.

Service Fabric uses X.509 certificates to secure a cluster and to provide application security features. You use Azure Key Vault to [manage certificates](#) for Service Fabric clusters in Azure. The Azure resource provider that creates the clusters pulls the certificates from a key vault. The provider then installs the certificates on the VMs when the cluster is deployed on Azure.

A certificate relationship exists between [Azure Key Vault](#), the Service Fabric cluster, and the resource provider that uses the certificates. When the cluster is created, information about the certificate relationship is stored in a key

vault.

There are two basic steps to set up a key vault:

1. Create a resource group specifically for your key vault.

We recommend that you put the key vault in its own resource group. This action helps to prevent the loss of your keys and secrets if other resource groups are removed, such as storage, compute, or the group that contains your cluster. The resource group that contains your key vault must be in the same region as the cluster that is using it.

2. Create a key vault in the new resource group.

The key vault must be enabled for deployment. The compute resource provider can then get the certificates from the vault and install them on the VM instances.

To learn more about how to set up a key vault, see [Get started with Azure Key Vault](#).

Assign users to roles

After you've created the applications to represent your cluster, assign your users to the roles that are supported by Service Fabric: read-only and admin. You can assign these roles by using the Azure classic portal.

NOTE

For more information about using roles in Service Fabric, see [Role-Based Access Control for Service Fabric clients](#).

Azure Service Fabric supports two access control types for clients that are connected to a [Service Fabric cluster](#): administrator and user. The cluster administrator can use access control to limit access to certain cluster operations for different groups of users. Access control makes the cluster more secure.

Next steps

- Set up your Service Fabric [development environment](#).
- Learn about [Service Fabric support options](#).

Azure Service Fabric security checklist

8/9/2017 • 2 min to read • [Edit Online](#)

This article provides an easy-to-use checklist that will help you secure your Azure Service Fabric environment.

Introduction

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices. Service Fabric also addresses the significant challenges in developing and managing cloud applications. Developers and administrators can avoid complex infrastructure problems and focus on implementing mission-critical, demanding workloads that are scalable, reliable, and manageable.

Checklist

Use the following checklist to help you make sure that you haven't overlooked any important issues in management and configuration of a secure Azure Service Fabric solution.

CHECKLIST CATEGORY	DESCRIPTION
Role based access control (RBAC)	<ul style="list-style-type: none">Access control allows the cluster administrator to limit access to certain cluster operations for different groups of users, making the cluster more secure.Administrators have full access to management capabilities (including read/write capabilities).Users, by default, have only read access to management capabilities (for example, query capabilities), and the ability to resolve applications and services.
X.509 certificates and Service Fabric	<ul style="list-style-type: none">Certificates used in clusters running production workloads should be created by using a correctly configured Windows Server certificate service or obtained from an approved Certificate Authority (CA).Never use any temporary or test certificates in production that are created with tools such as MakeCert.exe.You can use a self-signed certificate but, should only do so for test clusters and not in production.
Cluster Security	<ul style="list-style-type: none">The cluster security scenarios include Node-to-node security, Client-to-node security, Role-based access control (RBAC).
Cluster authentication	<ul style="list-style-type: none">Authenticates node-to-node communication for cluster federation.
Server authentication	<ul style="list-style-type: none">Authenticates the cluster management endpoints to a management client.

CHECKLIST CATEGORY	DESCRIPTION
Application security	<ul style="list-style-type: none"> Encryption and decryption of application configuration values. Encryption of data across nodes during replication.
Cluster Certificate	<ul style="list-style-type: none"> This certificate is required to secure the communication between the nodes on a cluster. Set the thumbprint of the primary certificate in the Thumbprint section and that of the secondary in the ThumbprintSecondary variables.
ServerCertificate	<ul style="list-style-type: none"> This certificate is presented to the client when it tries to connect to this cluster. You can use two different server certificates, a primary and a secondary for upgrade.
ClientCertificateThumbprints	<ul style="list-style-type: none"> This is a set of certificates that you want to install on the authenticated clients.
ClientCertificateCommonNames	<ul style="list-style-type: none"> Set the common name of the first client certificate for the CertificateCommonName. The CertificateIssuerThumbprint is the thumbprint for the issuer of this certificate.
ReverseProxyCertificate	<ul style="list-style-type: none"> This is an optional certificate that can be specified if you want to secure your Reverse Proxy.
Key Vault	<ul style="list-style-type: none"> Used to manage certificates for Service Fabric clusters in Azure.

Next steps

- [Service Fabric Cluster upgrade process and expectations from you](#)
- [Managing your Service Fabric applications in Visual Studio.](#)
- [Service Fabric Health model introduction.](#)

Azure Identity Management and access control security best practices

6/27/2017 • 10 min to read • [Edit Online](#)

Many consider identity to be the new boundary layer for security, taking over that role from the traditional network-centric perspective. This evolution of the primary pivot for security attention and investments come from the fact that network perimeters have become increasingly porous and that perimeter defense cannot be as effective as they once were prior to the explosion of **BYOD** devices and cloud applications.

In this article we will discuss a collection of Azure identity management and access control security best practices. These best practices are derived from our experience with [Azure AD](#) and the experiences of customers like yourself.

For each best practice, we'll explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

This Azure identity management and access control security best practices article is based on a consensus opinion and Azure platform capabilities and feature sets, as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure identity management and access control security best practices discussed in this article include:

- Centralize your identity management
- Enable Single Sign-On (SSO)
- Deploy password management
- Enforce multi-factor authentication (MFA) for users
- Use role based access control (RBAC)
- Control locations where resources are created using resource manager
- Guide developers to leverage identity capabilities for SaaS apps
- Actively monitor for suspicious activities

Centralize your identity management

One important step towards securing your identity is to ensure that IT can manage accounts from one single location regarding where this account was created. While the majority of the enterprises IT organizations will have their primary account directory on-premises, hybrid cloud deployments are on the rise and it is important that you understand how to integrate on-premises and cloud directories and provide a seamless experience to the end user.

To accomplish this [hybrid identity](#) scenario we recommend two options:

- Synchronize your on-premises directory with your cloud directory using [Azure AD Connect](#)
- Federate your on-premises identity with your cloud directory using [Active Directory Federation Services \(AD FS\)](#)

Organizations that fail to integrate their on-premises identity with their cloud identity will experience increased administrative overhead in managing accounts, which increases the likelihood of mistakes and security breaches.

For more information on Azure AD synchronization, please read the article [Integrating your on-premises identities](#)

with Azure Active Directory.

Enable Single Sign-On (SSO)

When you have multiple directories to manage, this becomes an administrative problem not only for IT, but also for end users that will have to remember multiple passwords. By using [SSO](#) you will provide your users the ability of use the same set of credentials to sign-in and access the resources that they need, regardless where this resource is located on-premises or in the cloud.

Use SSO to enable users to access their [SaaS applications](#) based on their organizational account in Azure AD. This is applicable not only for Microsoft SaaS apps, but also other apps, such as [Google Apps](#) and [Salesforce](#). Your application can be configured to use Azure AD as a [SAML-based identity](#) provider. As a security control, Azure AD will not issue a token allowing them to sign into the application unless they have been granted access using Azure AD. You may grant access directly, or through a group that they are a member of.

NOTE

the decision to use SSO will impact how you integrate your on-premises directory with your cloud directory. If you want SSO, you will need to use federation, because directory synchronization will only provide [same sign-on experience](#).

Organizations that do not enforce SSO for their users and applications are more exposed to scenarios where users will have multiple passwords which directly increases the likelihood of users reusing passwords or using weak passwords.

You can learn more about Azure AD SSO by reading the article [AD FS management and customization with Azure AD Connect](#).

Deploy password management

In scenarios where you have multiple tenants or you want to enable users to [reset their own password](#), it is important that you use appropriate security policies to prevent abuse. In Azure you can leverage the self-service password reset capability and customize the security options to meet your business requirements.

It is particularly important to obtain feedback from these users and learn from their experiences as they try to perform these steps. Based on these experiences, elaborate a plan to mitigate potential issues that may occur during the deployment for a larger group. It is also recommended that you use the [Password Reset Registration Activity report](#) to monitor the users that are registering.

Organizations that want to avoid password change support calls but do enable users to reset their own passwords are more susceptible to a higher call volume to the service desk due to password issues. In organizations that have multiple tenants, it is imperative that you implement this type of capability and enable users to perform password reset within security boundaries that were established in the security policy.

You can learn more about password reset by reading the article [Deploying Password Management and training users to use it](#).

Enforce multi-factor authentication (MFA) for users

For organizations that need to be compliant with industry standards, such as [PCI DSS version 3.2](#), multi-factor authentication is a must have capability for authenticate users. Beyond being compliant with industry standards, enforcing MFA to authenticate users can also help organizations to mitigate credential theft type of attack, such as [Pass-the-Hash \(PtH\)](#).

By enabling Azure MFA for your users, you are adding a second layer of security to user sign-ins and transactions. In this case, a transaction might be accessing a document located in a file server or in your SharePoint Online.

Azure MFA also helps IT to reduce the likelihood that a compromised credential will have access to organization's data.

For example: you enforce Azure MFA for your users and configure it to use a phone call or text message as verification. If the user's credentials are compromised, the attacker won't be able to access any resource since he will not have access to user's phone. Organizations that do not add extra layers of identity protection are more susceptible for credential theft attack, which may lead to data compromise.

One alternative for organizations that want to keep the entire authentication control on-premises is to use [Azure Multi-Factor Authentication Server](#), also called MFA on-premises. By using this method, you will still be able to enforce multi-factor authentication, while keeping the MFA server on-premises.

For more information on Azure MFA, please read the article [Getting started with Azure Multi-Factor Authentication in the cloud](#).

Use role based access control (RBAC)

Restricting access based on the [need to know](#) and [least privilege](#) security principles is imperative for organizations that want to enforce security policies for data access. Azure Role-Based Access Control (RBAC) can be used to assign permissions to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource.

You can leverage [built in RBAC](#) roles in Azure to assign privileges to users. Consider using *Storage Account Contributor* for cloud operators that need to manage storage accounts and *Classic Storage Account Contributor* role to manage classic storage accounts. For cloud operators that needs to manage VMs and storage account, consider adding them to *Virtual Machine Contributor* role.

Organizations that do not enforce data access control by leveraging capabilities such as RBAC may be giving more privileges than necessary to their users. This can lead to data compromise by allow users access to certain types of types of data (e.g., high business impact) that they shouldn't have in the first place.

You can learn more about Azure RBAC by reading the article [Azure Role-Based Access Control](#).

Control locations where resources are created using resource manager

Enabling cloud operators to perform tasks while preventing them from breaking conventions that are needed to manage your organization's resources is very important. Organizations that want to control the locations where resources are created should hard code these locations.

To achieve this, organizations can create security policies that have definitions that describe the actions or resources that are specifically denied. You assign those policy definitions at the desired scope, such as the subscription, resource group, or an individual resource.

NOTE

this is not the same as RBAC, it actually leverages RBAC to authenticate the users that have privilege to create those resources.

Leverage [Azure Resource Manager](#) to create custom policies also for scenarios where the organization wants to allow operations only when the appropriate cost center is associated; otherwise, they will deny the request.

Organizations that are not controlling how resources are created are more susceptible to users that may abuse the service by creating more resources than they need. Hardening the resource creation process is an important step to secure a multi-tenant scenario.

You can learn more about creating policies with Azure Resource Manager by reading the article [Use Policy to](#)

manage resources and control access.

Guide developers to leverage identity capabilities for SaaS apps

User identity will be leveraged in many scenarios when users access [SaaS apps](#) that can be integrated with on-premises or cloud directory. First and foremost, we recommend that developers use a secure methodology to develop these apps, such as [Microsoft Security Development Lifecycle \(SDL\)](#). Azure AD simplifies authentication for developers by providing identity as a service, with support for industry-standard protocols such as [OAuth 2.0](#) and [OpenID Connect](#), as well as open source libraries for different platforms.

Make sure to register any application that outsources authentication to Azure AD, this is a mandatory procedure. The reason behind this is because Azure AD needs to coordinate the communication with the application when handling sign-on (SSO) or exchanging tokens. The user's session expires when the lifetime of the token issued by Azure AD expires. Always evaluate if your application should use this time or if you can reduce this time. Reducing the lifetime can act as a security measure that will force users to sign out based on a period of inactivity.

Organizations that do not enforce identity control to access apps and do not guide their developers on how to securely integrate apps with their identity management system may be more susceptible to credential theft type of attack, such as [weak authentication and session management described in Open Web Application Security Project \(OWASP\) Top 10](#).

You can learn more about authentication scenarios for SaaS apps by reading [Authentication Scenarios for Azure AD](#).

Actively monitor for suspicious activities

According to [Verizon 2016 Data Breach report](#), credential compromise is still in the rise and becoming one of the most profitable businesses for cyber criminals. For this reason, it is important to have an active identity monitor system in place that can quickly detect suspicious behavior activity and trigger an alert for further investigation. Azure AD has two major capabilities that can help organizations monitor their identities: Azure AD Premium [anomaly reports](#) and Azure AD [identity protection](#) capability.

Make sure to use the anomaly reports to identify attempts to sign in [without being traced](#), [brute force](#) attacks against a particular account, attempts to sign in from multiple locations, sign in from [infected devices](#) and suspicious IP addresses. Keep in mind that these are reports. In other words, you must have processes and procedures in place for IT admins to run these reports on the daily basis or on demand (usually in an incident response scenario).

In contrast, Azure AD identity protection is an active monitoring system and it will flag the current risks on its own dashboard. Besides that, you will also receive daily summary notifications via email. We recommend that you adjust the risk level according to your business requirements. The risk level for a risk event is an indication (High, Medium, or Low) of the severity of the risk event. The risk level helps Identity Protection users prioritize the actions they must take to reduce the risk to their organization.

Organizations that do not actively monitor their identity systems are at risk of having user credentials compromised. Without knowledge that suspicious activities are taking place using these credentials, organizations won't be able to mitigate this type of threat. You can learn more about Azure Identity protection by reading [Azure Active Directory Identity Protection](#).

Securing PaaS deployments

6/27/2017 • 7 min to read • [Edit Online](#)

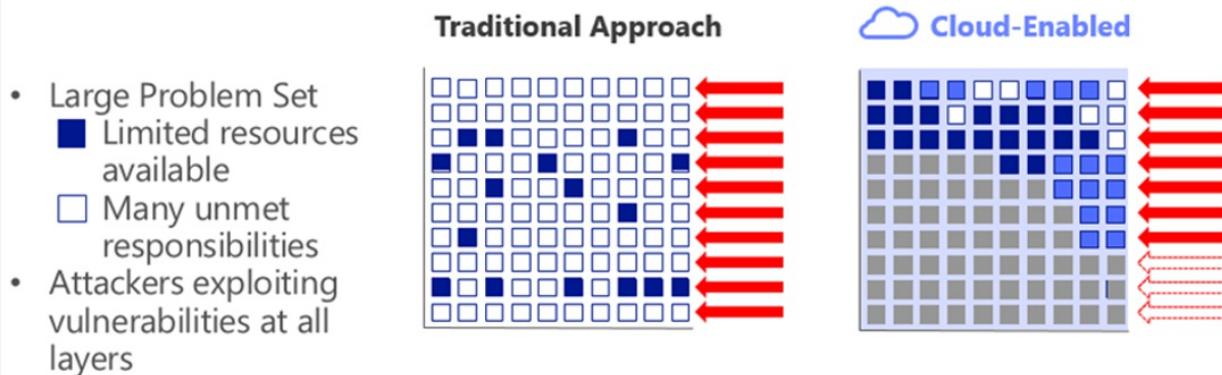
This article provides information that helps you:

- Understand the security advantages of hosting applications in the cloud
- Evaluate the security advantages of platform as a service (PaaS) versus other cloud service models
- Change your security focus from a network-centric to an identity-centric perimeter security approach
- Implement general PaaS security best practices recommendations

Cloud security advantages

There are security advantages to being in the cloud. In an on-premises environment, organizations likely have unmet responsibilities and limited resources available to invest in security, which creates an environment where attackers are able to exploit vulnerabilities at all layers.

Security Advantages of Cloud Era

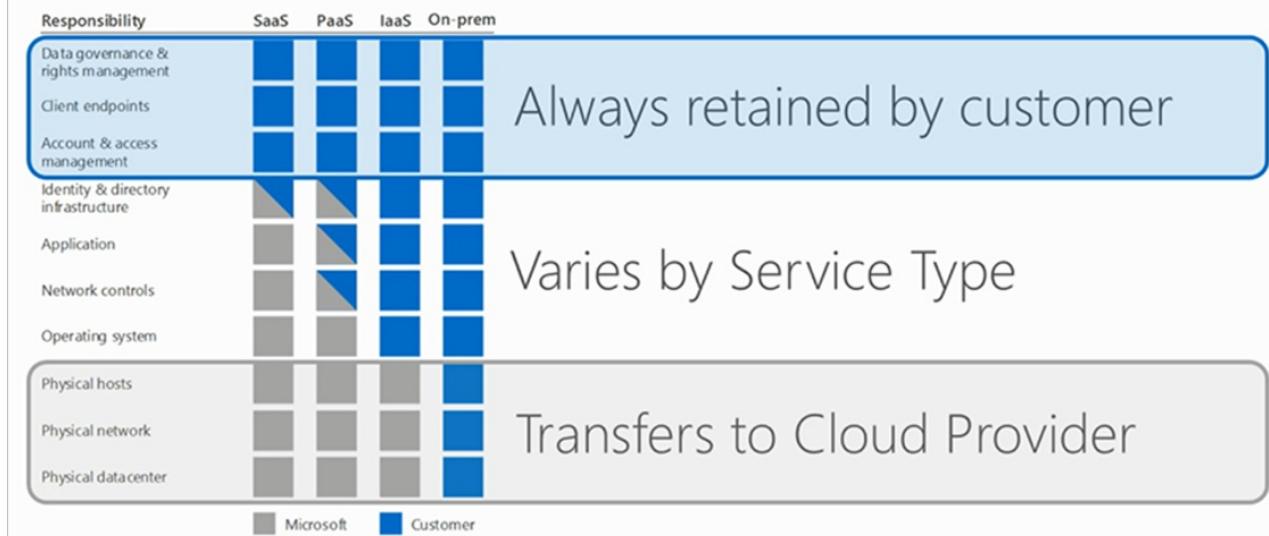


Organizations are able to improve their threat detection and response times by using a provider's cloud-based security capabilities and cloud intelligence. By shifting responsibilities to the cloud provider, organizations can get more security coverage, which enables them to reallocate security resources and budget to other business priorities.

Division of responsibility

It's important to understand the division of responsibility between you and Microsoft. On-premises, you own the whole stack but as you move to the cloud some responsibilities transfer to Microsoft. The following responsibility matrix shows the areas of the stack in a SaaS, PaaS, and IaaS deployment that you are responsible for and Microsoft is responsible for.

Responsibility Zones



For all cloud deployment types, you own your data and identities. You are responsible for protecting the security of your data and identities, on-premises resources, and the cloud components you control (which varies by service type).

Responsibilities that are always retained by you, regardless of the type of deployment, are:

- Data
- Endpoints
- Account
- Access management

Security advantages of a PaaS cloud service model

Using the same responsibility matrix, let's look at the security advantages of an Azure PaaS deployment versus on-premises.

Security advantages of PaaS

Responsibility	On-prem	PaaS		
Data governance & rights management	Customer	Customer	⚠ Application data –	Depends on key/data management
Client endpoints	Customer	Customer	⚠ User/endpoints –	Depends on least privilege design
Account & access management	Customer	Customer	⚠ Admin access –	One account → access to all apps / data / infra
Identity & directory infrastructure	Customer	Customer	⚠ Directory –	Depends on identity system / app authentication
Application	Customer	Customer	⚠ Application code –	One exploit can lead to access of all data
Network controls	Customer	Customer	⚠ Network configuration –	Depends on TLS usage
Operating system	Customer	Customer	Attack Azure Infrastructure – Extremely low attack return on investment (ROI) for a single tenant <ul style="list-style-type: none">• Active security monitoring & engineering make attack very expensive• Expense limits potential attackers to small pool with larger budgets	
Physical hosts	Customer	Customer		
Physical network	Customer	Customer		
Physical datacenter	Customer	Customer		

Legend:

- Red circle: Always attractive target
- Blue circle: App design can quickly deter attacker

Starting at the bottom of the stack, the physical infrastructure, Microsoft mitigates common risks and responsibilities. Because the Microsoft cloud is continually monitored by Microsoft, it is hard to attack. It doesn't

make sense for an attacker to pursue the Microsoft cloud as a target. Unless the attacker has lots of money and resources, the attacker is likely to move on to another target.

In the middle of the stack, there is no difference between a PaaS deployment and on-premises. At the application layer and the account and access management layer, you have similar risks. In the next steps section of this article, we will guide you to best practices for eliminating or minimizing these risks.

At the top of the stack, data governance and rights management, you take on one risk that can be mitigated by key management. (Key management is covered in best practices.) While key management is an additional responsibility, you have areas in a PaaS deployment that you no longer have to manage so you can shift resources to key management.

The Azure platform also provides you strong DDoS protection by using various network-based technologies. However, all types of network-based DDoS protection methods have their limits on a per-link and per-datacenter basis. To help avoid the impact of large DDoS attacks, you can take advantage of Azure's core cloud capability of enabling you to quickly and automatically scale out to defend against DDoS attacks. We'll go into more detail on how you can do this in the recommended practices articles.

Modernizing the defender's mindset

With PaaS deployments come a shift in your overall approach to security. You shift from needing to control everything yourself to sharing responsibility with Microsoft.

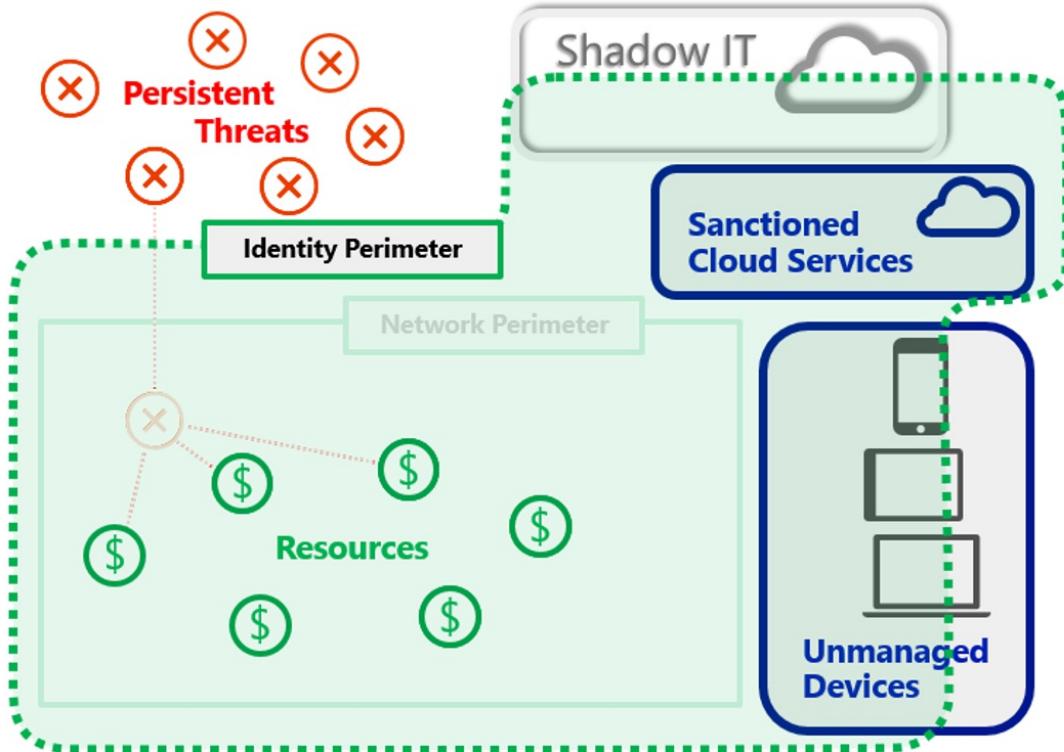
Another significant difference between PaaS and traditional on-premises deployments, is a new view of what defines the primary security perimeter. Historically, the primary on-premises security perimeter was your network and most on-premises security designs use the network as its primary security pivot. For PaaS deployments, you are better served by considering identity to be the primary security perimeter.

Identity as the primary security perimeter

One of the five essential characteristics of cloud computing is broad network access, which makes network-centric thinking less relevant. The goal of much of cloud computing is to allow users to access resources regardless of location. For most users, their location is going to be somewhere on the Internet.

The following figure shows how the security perimeter has evolved from a network perimeter to an identity perimeter. Security becomes less about defending your network and more about defending your data, as well as managing the security of your apps and users. The key difference is that you want to push security closer to what's important to your company.

The Evolving Security Perimeter



Initially, Azure PaaS services (for example, web roles and Azure SQL) provided little or no traditional network perimeter defenses. It was understood that the element's purpose was to be exposed to the Internet (web role) and that authentication provides the new perimeter (for example, BLOB or Azure SQL).

Modern security practices assume that the adversary has breached the network perimeter. Therefore, modern defense practices have moved to identity. Organizations must establish an identity-based security perimeter with strong authentication and authorization hygiene (best practices).

Recommendations for managing the identity perimeter

Principles and patterns for the network perimeter have been available for decades. In contrast, the industry has relatively less experience with using identity as the primary security perimeter. With that said, we have accumulated enough experience to provide some general recommendations that are proven in the field and apply to almost all PaaS services.

The following summarizes a general best practices approach to managing your identity perimeter.

- **Don't lose your keys or credentials** Securing keys and credentials is essential to secure PaaS deployments. Losing keys and credentials is a common problem. One good solution is to use a centralized solution where keys and secrets can be stored in hardware security modules (HSM). Azure provides you an HSM in the cloud with [Azure Key Vault](#).
- **Don't put credentials and other secrets into source code or GitHub** The only thing worse than losing your keys and credentials is having an unauthorized party gain access to them. Attackers are able to take advantage of bot technologies to find keys and secrets stored in code repositories such as GitHub. Do not put key and secrets in these public source code repositories.
- **Protect your VM management interfaces on hybrid PaaS and IaaS services** IaaS and PaaS services run on virtual machines (VMs). Depending on the type of service, several management interfaces are available that enable you to remote manage these VMs directly. Remote management protocols such as [Secure Shell Protocol \(SSH\)](#), [Remote Desktop Protocol \(RDP\)](#), and [Remote PowerShell](#) can be used. In general, we recommend that

you do not enable direct remote access to VMs from the Internet. If available, you should use alternate approaches such as using virtual private networking into an Azure virtual network. If alternative approaches are not available, then ensure that you use complex passphrases, and when available, two-factor authentication (such as [Azure Multi-Factor Authentication](#)).

- **Use strong authentication and authorization platforms**

- Use federated identities in Azure AD instead of custom user stores. When you use federated identities, you take advantage of a platform-based approach and you delegate the management of authorized identities to your partners. A federated identity approach is especially important in scenarios when employees are terminated and that information needs to be reflected through multiple identity and authorization systems.
- Use platform supplied authentication and authorization mechanisms instead of custom code. The reason is that developing custom authentication code can be error prone. Most of your developers are not security experts and are unlikely to be aware of the subtleties and the latest developments in authentication and authorization. Commercial code (for example from Microsoft) is often extensively security reviewed.
- Use multi-factor authentication. Multi-factor authentication is the current standard for authentication and authorization because it avoids the security weaknesses inherent in username and password types of authentication. Access to both the Azure management (portal/remote PowerShell) interfaces and to customer facing services should be designed and configured to use [Azure Multi-Factor Authentication \(MFA\)](#).
- Use standard authentication protocols, such as OAuth2 and Kerberos. These protocols have been extensively peer reviewed and are likely implemented as part of your platform libraries for authentication and authorization.

Next steps

In this article, we focused on security advantages of an Azure PaaS deployment. Next, learn recommended practices for securing your PaaS web and mobile solutions. We'll start with Azure App Service, Azure SQL Database, and Azure SQL Data Warehouse. As articles on recommended practices for other Azure services become available, links will be provided in the following list:

- [Azure App Service](#)
- [Azure SQL Database and Azure SQL Data Warehouse](#)
- [Azure Storage](#)
- [Azure REDIS Cache](#)
- [Azure Service Bus](#)
- [Web Application Firewalls](#)

Securing PaaS web and mobile applications using Azure App Services

7/18/2017 • 2 min to read • [Edit Online](#)

In this article, we discuss a collection of [Azure App Services](#) security best practices for securing your PaaS web and mobile applications. These best practices are derived from our experience with Azure and the experiences of customers like yourself.

Azure App Services

[Azure App Services](#) is a PaaS offering that lets you create web and mobile apps for any platform or device and connect to data anywhere, in the cloud or on-premises. App Services includes the web and mobile capabilities that were previously delivered separately as Azure Websites and Azure Mobile Services. It also includes new capabilities for automating business processes and hosting cloud APIs. As a single integrated service, App Services brings a rich set of capabilities to web, mobile, and integration scenarios.

To learn more, see overviews on [Mobile Apps](#) and [Web Apps](#).

Best practices

When using App Services, follow these best practices:

- [Authenticate through Azure Active Directory \(AD\)](#). App Services provides an OAuth 2.0 service for your identity provider. OAuth 2.0 focuses on client developer simplicity while providing specific authorization flows for Web applications, desktop applications, and mobile phones. Azure AD uses OAuth 2.0 to enable you to authorize access to mobile and web applications.
- Restrict access based on the need to know and least privilege security principles. Restricting access is imperative for organizations that want to enforce security policies for data access. Role-Based Access Control (RBAC) can be used to assign permissions to users, groups, and applications at a certain scope. To learn more about granting users access to applications, see [get started with access management](#).
- Protect your keys. It doesn't matter how good your security is if you lose your subscription keys. Azure Key Vault helps safeguard cryptographic keys and secrets used by cloud applications and services. By using Key Vault, you can encrypt keys and secrets (such as authentication keys, storage account keys, data encryption keys, .PFX files, and passwords) by using keys that are protected by hardware security modules (HSMs). For added assurance, you can import or generate keys in HSMs. See [Azure Key Vault](#) to learn more. You can also use Key Vault to manage your TLS certificates with auto-renewal.
- Restrict incoming source IP addresses. [App Services Environment](#) has a virtual network integration feature that helps you restrict incoming source IP addresses through network security groups (NSGs). If you are unfamiliar with Azure Virtual Networks (VNets), this is a capability that allows you to place many of your Azure resources in a non-internet, routable network that you control access to. To learn more, see [Integrate your app with an Azure Virtual Network](#).

Next steps

This article introduced you to a collection of App Services security best practices for securing your PaaS web and mobile applications. To learn more about securing your PaaS deployments, see:

- [Securing PaaS deployments](#)
- [Securing PaaS web and mobile applications using Azure SQL Database and SQL Data Warehouse](#)

Securing PaaS databases in Azure

7/18/2017 • 5 min to read • [Edit Online](#)

In this article, we discuss a collection of [Azure SQL Database](#) and [SQL Data Warehouse](#) security best practices for securing your PaaS web and mobile applications. These best practices are derived from our experience with Azure and the experiences of customers like yourself.

Azure SQL Database and SQL Data Warehouse

[Azure SQL Database](#) and [SQL Data Warehouse](#) provide a relational database service for your Internet-based applications. Let's look at services that help protect your applications and data when using Azure SQL Database and SQL Data Warehouse in a PaaS deployment:

- Azure Active Directory authentication (instead of SQL Server authentication)
- Azure SQL firewall
- Transparent Data Encryption (TDE)

Best practices

Use a centralized identity repository for authentication and authorization

Azure SQL databases can be configured to use one of two types of authentication:

- **SQL Authentication** uses a username and password. When you created the logical server for your database, you specified a "server admin" login with a username and password. Using these credentials, you can authenticate to any database on that server as the database owner.
- **Azure Active Directory Authentication** uses identities managed by Azure Active Directory and is supported for managed and integrated domains. To use Azure Active Directory Authentication, you must create another server admin called the "Azure AD admin," which is allowed to administer Azure AD users and groups. This admin can also perform all operations that a regular server admin can.

[Azure Active Directory authentication](#) is a mechanism of connecting to Azure SQL Database and SQL Data Warehouse by using identities in Azure Active Directory (AD). Azure AD provides an alternative to SQL Server authentication so you can stop the proliferation of user identities across database servers. Azure AD authentication enables you to centrally manage the identities of database users and other Microsoft services in one central location. Central ID management provides a single place to manage database users and simplifies permission management.

Benefits of using Azure AD authentication instead of SQL authentication include:

- Allows password rotation in a single place.
- Manages database permissions using external Azure AD groups.
- Eliminates storing passwords by enabling integrated Windows authentication and other forms of authentication supported by Azure AD.
- Uses contained database users to authenticate identities at the database level.
- Supports token-based authentication for applications connecting to SQL Database.
- Supports ADFS (domain federation) or native user/password authentication for a local Azure AD without domain synchronization.
- Supports connections from SQL Server Management Studio that use Active Directory Universal Authentication, which includes [Multi-Factor Authentication \(MFA\)](#). MFA includes strong authentication with a range of easy

verification options — phone call, text message, smart cards with pin, or mobile app notification. For more information, see [SSMS support for Azure AD MFA with SQL Database and SQL Data Warehouse](#).

To learn more about Azure AD authentication, see:

- [Connecting to SQL Database or SQL Data Warehouse By Using Azure Active Directory Authentication](#)
- [Authentication to Azure SQL Data Warehouse](#)
- [Token-based authentication support for Azure SQL DB using Azure AD authentication](#)

NOTE

To ensure that Azure Active Directory is a good fit for your environment, see [Azure AD features and limitations](#), specifically the additional considerations.

Restrict Access based on IP Address

You can create firewall rules that specify ranges of acceptable IP addresses. These rules can be targeted at both the server and database levels. We recommend using database-level firewall rules whenever possible to enhance security and to make your database more portable. Server-level firewall rules are best used for administrators and when you have many databases that have the same access requirements but you don't want to spend time configuring each database individually.

SQL Database's default source IP address restrictions allow access from any Azure address (including other subscriptions and tenants). You can restrict this to only allow your IP addresses to access the instance. Even with your SQL firewall and IP address restrictions, strong authentication is still needed. See the recommendations made earlier in this article.

To learn more about Azure SQL Firewall and IP restrictions, see:

- [Azure SQL Database access control](#)
- [Configure Azure SQL Database firewall rules - overview](#)
- [Configure an Azure SQL Database server-level firewall rule using the Azure portal](#)

Encryption of data at rest

[Transparent Data Encryption \(TDE\)](#) is enabled by default. TDE transparently encrypts SQL Server, Azure SQL Database, and Azure SQL Data Warehouse data and log files. TDE protects against a compromise of direct access to the files or their backup. This enables you to encrypt data at rest without changing existing applications. TDE should always stay enabled; however, this will not stop an attacker using the normal access path. TDE provides the ability to comply with many laws, regulations, and guidelines established in various industries.

Azure SQL manages key related issues for TDE. As with TDE, on-premises special care must be taken to ensure recoverability and when moving databases. In more sophisticated scenarios, the keys can be explicitly managed in Azure Key Vault through extensible key management (see [Enable TDE on SQL Server Using EKM](#)). This also allows for Bring Your Own Key (BYOK) through Azure Key Vaults BYOK capability.

Azure SQL provides encryption for columns through [Always Encrypted](#). This allows only authorized applications access to sensitive columns. Using this kind of encryption limits SQL queries for encrypted columns to equality-based values.

Application level encryption should also be used for selective data. Data sovereignty concerns can sometimes be mitigated by encrypting data with a key that is kept in the correct country. This prevents even accidental data transfer from causing an issue since it is impossible to decrypt the data without the key, assuming a strong algorithm is used (such as AES 256).

You can use additional precautions to help secure the database such as designing a secure system, encrypting confidential assets, and building a firewall around the database servers.

Next steps

This article introduced you to a collection of SQL Database and SQL Data Warehouse security best practices for securing your PaaS web and mobile applications. To learn more about securing your PaaS deployments, see:

- [Securing PaaS deployments](#)
- [Securing PaaS web and mobile applications using Azure App Services](#)

Azure database security best practices

8/2/2017 • 10 min to read • [Edit Online](#)

Security is a top concern when managing databases, and it has always been a priority for Azure SQL Database. Your databases can be tightly secured to help satisfy most regulatory or security requirements, including HIPAA, ISO 27001/27002, and PCI DSS Level 1, among others. A current list of security compliance certifications is available at the [Microsoft Trust Center site](#). You also can choose to place your databases in specific Azure datacenters based on regulatory requirements.

In this article, we will discuss a collection of Azure database security best practices. These best practices are derived from our experience with Azure database security and the experiences of customers like yourself.

For each best practice, we explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- How you can learn to enable the best practice

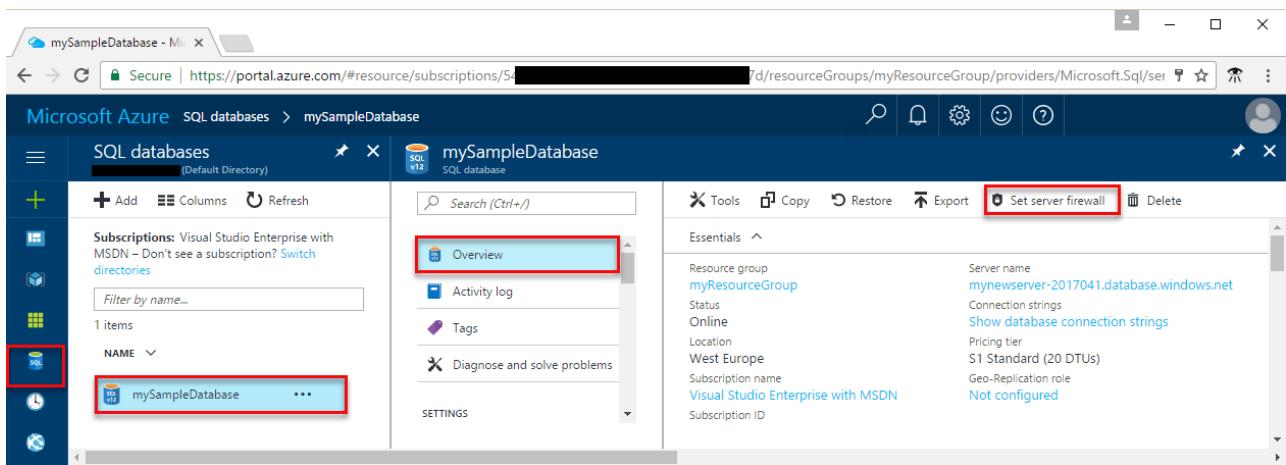
This Azure Database Security Best Practices article is based on a consensus opinion and Azure platform capabilities and feature sets as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure database security best practices discussed in this article include:

- Use firewall rules to restrict database access
- Enable database authentication
- Protect your data using encryption
- Protect data in transit
- Enable database auditing
- Enable database threat detection

Use firewall rules to restrict database access

Microsoft Azure SQL Database provides a relational database service for Azure and other Internet-based applications. To provide access security, SQL Database controls access with firewall rules limiting connectivity by IP address, authentication mechanisms requiring users to prove their identity, and authorization mechanisms limiting users to specific actions and data. Firewalls prevent all access to your database server until you specify which computers have permission. The firewall grants access to databases based on the originating IP address of each request.



The Azure SQL Database service is only available through TCP port 1433. To access a SQL Database from your computer, ensure that your client computer firewall allows outgoing TCP communication on TCP port 1433. If not needed for other applications, block inbound connections on TCP port 1433 using firewall rules.

As part of the connection process, connections from Azure virtual machines are redirected to a different IP address and port, unique for each worker role. The port number is in the range from 11000 to 11999. For more information about TCP ports, see [Ports beyond 1433 for ADO.NET 4.5 and SQL Database](#).

NOTE

For more information about firewall rules in SQL Database, see [SQL Database firewall rules](#).

Enable database authentication

SQL Database supports two types of authentication, SQL Authentication and Azure Active Directory Authentication (Azure AD Authentication).

SQL Authentication is recommended in following cases:

- It allows SQL Azure to support environments with mixed operating systems, where all users are not authenticated by a Windows domain.
- Allows SQL Azure to support older applications and applications provided by third parties that require SQL Server Authentication.
- Allows users to connect from unknown or untrusted domains. For instance, an application where established customers connect with assigned SQL Server logins to receive the status of their orders.
- Allows SQL Azure to support Web-based applications where users create their own identities.
- Allows software developers to distribute their applications by using a complex permission hierarchy based on known, preset SQL Server logins.

NOTE

However, SQL Server Authentication cannot use Kerberos security protocol.

If you use **SQL Authentication** you must:

- Manage the strong credentials yourself.
- Protect the credentials in the connection string.
- (Potentially) protect the credentials passed over the network from the Web server to the database. For more information see [how to: Connect to SQL Server Using SQL Authentication in ASP.NET 2.0](#).

Azure Active Directory authentication is a mechanism of connecting to Microsoft Azure SQL Database and [SQL](#)

[Data Warehouse](#) by using identities in Azure Active Directory (Azure AD). With Azure AD authentication, you can centrally manage the identities of database users and other Microsoft services in one central location. Central ID management provides a single place to manage database users and simplifies permission management. Benefits include the following:

- It provides an alternative to SQL Server authentication.
- Helps stop the proliferation of user identities across database servers.
- Allows password rotation in a single place.
- Customers can manage database permissions using external (AAD) groups.
- It can eliminate storing passwords by enabling integrated Windows authentication and other forms of authentication supported by Azure Active Directory.
- Azure AD authentication uses contained database users to authenticate identities at the database level.
- Azure AD supports token-based authentication for applications connecting to SQL Database.
- Azure AD authentication supports ADFS (domain federation) or native user/password authentication for a local Azure Active Directory without domain synchronization.
- Azure AD supports connections from SQL Server Management Studio that use Active Directory Universal Authentication, which includes Multi-Factor Authentication (MFA). MFA includes strong authentication with a range of easy verification options — phone call, text message, smart cards with pin, or mobile app notification.
For more information, see [SSMS support for Azure AD MFA with SQL Database and SQL Data Warehouse](#).

The configuration steps include the following procedures to configure and use Azure Active Directory authentication.

- Create and populate Azure AD.
- Optional: Associate or change the active directory that is currently associated with your Azure Subscription.
- Create an Azure Active Directory administrator for Azure SQL server or [Azure SQL Data Warehouse](#).
- Configure your client computers.
- Create contained database users in your database mapped to Azure AD identities.
- Connect to your database by using Azure AD identities.

You can find details information [here](#).

Protect your data using encryption

[Azure SQL Database transparent data encryption \(TDE\)](#) helps protect against the threat of malicious activity by performing real-time encryption and decryption of the database, associated backups, and transaction log files at rest without requiring changes to the application. TDE encrypts the storage of an entire database by using a symmetric key called the database encryption key.

Even when the entire storage is encrypted, it is very important to also encrypt your database itself. This is an implementation of the defense in depth approach for data protection. If you are using Azure SQL Database and wish to protect sensitive data such as credit card or social security numbers, you can encrypt databases with FIPS 140-2 validated 256 bit AES encryption which meets the requirements of many industry standards (e.g., HIPAA, PCI).

It's important to understand that files related to [buffer pool extension \(BPE\)](#) are not encrypted when a database is encrypted using TDE. You must use file system level encryption tools like [BitLocker](#) or the [Encrypting File System \(EFS\)](#) for BPE related files.

Since an authorized user such as a security administrator or a database administrator can access the data even if the database is encrypted with TDE, you should also follow the recommendations below:

- Enable SQL authentication at the database level.
- Use Azure AD authentication using [RBAC roles](#).

- Users and applications should use separate accounts to authenticate. This way you can limit the permissions granted to users and applications and reduce the risks of malicious activity.
- Implement database-level security by using fixed database roles (such as db_datareader or db_datawriter), or you can create custom roles for your application to grant explicit permissions to selected database objects.

For other ways to encrypt your data, consider:

- [Cell-level encryption](#) to encrypt specific columns or even cells of data with different encryption keys.
- Encryption in use using Always Encrypted: [Always Encrypted](#) allows clients to encrypt sensitive data inside client applications and never reveal the encryption keys to the Database Engine (SQL Database or SQL Server). As a result, Always Encrypted provides a separation between those who own the data (and can view it) and those who manage the data (but should have no access).
- Using Row-level security: Row-Level Security enables customers to control access to rows in a database table based on the characteristics of the user executing a query (e.g., group membership or execution context). For more information, see [Row-Level security](#).

Protect data in transit

Protecting data in transit should be essential part of your data protection strategy. Since data will be moving back and forth from many locations, the general recommendation is that you always use SSL/TLS protocols to exchange data across different locations. In some circumstances, you may want to isolate the entire communication channel between your on-premises and cloud infrastructure by using a virtual private network (VPN).

For data moving between your on-premises infrastructure and Azure, you should consider appropriate safeguards such as HTTPS or VPN.

For organizations that need to secure access from multiple workstations located on-premises to Azure, use [Azure site-to-site VPN](#).

For organizations that need to secure access from individual workstations located on-premises or off-premises to Azure, consider using [Point-to-Site VPN](#).

Larger data sets can be moved over a dedicated high-speed WAN link such as [ExpressRoute](#). If you choose to use ExpressRoute, you can also encrypt the data at the application-level using [SSL/TLS](#) or other protocols for added protection.

If you are interacting with Azure Storage through the Azure Portal, all transactions occur via HTTPS. [Storage REST API](#) over HTTPS can also be used to interact with [Azure Storage](#) and [Azure SQL Database](#).

Organizations that fail to protect data in transit are more susceptible for [man-in-the-middle attacks](#), [eavesdropping](#) and session hijacking. These attacks can be the first step in gaining access to confidential data.

To learn more about Azure VPN option by reading the article [Planning and design for VPN Gateway](#).

Enable database auditing

Auditing an instance of the SQL Server Database Engine or an individual database involves tracking and logging events that occur on the Database Engine. SQL Server audit lets you create server audits, which can contain server audit specifications for server level events, and database audit specifications for database level events. Audited events can be written to the event logs or to audit files.

There are several levels of auditing for SQL Server, depending on government or standards requirements for your installation. SQL Server Audit provides the tools and processes you must have to enable, store, and view audits on various server and database objects.

[Azure SQL Database Auditing](#) tracks database events and writes them to an audit log in your Azure Storage account.

Auditing can help you maintain regulatory compliance, understand database activity, and gain insight into discrepancies and anomalies that could indicate business concerns or suspected security violations.

Auditing enables and facilitates adherence to compliance standards but doesn't guarantee compliance.

To learn more about database auditing and how to enable it, please read the article [Enable auditing and threat detection on SQL servers in Azure Security Center](#).

Enable database threat detection

SQL Threat Detection enables you to detect and respond to potential threats as they occur by providing security alerts on anomalous activities. You will receive an alert upon suspicious database activities, potential vulnerabilities, and SQL injection attacks, as well as anomalous database access patterns. SQL Threat Detection alerts provide details of suspicious activity and recommend action on how to investigate and mitigate the threat.

For example, SQL injection is one of the common Web application security issues on the Internet, used to attack data-driven applications. Attackers take advantage of application vulnerabilities to inject malicious SQL statements into application entry fields, breaching or modifying data in the database.

To learn about how to set up threat detection for your database in the Azure portal see, [SQL Database Threat Detection](#).

In addition, SQL Threat Detection integrates alerts with [Azure Security Center](#). We invite you to try it out for 60 days for free.

To learn more about Database Threat Detection and how to enable it, please read the article [Enable auditing and threat detection on SQL servers in Azure Security Center](#).

Conclusion

Azure Database is a robust database platform, with a full range of security features that meet many organizational and regulatory compliance requirements. You can help protect data by controlling the physical access to your data, and using a variety of options for data security at the file-, column-, or row level with Transparent Data Encryption, Cell-Level Encryption, or Row-Level Security. Always Encrypted also enables operations against encrypted data, simplifying the process of application updates. In turn, access to auditing logs of SQL Database activity provides you with the information you need, allowing you to know how and when data is accessed.

Next steps

- To learn more about firewall rules, see [Firewall rules](#).
- To learn about users and logins, see [Manage logins](#).
- For a tutorial, see [Secure your Azure SQL Database](#).

Azure database security checklist

8/1/2017 • 2 min to read • [Edit Online](#)

To help improve security, Azure Database includes a number of built-in security controls that you can use to limit and control access.

These include:

- A firewall that enables you to create [firewall rules](#) limiting connectivity by IP address,
- Server-level firewall accessible from the Azure portal
- Database-level firewall rules accessible from SSMS
- Secure connectivity to your database using secure connection strings
- Use access management
- Data encryption
- SQL Database auditing
- SQL Database threat detection

Introduction

Cloud computing requires new security paradigms that are unfamiliar to many application users, database administrators, and programmers. As a result, some organizations are hesitant to implement a cloud infrastructure for data management due to perceived security risks. However, much of this concern can be alleviated through a better understanding of the security features built into Microsoft Azure and Microsoft Azure SQL Database.

Checklist

We recommend that you read the [Azure Database Security Best Practices](#) article prior to reviewing this checklist. You will be able to get the most out of this checklist after you understand the best practices. You can then use this checklist to make sure that you've addressed the important issues in Azure database security.

CHECKLIST CATEGORY	DESCRIPTION
Protect Data	
Encryption in Motion/Transit	<ul style="list-style-type: none">• Transport Layer Security, for data encryption when data is moving to the networks.• Database requires secure communication from clients based on the TDS(Tabular Data Stream) protocol over TLS (Transport Layer Security).
Encryption at rest	<ul style="list-style-type: none">• Transparent Data Encryption, when inactive data is stored physically in any digital form.
Control Access	

CHECKLIST CATEGORY	DESCRIPTION
Database Access	<ul style="list-style-type: none"> • Authentication (Azure Active Directory Authentication) AD authentication uses identities managed by Azure Active Directory. • Authorization grant users the least privileges necessary.
Application Access	<ul style="list-style-type: none"> • Row level Security (Using Security Policy, at the same time restricting row-level access based on a user's identity, role, or execution context). • Dynamic Data Masking (Using Permission & Policy, limits sensitive data exposure by masking it to non-privileged users)
Proactive Monitoring	
Tracking & Detecting	<ul style="list-style-type: none"> • Auditing tracks database events and writes them to an Audit log/ Activity log in your Azure Storage account. • Track Azure Database health using Azure Monitor Activity Logs. • Threat Detection detects anomalous database activities indicating potential security threats to the database.
Azure Security Center	<ul style="list-style-type: none"> • Data Monitoring Use Azure Security Center as a centralized security monitoring solution for SQL and other Azure services.

Conclusion

Azure Database is a robust database platform, with a full range of security features that meet many organizational and regulatory compliance requirements. You can easily protect data by controlling the physical access to your data, and using a variety of options for data security at the file-, column-, or row-level with Transparent Data Encryption, Cell-Level Encryption, or Row-Level Security. Always Encrypted also enables operations against encrypted data, simplifying the process of application updates. In turn, access to auditing logs of SQL Database activity provides you with the information you need, allowing you to know how and when data is accessed.

Next steps

You can improve the protection of your database against malicious users or unauthorized access with just a few simple steps. In this tutorial you learn to:

- Set up [firewall rules](#) for your sever and or database.
- Protect your data with [encryption](#).
- Enable [SQL Database auditing](#).

Azure operational security checklist

8/1/2017 • 4 min to read • [Edit Online](#)

Deploying an application on Azure is fast, easy, and cost-effective. Before deploying cloud application in production useful to have a checklist to assist in evaluating your application against a list of essential and recommended operational security actions for you to consider.

Introduction

Azure provides a suite of infrastructure services that you can use to deploy your applications. Azure Operational Security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure.

- To get the maximum benefit out of the cloud platform, we recommend that you leverage Azure services and follow the checklist.
- Organizations that invest time and resources assessing the operational readiness of their applications before launch have a much higher rate of satisfaction than those who don't. When performing this work, checklists can be an invaluable mechanism to ensure that applications are evaluated consistently and holistically.
- The level of operational assessment will varies depending on the organization's cloud maturity level and the application's development phase, availability needs, and data sensitivity requirements.

Checklist

This checklist is intended to help enterprises think through various operational security considerations as they deploy sophisticated enterprise applications on Azure. It can also be used to help you build a secure cloud migration and operation strategy for your organization.

CHECKLIST CATEGORY	DESCRIPTION
Security Roles & Access Controls	<ul style="list-style-type: none">• Use Role based access control (RBAC) to provide user-specific that used to assign permissions to users, groups, and applications at a certain scope.

CHECKLIST CATEGORY	DESCRIPTION
Data Collection & Storage	<ul style="list-style-type: none"> • Use Management Plane Security to secure your Storage Account using Role-Based Access Control (RBAC). • Data Plane Security to Securing Access to your Data using Shared Access Signatures (SAS) and Stored Access Policies. • Use Transport-Level Encryption – Using HTTPS and the encryption used by SMB (Server message block protocols) 3.0 for Azure File Shares. • Use Client-side encryption to secure data that you send to storage accounts when you require sole control of encryption keys. • Use Storage Service Encryption (SSE) to automatically encrypt data in Azure Storage, and Azure Disk Encryption to encrypt virtual machine disk files for the OS and data disks. • Use Azure Storage Analytics to monitor authorization type; like with Blob Storage, you can see if users have used a Shared Access Signature or the storage account keys. • Use Cross-Origin Resource Sharing (CORS) to access storage resources from different domains.
Security Policies & Recommendations	<ul style="list-style-type: none"> • Use Azure Security Center to deploy endpoint solutions. • Add a web application firewall (WAF) to secure web applications. • Use a next generation firewall (NGFW) from a Microsoft partner to increase your security protections. • Apply security contact details for your Azure subscription; this the Microsoft Security Response Centre (MSRC) contacts you if it discovers that your customer data has been accessed by an unlawful or unauthorized party.
Identity & Access Management	<ul style="list-style-type: none"> • Synchronize your on-premises directory with your cloud directory using Azure AD. • Use Single Sign-On to enable users to access their SaaS applications based on their organizational account in Azure AD. • Use the Password Reset Registration Activity report to monitor the users that are registering. • Enable multi-factor authentication (MFA) for users. • Developers to use secure identity capabilities for apps like Microsoft Security Development Lifecycle (SDL). • Actively monitor for suspicious activities by using Azure AD Premium anomaly reports and Azure AD identity protection capability.

CHECKLIST CATEGORY	DESCRIPTION
Ongoing Security Monitoring	<ul style="list-style-type: none"> • Use Malware Assessment Solution Log Analytics to report on the status of antimalware protection in your infrastructure. • Use Update assessment to determine the overall exposure to potential security problems, and whether or how critical these updates are for your environment. • The Identity and Access provide you an overview of user <ul style="list-style-type: none"> • user identity state, • number of failed attempts to log on, • the user's account that were used during those attempts, accounts that were locked out • accounts with changed or reset password • Currently number of accounts that are logged in.
Azure Security Center detection capabilities	<ul style="list-style-type: none"> • Use detection capabilities, to identify active threats targeting your Microsoft Azure resources. • Use integrated threat intelligence that looks for known bad actors by leveraging global threat intelligence from Microsoft products and services, the Microsoft Digital Crimes Unit (DCU), the Microsoft Security Response Center (MSRC), and external feeds. • Use Behavioral analytics that applies known patterns to discover malicious behavior. • Use Anomaly detection that uses statistical profiling to build a historical baseline.
Developer Operations (DevOps)	<ul style="list-style-type: none"> • Infrastructure as Code (IaC) is a practice, which enables the automation and validation of creation and teardown of networks and virtual machines to help with delivering secure, stable application hosting platforms. • Continuous Integration and Deployment drive the ongoing merging and testing of code, which leads to finding defects early. • Release Management Manage automated deployments through each stage of your pipeline. • App Performance Monitoring of running applications including production environments for application health as well as customer usage help organizations form a hypothesis and quickly validate or disprove strategies. • Using Load Testing & Auto-Scale we can find performance problems in our app to improve deployment quality and to make sure our app is always up or available to cater to the business needs.

Conclusion

Many organizations have successfully deployed and operated their cloud applications on Azure. The checklists provided highlight several checklists that is essential and help you to increase the likelihood of successful deployments and frustration-free operations. We highly recommend these operational and strategic considerations for your existing and new application deployments on Azure.

Next steps

In this document, you were introduced to OMS Security and Audit solution. To learn more about OMS Security, see the following articles:

- [Operations Management Suite \(OMS\) overview.](#)
- [Design and operational security.](#)
- [Azure Security Center planning and operations.](#)

Securing PaaS web and mobile applications using Azure Storage

8/23/2017 • 7 min to read • [Edit Online](#)

In this article, we discuss a collection of Azure Storage security best practices for securing your PaaS web and mobile applications. These best practices are derived from our experience with Azure and the experiences of customers like yourself.

The [Azure Storage security guide](#) is a great source for detailed information about Azure Storage and security. This article addresses at a high level some of the concepts found in the security guide and links to the security guide, as well as other sources, for more information.

Azure Storage

Azure makes it possible to deploy and use storage in ways not easily achievable on-premises. With Azure storage, you can reach high levels of scalability and availability with relatively little effort. Not only is Azure storage the foundation for Windows and Linux Azure Virtual Machines, it can also support large distributed applications.

Azure storage provides the following four services: Blob storage, Table storage, Queue storage, and File storage. To learn more, see [Introduction to Microsoft Azure Storage](#).

Best practices

This article addresses the following best practices:

- Access protection:
 - Shared Access Signatures (SAS)
 - Managed disk
 - Role-Based Access Control (RBAC)
- Storage encryption:
 - Client side encryption for high value data
 - Azure Disk Encryption for virtual machines (VMs)
 - Storage Service Encryption

Access protection

Use Shared Access Signature instead of a storage account key

In an IaaS solution, usually running Windows Server or Linux virtual machines, files are protected from disclosure and tampering threats using access control mechanisms. On Windows you'd use [access control lists \(ACL\)](#) and on Linux you'd probably use [chmod](#). Essentially, this is exactly what you would do if you were protecting files on a server in your own data center today.

PaaS is different. One of the most common ways to store files in Microsoft Azure is to use [Azure Blob storage](#). A difference between Blob storage and other file storage is the file I/O, and the protection methods that come with file I/O.

Access control is critical. To help you control access to Azure storage, the system will generate two 512-bit storage account keys (SAKs) when you [create a storage account](#). The level of key redundancy makes it possible for you to

avoid service interrupt during routine key rotation.

Storage access keys are high priority secrets and should only be accessible to those responsible for storage access control. If the wrong people get access to these keys, they will have complete control of storage and could replace, delete or add files to storage. This includes malware and other types of content that can potentially compromise your organization or your customers.

You still need a way to provide access to objects in storage. To provide more granular access you can take advantage of [Shared Access Signature](#) (SAS). The SAS makes it possible for you to share specific objects in storage for a pre-defined time-interval and with specific permissions. A Shared Access Signature allows you to define:

- The interval over which the SAS is valid, including the start time and the expiry time.
- The permissions granted by the SAS. For example, a SAS on a blob might grant a user read and write permissions to that blob, but not delete permissions.
- An optional IP address or range of IP addresses from which Azure Storage accepts the SAS. For example, you might specify a range of IP addresses belonging to your organization. This provides another measure of security for your SAS.
- The protocol over which Azure Storage accepts the SAS. You can use this optional parameter to restrict access to clients using HTTPS.

SAS allows you to share content the way you want to share it without giving away your Storage Account Keys. Always using SAS in your application is a secure way to share your storage resources without compromising your storage account keys.

To learn more, see [Using Shared Access Signatures](#) (SAS). To learn more about potential risks and recommendations to mitigate those risks, see [Best practices when using SAS](#).

Use managed disks for VMs

When you choose [Azure Managed Disks](#), Azure manages the storage accounts that you use for your VM disks. All you need to do is choose the type of disk (Premium or Standard) and the disk size; Azure storage will do the rest. You don't have to worry about scalability limits that might have otherwise required to you to multiple storage accounts.

To learn more, see [Frequently Asked Questions about managed and unmanaged premium disks](#).

Use Role-Based Access Control

Earlier we discussed using Shared Access Signature (SAS) to grant limited access to objects in your storage account to other clients, without exposing your account storage account key. Sometimes the risks associated with a particular operation against your storage account outweigh the benefits of SAS. Sometimes it's simpler to manage access in other ways.

Another way to manage access is to use [Azure Role-Based Access Control](#) (RBAC). With RBAC, you focus on giving employees the exact permissions they need, based on the need to know and least privilege security principles. Too many permissions can expose an account to attackers. Too few permissions means that employees can't get their work done efficiently. RBAC helps address this problem by offering fine-grained access management for Azure. This is imperative for organizations that want to enforce security policies for data access.

You can leverage built-in RBAC roles in Azure to assign privileges to users. Consider using Storage Account Contributor for cloud operators that need to manage storage accounts and Classic Storage Account Contributor role to manage classic storage accounts. For cloud operators that need to manage VMs but not the virtual network or storage account to which they are connected, consider adding them to the Virtual Machine Contributor role.

Organizations that do not enforce data access control by leveraging capabilities such as RBAC may be giving more privileges than necessary for their users. This can lead to data compromise by allowing some users access to data they shouldn't have in the first place.

To learn more about RBAC see:

- [Azure Role-Based Access Control](#)
- [Built-in roles for Azure role-based access control](#)
- [Azure Storage Security Guide](#) for detail on how to secure your storage account with RBAC

Storage encryption

Use client-side encryption for high value data

Client-side encryption enables you to programmatically encrypt data in transit before uploading to Azure Storage and programmatically decrypt data when retrieving it from storage. This provides encryption of data in transit but it also provides encryption of data at rest. Client-side encryption is the most secure method of encrypting your data but it does require you to make programmatic changes to your application and put key management processes in place.

Client-side encryption also enables you to have sole control over your encryption keys. You can generate and manage your own encryption keys. Client-side encryption uses an envelope technique where the Azure storage client library generates a content encryption key (CEK) that is then wrapped (encrypted) using the key encryption key (KEK). The KEK is identified by a key identifier and can be an asymmetric key pair or a symmetric key and can be managed locally or stored in [Azure Key Vault](#).

Client-side encryption is built into the Java and the .NET storage client libraries. See [Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage](#) for information on encrypting data within client applications and generating and managing your own encryption keys.

Azure Disk Encryption for VMs

Azure Disk Encryption is a capability that helps you encrypt your Windows and Linux IaaS virtual machine disks. Azure Disk Encryption leverages the industry standard BitLocker feature of Windows and the DM-Crypt feature of Linux to provide volume encryption for the OS and the data disks. The solution is integrated with Azure Key Vault to help you control and manage the disk-encryption keys and secrets in your key vault subscription. The solution also ensures that all data on the virtual machine disks are encrypted at rest in your Azure storage.

See [Azure Disk Encryption for Windows and Linux IaaS VMs](#).

Storage Service Encryption

When [Storage Service Encryption](#) for File storage is enabled, the data is encrypted automatically using AES-256 encryption. Microsoft handles all the encryption, decryption, and key management. This feature is available for LRS and GRS redundancy types.

Next steps

This article introduced you to a collection of Azure Storage security best practices for securing your PaaS web and mobile applications. To learn more about securing your PaaS deployments, see:

- [Securing PaaS deployments](#)
- [Securing PaaS web and mobile applications using Azure App Services](#)
- [Securing PaaS databases in Azure](#)

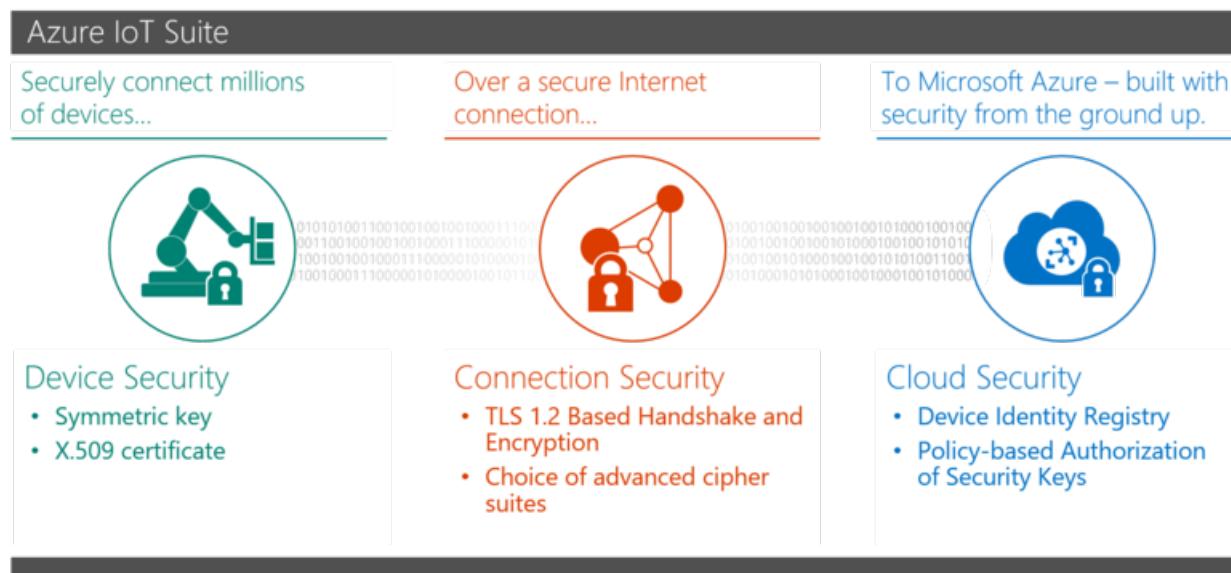
Secure your IoT deployment

8/24/2017 • 7 min to read • [Edit Online](#)

This article provides the next level of detail for securing the Azure IoT-based Internet of Things (IoT) infrastructure. It links to implementation level details for configuring and deploying each component. It also provides comparisons and choices between various competing methods.

Securing the Azure IoT deployment can be divided into the following three security areas:

- **Device Security:** Securing the IoT device while it is deployed in the wild.
- **Connection Security:** Ensuring all data transmitted between the IoT device and IoT Hub is confidential and tamper-proof.
- **Cloud Security:** Providing a means to secure data while it moves through, and is stored in the cloud.



Secure device provisioning and authentication

The Azure IoT Suite secures IoT devices by the following two methods:

- By providing a unique identity key (security tokens) for each device, which can be used by the device to communicate with the IoT Hub.
- By using an on-device [X.509 certificate](#) and private key as a means to authenticate the device to the IoT Hub. This authentication method ensures that the private key on the device is not known outside the device at any time, providing a higher level of security.

The security token method provides authentication for each call made by the device to IoT Hub by associating the symmetric key to each call. X.509-based authentication allows authentication of an IoT device at the physical layer as part of the TLS connection establishment. The security-token-based method can be used without the X.509 authentication which is a less secure pattern. The choice between the two methods is primarily dictated by how secure the device authentication needs to be, and availability of secure storage on the device (to store the private key securely).

IoT Hub security tokens

IoT Hub uses security tokens to authenticate devices and services to avoid sending keys on the network. Additionally, security tokens are limited in time validity and scope. Azure IoT SDKs automatically generate tokens

without requiring any special configuration. Some scenarios, however, require the user to generate and use security tokens directly. These include the direct use of the MQTT, AMQP, or HTTP surfaces, or the implementation of the token service pattern.

More details on the structure of the security token and its usage can be found in the following articles:

- [Security token structure](#)
- [Using SAS tokens as a device](#)

Each IoT Hub has an [identity registry](#) that can be used to create per-device resources in the service, such as a queue that contains in-flight cloud-to-device messages, and to allow access to the device-facing endpoints. The IoT Hub identity registry provides secure storage of device identities and security keys for a solution. Individual or groups of device identities can be added to an allow list, or a block list, enabling complete control over device access. The following articles provide more details on the structure of the identity registry and supported operations.

IoT Hub supports protocols such as [MQTT](#), [AMQP](#), and [HTTP](#). Each of these protocols use security tokens from the IoT device to IoT Hub differently:

- AMQP: SASL PLAIN and AMQP Claims-based security (`{policyName}@sas.root.{iothubName}` in the case of IoT hub-level tokens; `{deviceId}` in case of device-scoped tokens).
- MQTT: CONNECT packet uses `{deviceId}` as the `{ClientId}`, `{IoThubhostname}/{deviceId}` in the **Username** field and a SAS token in the **Password** field.
- HTTP: Valid token is in the authorization request header.

IoT Hub identity registry can be used to configure per-device security credentials and access control. However, if an IoT solution already has a significant investment in a [custom device identity registry and/or authentication scheme](#), it can be integrated into an existing infrastructure with IoT Hub by creating a token service.

X.509 certificate-based device authentication

The use of a [device-based X.509 certificate](#) and its associated private and public key pair allows additional authentication at the physical layer. The private key is stored securely in the device and is not discoverable outside the device. The X.509 certificate contains information about the device, such as device ID, and other organizational details. A signature of the certificate is generated by using the private key.

High-level device provisioning flow:

- Associate an identifier to a physical device – device identity and/or X.509 certificate associated to the device during device manufacturing or commissioning.
- Create a corresponding identity entry in IoT Hub – device identity and associated device information in the IoT Hub identity registry.
- Securely store X.509 certificate thumbprint in IoT Hub identity registry.

Root certificate on device

While establishing a secure TLS connection with IoT Hub, the IoT device authenticates IoT Hub using a root certificate which is part of the device SDK. For the C client SDK the certificate is located under the folder "`\certs`" under the root of the repo. Though these root certificates are long-lived, they still may expire or be revoked. If there is no way of updating the certificate on the device, the device may not be able to subsequently connect to the IoT Hub (or any other cloud service). Having a means to update the root certificate once the IoT device is deployed will effectively mitigate this risk.

Securing the connection

Internet connection between the IoT device and IoT Hub is secured using the Transport Layer Security (TLS) standard. Azure IoT supports [TLS 1.2](#), TLS 1.1 and TLS 1.0, in this order. Support for TLS 1.0 is provided for backward compatibility only. It is recommended to use TLS 1.2 since it provides the most security.

Azure IoT Suite supports the following Cipher Suites, in this order.

CIPHER SUITE	LENGTH
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028) ECDH secp384r1 (eq. 7680 bits RSA) FS	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027) ECDH secp256r1 (eq. 3072 bits RSA) FS	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014) ECDH secp384r1 (eq. 7680 bits RSA) FS	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013) ECDH secp256r1 (eq. 3072 bits RSA) FS	128
TLS_RSA_WITH_AES_256_GCM_SHA384 (0x9d)	256
TLS_RSA_WITH_AES_128_GCM_SHA256 (0x9c)	128
TLS_RSA_WITH_AES_256_CBC_SHA256 (0x3d)	256
TLS_RSA_WITH_AES_128_CBC_SHA256 (0x3c)	128
TLS_RSA_WITH_AES_256_CBC_SHA (0x35)	256
TLS_RSA_WITH_AES_128_CBC_SHA (0x2f)	128
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0xa)	112

Securing the cloud

Azure IoT Hub allows definition of [access control policies](#) for each security key. It uses the following set of permissions to grant access to each of IoT Hub's endpoints. Permissions limit the access to an IoT Hub based on functionality.

- **RegistryRead**. Grants read access to the identity registry. For more information, see [identity registry](#).
- **RegistryReadWrite**. Grants read and write access to the identity registry. For more information, see [identity registry](#).
- **ServiceConnect**. Grants access to cloud service-facing communication and monitoring endpoints. For example, it grants permission to back-end cloud services to receive device-to-cloud messages, send cloud-to-device messages, and retrieve the corresponding delivery acknowledgments.
- **DeviceConnect**. Grants access to device-facing endpoints. For example, it grants permission to send device-to-cloud messages and receive cloud-to-device messages. This permission is used by devices.

There are two ways to obtain **DeviceConnect** permissions with IoT Hub with [security tokens](#): using a device identity key, or a shared access key. Moreover, it is important to note that all functionality accessible from devices is exposed by design on endpoints with prefix `/devices/{deviceId}`.

[Service components can only generate security tokens](#) using shared access policies granting the appropriate permissions.

Azure IoT Hub and other services which may be part of the solution allow management of users using the Azure Active Directory.

Data ingested by Azure IoT Hub can be consumed by a variety of services such as Azure Stream Analytics and Azure blob storage. These services allow management access. Read more about these services and available options below:

- [Azure Cosmos DB](#): A scalable, fully-indexed database service for semi-structured data that manages metadata for the devices you provision, such as attributes, configuration, and security properties. Cosmos DB offers high-performance and high-throughput processing, schema-agnostic indexing of data, and a rich SQL query interface.
- [Azure Stream Analytics](#): Real-time stream processing in the cloud that enables you to rapidly develop and deploy a low-cost analytics solution to uncover real-time insights from devices, sensors, infrastructure, and applications. The data from this fully-managed service can scale to any volume while still achieving high throughput, low latency, and resiliency.
- [Azure App Services](#): A cloud platform to build powerful web and mobile apps that connect to data anywhere; in the cloud or on-premises. Build engaging mobile apps for iOS, Android, and Windows. Integrate with your Software as a Service (SaaS) and enterprise applications with out-of-the-box connectivity to dozens of cloud-based services and enterprise applications. Code in your favorite language and IDE (.NET, Node.js, PHP, Python, or Java) to build web apps and APIs faster than ever.
- [Logic Apps](#): The Logic Apps feature of Azure App Service helps integrate your IoT solution to your existing line-of-business systems and automate workflow processes. Logic Apps enables developers to design workflows that start from a trigger and then execute a series of steps—rules and actions that use powerful connectors to integrate with your business processes. Logic Apps offers out-of-the-box connectivity to a vast ecosystem of SaaS, cloud-based, and on-premises applications.
- [Azure blob storage](#): Reliable, economical cloud storage for the data that your devices send to the cloud.

Conclusion

This article provides overview of implementation level details for designing and deploying an IoT infrastructure using Azure IoT. Configuring each component to be secure is key in securing the overall IoT infrastructure. The design choices available in Azure IoT provide some level of flexibility and choice; however, each choice may have security implications. It is recommended that each of these choices be evaluated through a risk/cost assessment.

See also

You can also explore some of the other features and capabilities of the IoT Suite preconfigured solutions:

- [Predictive maintenance preconfigured solution overview](#)
- [Frequently asked questions for IoT Suite](#)

You can read about IoT Hub security in [Control access to IoT Hub](#) in the IoT Hub developer guide.

Internet of Things security best practices

7/3/2017 • 6 min to read • [Edit Online](#)

To secure an Internet of Things (IoT) infrastructure requires a rigorous security-in-depth strategy. This strategy requires you to secure data in the cloud, protect data integrity while in transit over the public internet, and securely provision devices. Each layer builds greater security assurance in the overall infrastructure.

Secure an IoT infrastructure

This security-in-depth strategy can be developed and executed with active participation of various players involved with the manufacturing, development, and deployment of IoT devices and infrastructure. Following is a high-level description of these players.

- **IoT hardware manufacturer/integrator:** Typically, these are the manufacturers of IoT hardware being deployed, integrators assembling hardware from various manufacturers, or suppliers providing hardware for an IoT deployment manufactured or integrated by other suppliers.
- **IoT solution developer:** The development of an IoT solution is typically done by a solution developer. This developer may part of an in-house team or a system integrator (SI) specializing in this activity. The IoT solution developer can develop various components of the IoT solution from scratch, integrate various off-the-shelf or open-source components, or adopt preconfigured solutions with minor adaptation.
- **IoT solution deployer:** After an IoT solution is developed, it needs to be deployed in the field. This involves deployment of hardware, interconnection of devices, and deployment of solutions in hardware devices or the cloud.
- **IoT solution operator:** After the IoT solution is deployed, it requires long-term operations, monitoring, upgrades, and maintenance. This can be done by an in-house team that comprises information technology specialists, hardware operations and maintenance teams, and domain specialists who monitor the correct behavior of overall IoT infrastructure.

The sections that follow provide best practices for each of these players to help develop, deploy, and operate a secure IoT infrastructure.

IoT hardware manufacturer/integrator

The following are the best practices for IoT hardware manufacturers and hardware integrators.

- **Scope hardware to minimum requirements:** The hardware design should include the minimum features required for operation of the hardware, and nothing more. An example is to include USB ports only if necessary for the operation of the device. These additional features open the device for unwanted attack vectors that should be avoided.
- **Make hardware tamper proof:** Build in mechanisms to detect physical tampering, such as opening of the device cover or removing a part of the device. These tamper signals may be part of the data stream uploaded to the cloud, which could alert operators of these events.
- **Build around secure hardware:** If COGS permits, build security features such as secure and encrypted storage, or boot functionality based on Trusted Platform Module (TPM). These features make devices more secure and help protect the overall IoT infrastructure.
- **Make upgrades secure:** Firmware upgrades during the lifetime of the device are inevitable. Building devices with secure paths for upgrades and cryptographic assurance of firmware versions will allow the device to be secure during and after upgrades.

IoT solution developer

The following are the best practices for IoT solution developers:

- **Follow secure software development methodology:** Development of secure software requires ground-up thinking about security, from the inception of the project all the way to its implementation, testing, and deployment. The choices of platforms, languages, and tools are all influenced with this methodology. The Microsoft Security Development Lifecycle provides a step-by-step approach to building secure software.
- **Choose open-source software with care:** Open-source software provides an opportunity to quickly develop solutions. When you're choosing open-source software, consider the activity level of the community for each open-source component. An active community ensures that software is supported and that issues are discovered and addressed. Alternatively, an obscure and inactive open-source software might not be supported and issues will probably not be discovered.
- **Integrate with care:** Many software security flaws exist at the boundary of libraries and APIs. Functionality that may not be required for the current deployment might still be available via an API layer. To ensure overall security, make sure to check all interfaces of components being integrated for security flaws.

IoT solution deployer

The following are best practices for IoT solution deployers:

- **Deploy hardware securely:** IoT deployments may require hardware to be deployed in unsecure locations, such as in public spaces or unsupervised locales. In such situations, ensure that hardware deployment is tamper-proof to the maximum extent. If USB or other ports are available on the hardware, ensure that they are covered securely. Many attack vectors can use these as entry points.
- **Keep authentication keys safe:** During deployment, each device requires device IDs and associated authentication keys generated by the cloud service. Keep these keys physically safe even after the deployment. Any compromised key can be used by a malicious device to masquerade as an existing device.

IoT solution operator

The following are the best practices for IoT solution operators:

- **Keep the system up to date:** Ensure that device operating systems and all device drivers are upgraded to the latest versions. If you turn on automatic updates in Windows 10 (IoT or other SKUs), Microsoft keeps it up to date, providing a secure operating system for IoT devices. Keeping other operating systems (such as Linux) up to date helps ensure that they are also protected against malicious attacks.
- **Protect against malicious activity:** If the operating system permits, install the latest antivirus and antimalware capabilities on each device operating system. This can help mitigate most external threats. You can protect most modern operating systems against threats by taking appropriate steps.
- **Audit frequently:** Auditing IoT infrastructure for security-related issues is key when responding to security incidents. Most operating systems provide built-in event logging that should be reviewed frequently to make sure no security breach has occurred. Audit information can be sent as a separate telemetry stream to the cloud service where it can be analyzed.
- **Physically protect the IoT infrastructure:** The worst security attacks against IoT infrastructure are launched using physical access to devices. One important safety practice is to protect against malicious use of USB ports and other physical access. One key to uncovering breaches that might have occurred is logging of physical access, such as USB port use. Again, Windows 10 (IoT and other SKUs) enables detailed logging of these events.
- **Protect cloud credentials:** Cloud authentication credentials used for configuring and operating an IoT deployment are possibly the easiest way to gain access and compromise an IoT system. Protect the credentials by changing the password frequently, and refrain from using these credentials on public machines.

Capabilities of different IoT devices vary. Some devices might be computers running common desktop operating

systems, and some devices might be running very light-weight operating systems. The security best practices described previously might be applicable to these devices in varying degrees. If provided, additional security and deployment best practices from the manufacturers of these devices should be followed.

Some legacy and constrained devices might not have been designed specifically for IoT deployment. These devices might lack the capability to encrypt data, connect with the Internet, or provide advanced auditing. In these cases, a modern and secure field gateway can aggregate data from legacy devices and provide the security required for connecting these devices over the Internet. Field gateways can provide secure authentication, negotiation of encrypted sessions, receipt of commands from the cloud, and many other security features.

See also

To learn more about securing your IoT solution, see:

- [IoT security architecture](#)
- [Secure your IoT deployment](#)

You can also explore some of the other features and capabilities of the IoT Suite preconfigured solutions:

- [Predictive maintenance preconfigured solution overview](#)
- [Frequently asked questions for Azure IoT Suite](#)

You can read about IoT Hub security in [Control access to IoT Hub](#) in the IoT Hub developer guide.

4 min to read •

Microsoft Trust Center

6/27/2017 • 1 min to read • [Edit Online](#)

The Azure Security Information site on Azure.com gives you the information you need to plan, design, deploy, configure, and manage your cloud solutions securely. With the Microsoft Trust center, you also have the information you need to be confident that the Azure platform on which you run your services is secure.

We know that when you entrust your applications and data to Azure, you're going to have questions. Where is it? Who can access it? What is Microsoft doing to protect it? How can you verify that Microsoft is doing what it says?

And we have answers. Because it's your data, you decide who has access, and you work with us to decide where it is located. To safeguard your data, we use state-of-the-art security technology and world-class cryptography. Our compliance is independently audited, and we're transparent on many levels—from how we handle legal demands for your customer data to the security of our code.

Here's what you find at the Microsoft Trust Center:

- [Security](#) – Learn how all the Microsoft Cloud services are secured.
- [Privacy](#) – Understand how Microsoft ensures privacy of your Data in the Microsoft cloud.
- [Compliance](#) – Discover how Microsoft helps organizations comply with national, regional, and industry-specific requirements governing the collection and use of individuals' data.
- [Transparency](#) – View how Microsoft believes that you control your data in the cloud and how Microsoft helps you know as much as possible about how that data is handled.
- [Products and Services](#) – See all the Microsoft Cloud products and services in one place
- [Service Trust Portal](#) – Obtain copies of independent audit reports of Microsoft cloud services, risk assessments, security best practices, and related materials.
- [What's New](#) – Find out what's new in Microsoft Cloud Trust
- [Resources](#) – Investigate white papers, videos, and case studies on Microsoft Trusted Cloud

The [Microsoft Trust Center](#) has what you need to understand what we do to secure the Microsoft Cloud.

Microsoft Security Response Center

6/27/2017 • 1 min to read • [Edit Online](#)

The Microsoft Security Response Center (MSRC) is led by some of the world's most experienced security experts. These experts identify, monitor, respond to and resolve security incidents and on-premises and cloud vulnerabilities around the clock, each day of the year.

In addition to the continuous work the MSRC does in the background, the MSRC team has a number of resources available to you so that you can understand how to secure your Azure assets and deployments more effectively.

The MSRC Blog

The [MSRC blog](#) is the place to go to get the latest news on what the MSRC is doing to help protect you against cloud threats.

White Papers

The MSRC has published a number of [white papers](#) that will help you understand what they do and how they do it. Some provide insights into how we secure the Microsoft cloud and include useful information on how you can employ the same security configurations.

Security Researcher Engagement and Bounty Programs

The MSRC supports collaboration and relationships with security researchers globally to advance Microsoft product security.

Microsoft bounty programs pay researchers for novel exploitation techniques, defensive ideas that mitigate novel exploitations, and identification of critical vulnerabilities in Microsoft on-premises and cloud software.

Learn more about these programs at the [MSRC Bug Bounty](#) page and the [MSRC blog](#).

To learn more about the MSRC, please visit the [MSRC home page](#).

Pen Testing

8/24/2017 • 2 min to read • [Edit Online](#)

One of the great things about using Microsoft Azure for application testing and deployment is that you don't need to put together an on-premises infrastructure to develop, test and deploy your applications. All the infrastructure is taken care of by the Microsoft Azure platform services. You don't have to worry about requisitioning, acquiring, and "racking and stacking" your own on-premises hardware.

This is great – but you still need to make sure you perform your normal security due diligence. One of the things you need to do is penetration test the applications you deploy in Azure.

You might already know that Microsoft performs [penetration testing of our Azure environment](#). This helps us improve our platform and guides our actions in terms of improving security controls, introducing new security controls, and improving our security processes.

We don't pen test your application for you, but we do understand that you will want and need to perform pen testing on your own applications. That's a good thing, because when you enhance the security of your applications, you help make the entire Azure ecosystem more secure.

When you pen test your applications, it might look like an attack to us. We [continuously monitor](#) for attack patterns and will initiate an incident response process if we need to. It doesn't help you and it doesn't help us if we trigger an incident response due to your own due diligence pen testing.

What to do?

When you're ready to pen test your Azure-hosted applications, you have an option to [let us know](#). Once we know that you're going to be performing specific tests, we won't inadvertently shut you down (such as blocking the IP address that you're testing from), as long as your tests conform to the Azure pen testing terms and conditions described in [Microsoft Cloud Unified Penetration Testing Rules of Engagement](#). Standard tests you can perform include:

- Tests on your endpoints to uncover the [Open Web Application Security Project \(OWASP\) top 10 vulnerabilities](#)
- [Fuzz testing](#) of your endpoints
- [Port scanning](#) of your endpoints

One type of test that you can't perform is any kind of [Denial of Service \(DoS\)](#) attack. This includes initiating a DoS attack itself, or performing related tests that might determine, demonstrate or simulate any type of DoS attack.

Are you ready to get started with pen testing your applications hosted in Microsoft Azure? If so, then head on over to the [Penetration Test Overview](#) page (and click the Create a Testing Request button at the bottom of the page). You'll also find more information on the pen testing terms and conditions and helpful links on how you can report security flaws related to Azure or any other Microsoft service.

Introduction to Azure Security Center

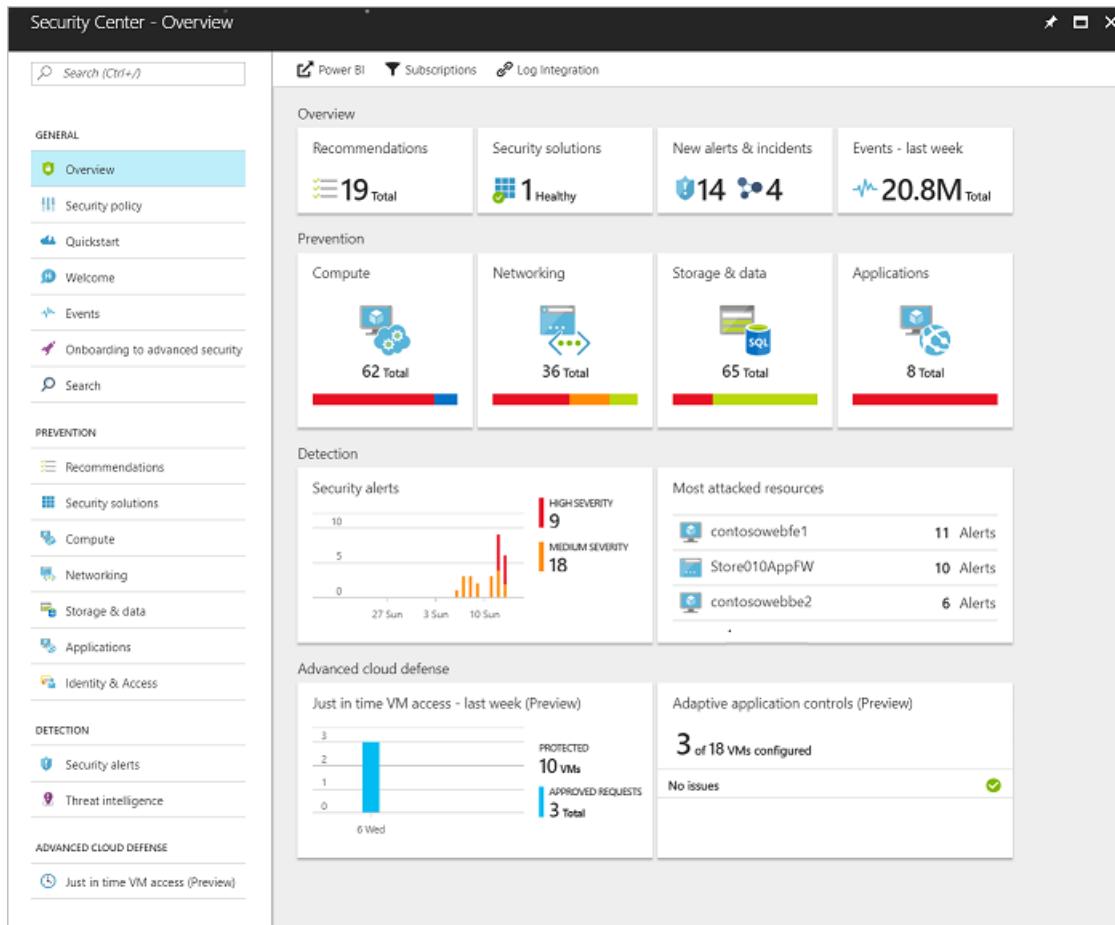
9/13/2017 • 3 min to read • [Edit Online](#)

Learn about Azure Security Center, its key capabilities, and how to get started.

What is Azure Security Center?

Azure Security Center provides unified security management and advanced threat protection for workloads running in Azure, on-premises, and in other clouds. It delivers visibility and control over hybrid cloud workloads, active defenses that reduce your exposure to threats, and intelligent detection to help you keep pace with rapidly evolving cyber attacks.

The Security Center **Overview** provides a quick view into the security posture of your Azure and non-Azure workloads, enabling you to discover and assess the security of your workloads and to identify and mitigate risk.



Why use Security Center?

Unified security management

- **Reduced management complexity.** Manage security across all your hybrid cloud workloads – on-premises, Azure, and other cloud platforms – in one console. Built-in dashboards provide instant insights into security issues that require attention.
- **Centralized policy management.** Ensure compliance with company or regulatory security requirements by centrally managing security policies across all your hybrid cloud workloads.
- **Security data from many sources.** Collect, search, and analyze security data from a variety of sources,

including connected partner solutions like network firewalls and other Microsoft services.

- **Integration with existing security workflows.** Access, integrate, and analyze security information using REST APIs to connect existing tools and processes.
- **Compliance reporting.** Use security data and insights to demonstrate compliance and easily generate evidence for auditors.

Multi-layer cyber defense

- **Continuous security assessment.** Monitor the security of machines, networks, and Azure services using hundreds of built-in security assessments or create your own. Identify software and configurations that are vulnerable to attack.
- **Actionable recommendations.** Remediate security vulnerabilities before they can be exploited by attackers with prioritized, actionable security recommendations and built-in automation playbooks.
- **Adaptive application controls.** Block malware and other unwanted applications by applying whitelisting recommendations adapted to your specific Azure workloads and powered by machine learning.
- **Network access security.** Reduce the network attack surface with just-in-time, controlled access to management ports on Azure VMs, drastically reducing exposure to brute force and other network attacks.

Intelligent threat detection and response

- **Industry's most extensive threat intelligence.** Tap into the Microsoft Intelligent Security Graph, which uses trillions of signals from Microsoft services and systems around the globe to identify new and evolving threats.
- **Advanced threat detection.** Use built-in behavioral analytics and machine learning to identify attacks and zero-day exploits. Monitor networks, machines, and cloud services for incoming attacks and post-breach activity.
- **Alerts and Incidents.** Focus on the most critical threats first with prioritized security alerts and incidents that map alerts of different types into a single attack campaign. Create your own custom security alerts as well.
- **Streamlined investigation.** Quickly assess the scope and impact of an attack with a visual, interactive experience. Use predefined or ad hoc queries for deeper exploration of security data.
- **Contextual threat intelligence.** Visualize the source of attacks on an interactive world map. Use built-in threat intelligence reports to gain valuable insight into the techniques and objectives of known malicious actors.

Get started

To get started with Security Center, you need a subscription to Microsoft Azure. Security Center is enabled with your Azure subscription. If you do not have a subscription, you can sign up for a [free trial](#).

You access Security Center from the [Azure portal](#). See the [portal documentation](#) to learn more.

[Getting started with Azure Security Center](#) quickly guides you through the security-monitoring and policy-management components of Security Center.

Next steps

In this document, you were introduced to Security Center, its key capabilities, and how to get started. To learn more, see the following resources:

- [Planning and operations guide](#) - Learn how to optimize your use of Security Center based on your organization's security requirements and cloud management model.
- [Setting security policies](#) — Learn how to configure security policies for your Azure subscriptions and resource groups.
- [Managing security recommendations](#) — Learn how recommendations help you protect your Azure resources.

- [Security health monitoring](#) — Learn how to monitor the health of your Azure resources.
- [Managing and responding to security alerts](#) — Learn how to manage and respond to security alerts.
- [Monitoring and processing security events](#) - Learn how to monitor and process security events collected over time.
- [Monitoring partner solutions](#) — Learn how to monitor the health status of your partner solutions.
- [Azure Security Center FAQ](#) — Find frequently asked questions about using the service.
- [Azure Security blog](#) — Get the latest Azure security news and information.

What is Azure Key Vault?

7/21/2017 • 3 min to read • [Edit Online](#)

Azure Key Vault is available in most regions. For more information, see the [Key Vault pricing page](#).

Introduction

Azure Key Vault helps safeguard cryptographic keys and secrets used by cloud applications and services. By using Key Vault, you can encrypt keys and secrets (such as authentication keys, storage account keys, data encryption keys, .PFX files, and passwords) by using keys that are protected by hardware security modules (HSMs). For added assurance, you can import or generate keys in HSMs. If you choose to do this, Microsoft processes your keys in FIPS 140-2 Level 2 validated HSMs (hardware and firmware).

Key Vault streamlines the key management process and enables you to maintain control of keys that access and encrypt your data. Developers can create keys for development and testing in minutes, and then seamlessly migrate them to production keys. Security administrators can grant (and revoke) permission to keys, as needed.

Use the following table to better understand how Key Vault can help to meet the needs of developers and security administrators.

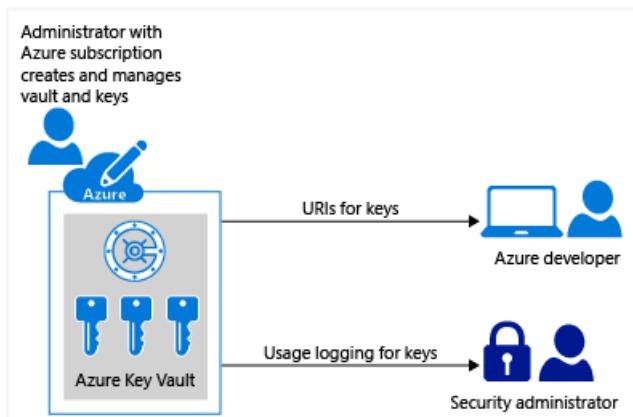
ROLE	PROBLEM STATEMENT	SOLVED BY AZURE KEY VAULT
Developer for an Azure application	<p>"I want to write an application for Azure that uses keys for signing and encryption, but I want these keys to be external from my application so that the solution is suitable for an application that is geographically distributed.</p> <p>I also want these keys and secrets to be protected, without having to write the code myself. I also want these keys and secrets to be easy for me to use from my applications, with optimal performance."</p>	<ul style="list-style-type: none">✓ Keys are stored in a vault and invoked by URI when needed.✓ Keys are safeguarded by Azure, using industry-standard algorithms, key lengths, and hardware security modules (HSMs).✓ Keys are processed in HSMs that reside in the same Azure datacenters as the applications. This provides better reliability and reduced latency than if the keys reside in a separate location, such as on-premises.
Developer for Software as a Service (SaaS)	<p>"I don't want the responsibility or potential liability for my customers' tenant keys and secrets.</p> <p>I want the customers to own and manage their keys so that I can concentrate on doing what I do best, which is providing the core software features."</p>	<ul style="list-style-type: none">✓ Customers can import their own keys into Azure, and manage them. When a SaaS application needs to perform cryptographic operations by using their customers' keys, Key Vault does these operations on behalf of the application. The application does not see the customers' keys.

ROLE	PROBLEM STATEMENT	SOLVED BY AZURE KEY VAULT
Chief security officer (CSO)	<p>"I want to know that our applications comply with FIPS 140-2 Level 2 HSMs for secure key management."</p> <p>I want to make sure that my organization is in control of the key life cycle and can monitor key usage.</p> <p>And although we use multiple Azure services and resources, I want to manage the keys from a single location in Azure."</p>	<ul style="list-style-type: none"> ✓ HSMs are FIPS 140-2 Level 2 validated. ✓ Key Vault is designed so that Microsoft does not see or extract your keys. ✓ Near real-time logging of key usage. ✓ The vault provides a single interface, regardless of how many vaults you have in Azure, which regions they support, and which applications use them.

Anybody with an Azure subscription can create and use key vaults. Although Key Vault benefits developers and security administrators, it could be implemented and managed by an organization's administrator who manages other Azure services for an organization. For example, this administrator would sign in with an Azure subscription, create a vault for the organization in which to store keys, and then be responsible for operational tasks, such as:

- Create or import a key or secret
- Revoke or delete a key or secret
- Authorize users or applications to access the key vault, so they can then manage or use its keys and secrets
- Configure key usage (for example, sign or encrypt)
- Monitor key usage

This administrator would then provide developers with URIs to call from their applications, and provide their security administrator with key usage logging information.



Developers can also manage the keys directly, by using APIs. For more information, see [the Key Vault developer's guide](#).

Next Steps

For a getting started tutorial for an administrator, see [Get Started with Azure Key Vault](#).

For more information about usage logging for Key Vault, see [Azure Key Vault Logging](#).

For more information about using keys and secrets with Azure Key Vault, see [About Keys, Secrets, and Certificates](#).

What is Log Analytics?

7/10/2017 • 4 min to read • [Edit Online](#)

Log Analytics is a service in [Operations Management Suite \(OMS\)](#) that monitors your cloud and on-premises environments to maintain their availability and performance. It collects data generated by resources in your cloud and on-premises environments and from other monitoring tools to provide analysis across multiple sources. This article provides a brief discussion of the value that Log Analytics provides, an overview of how it operates, and links to more detailed content so you can dig further.

Is Log Analytics for you?

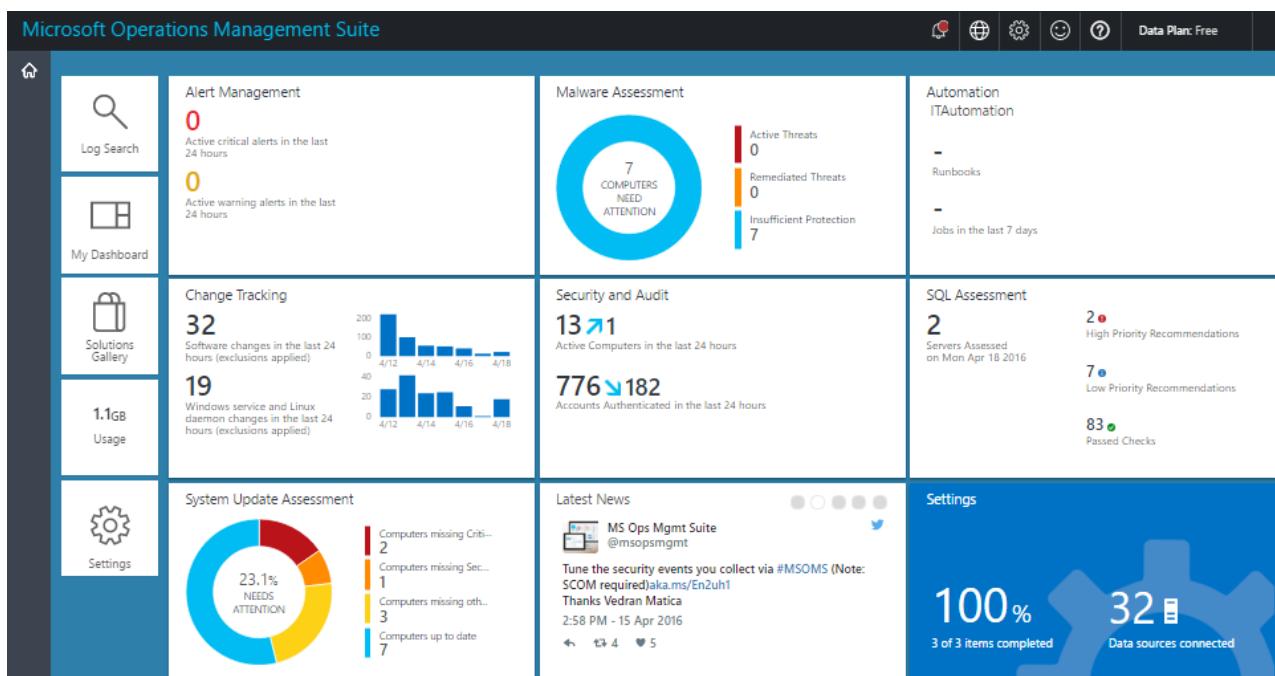
If you have no current monitoring in place for your Azure environment, you should start with [Azure Monitor](#) which collects and analyzes monitoring data for your Azure resources. Log Analytics can [collect data from Azure Monitor](#) to correlate it with other data and provide additional analysis.

If you want to monitor your on-premises environment or you have existing monitoring using services such as Azure Monitor or System Center Operations Manager, then Log Analytics can add significant value. It can collect data directly from your agents and also from these other tools into a single repository. Analysis tools in Log Analytics such as log searches, views, and solutions work against all collected data providing you with centralized analysis of your entire environment.

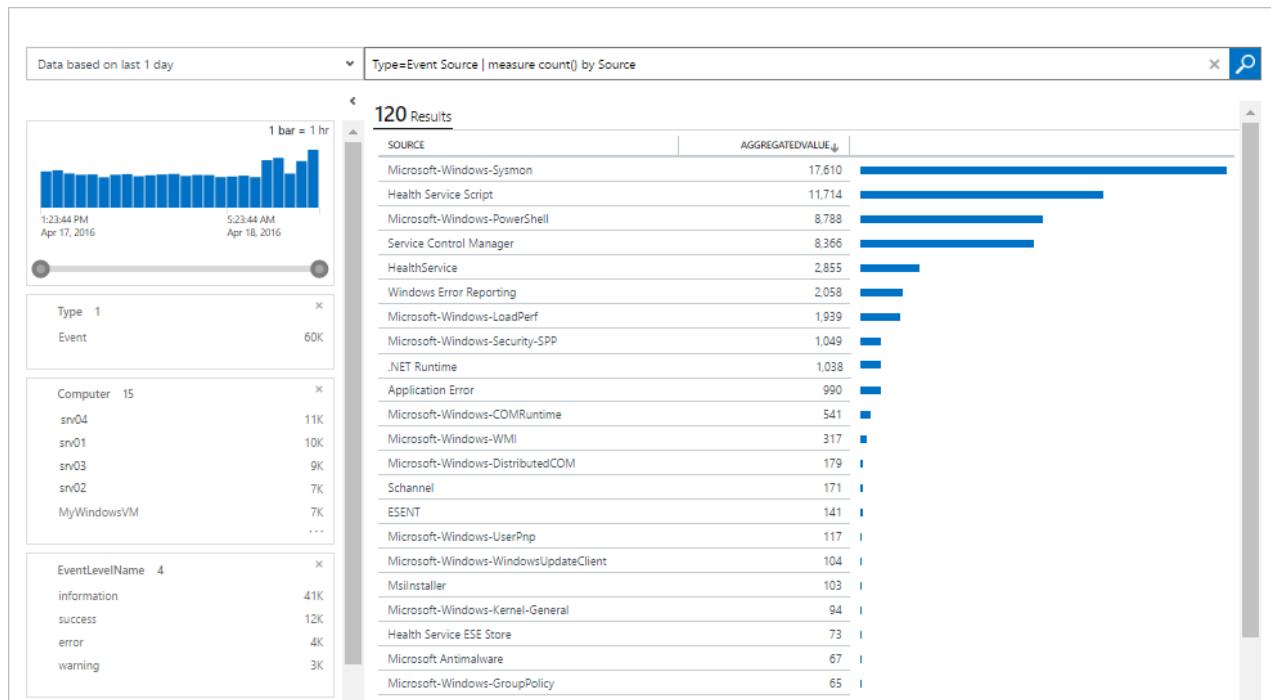
Using Log Analytics

You can access Log Analytics through the OMS portal or the Azure portal which run in any browser and provide you with access to configuration settings and multiple tools to analyze and act on collected data. From the portal you can leverage [log searches](#) where you construct queries to analyze collected data, [dashboards](#) which you can customize with graphical views of your most valuable searches, and [solutions](#) which provide additional functionality and analysis tools.

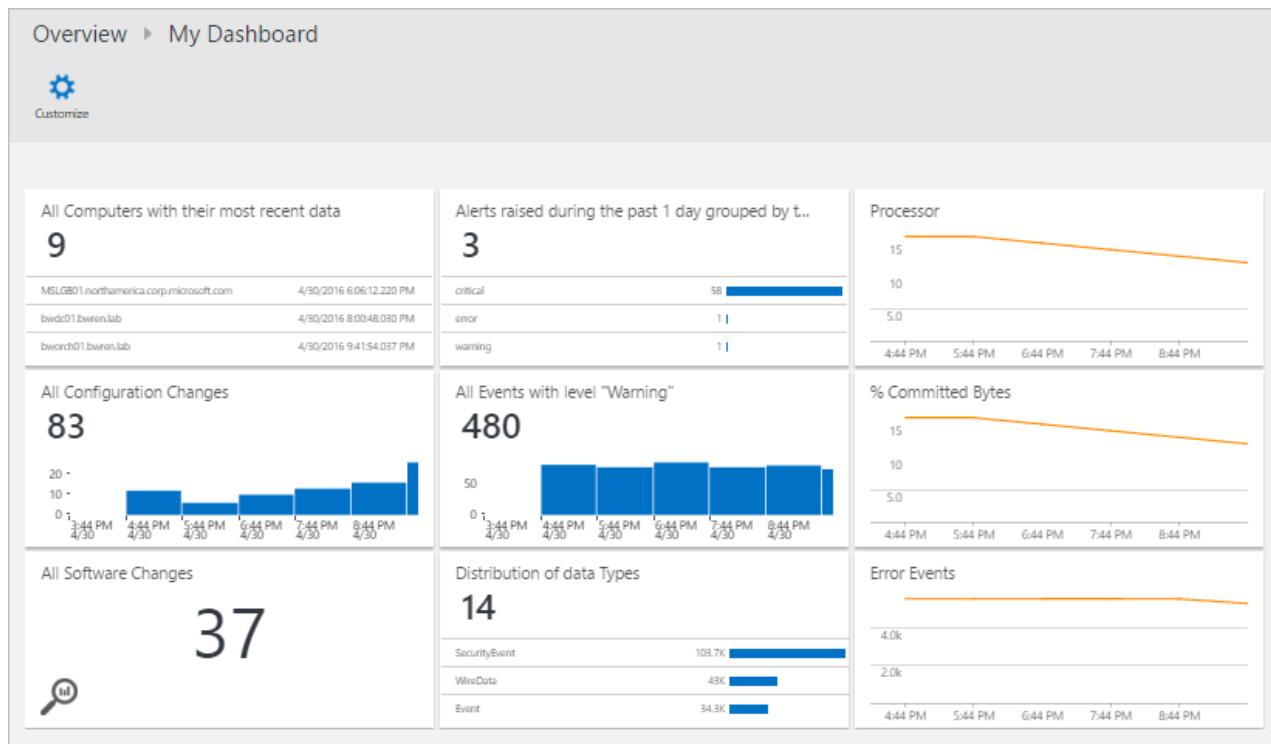
The image below is from the OMS portal which shows the dashboard that displays summary information for the [solutions](#) that are installed in the workspace. You can click on any tile to drill further into the data for that solution.



Log Analytics includes a query language to quickly retrieve and consolidate data in the repository. You can create and save [Log Searches](#) to directly analyze data in the portal or have log searches run automatically to create an alert if the results of the query indicate an important condition.



To get a quick graphical view of the health of your overall environment, you can add visualizations for saved log searches to your [dashboard](#).

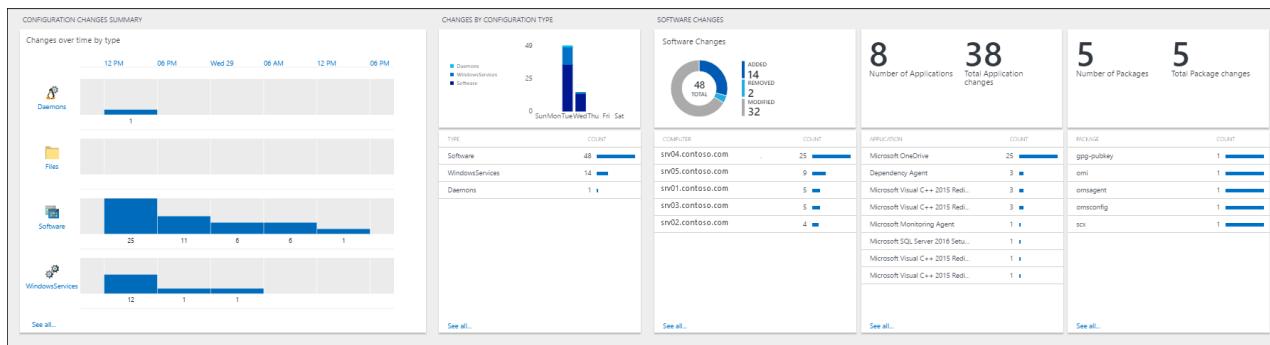


In order to analyze data outside of Log Analytics, you can export the data from the OMS repository into tools such as [Power BI](#) or Excel. You can also leverage the [Log Search API](#) to build custom solutions that leverage Log Analytics data or to integrate with other systems.

Add functionality with management solutions

[Management solutions](#) add functionality to OMS, providing additional data and analysis tools to Log Analytics. They may also define new record types to be collected that can be analyzed with Log Searches or by additional user

interface provided by the solution in the dashboard. The example image below shows the [Change Tracking](#) solution



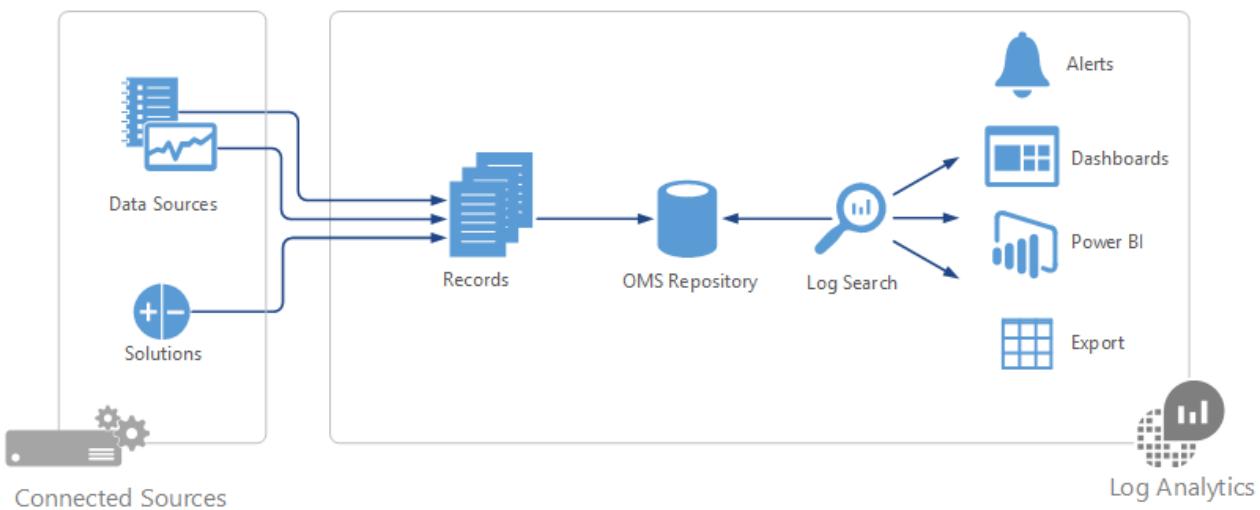
Solutions are available for a variety of functions, and additional solutions are consistently being added. You can easily browse available solutions and [add them to your OMS workspace](#) from the Solutions Gallery or Azure Marketplace. Many will be automatically deployed and start working immediately while others may require moderate configuration.

Solutions Gallery

App Dependency Monitor Coming Soon Automatically discover and map servers and their dependencies in real-time.	Malware Assessment Owned View status of antivirus and antimalware scans across your servers.	Containers Coming Soon See Docker container performance metrics and logs from containers across your public or private cloud environments.	Network Performance Monitor Coming Soon Offers near real time monitoring of network performance parameters like loss and latency.	Security and Audit Owned Provides the ability to explore security related data and helps identify security breaches.	System Update Assessment Owned Identify missing system updates across your servers.	AD Replication Status Owned Identify Active Directory replication issues in your environment.	Malware Assessment Owned View status of antivirus and antimalware scans across your servers.
Azure Networking Analytics Coming Soon Gain insight into your Azure Network data	Security and Audit Owned Provides the ability to explore security related data and helps identify security breaches.	Wire Data Coming Soon Provides the ability to explore wire data and helps identify network related issues.	Office 365 Coming Soon Get full visibility into your Office 365 user activities, perform forensics as well as audit and compliance.	SQL Assessment Free Assess the risk and health of SQL Server environments.	AD Assessment Owned Assess the risk and health of Active Directory environments.	Alert Management Owned View your Operations Manager and OMS alerts to easily triage alerts and identify the root causes of problems in your environment.	Automation Owned Automate time consuming and frequently repeated tasks in the cloud and on-premises.

Log Analytics components

At the center of Log Analytics is the OMS repository which is hosted in the Azure cloud. Data is collected into the repository from connected sources by configuring data sources and adding solutions to your subscription. Data sources and solutions will each create different record types that have their own set of properties but may still be analyzed together in queries to the repository. This allows you to use the same tools and methods to work with different kinds of data collected by different sources.



Connected sources are the computers and other resources that generate data collected by Log Analytics. This can include agents installed on [Windows](#) and [Linux](#) computers that connect directly or agents in a [connected System Center Operations Manager management group](#). For Azure resources, Log Analytics collects data from [Azure Monitor](#) and [Azure Diagnostics](#).

[Data sources](#) are the different kinds of data collected from each connected source. This includes [events](#) and [performance data](#) from [Windows](#) and Linux agents in addition to sources such as [IIS logs](#), and [custom text logs](#). You configure each data source that you want to collect, and the configuration is automatically delivered to each connected source.

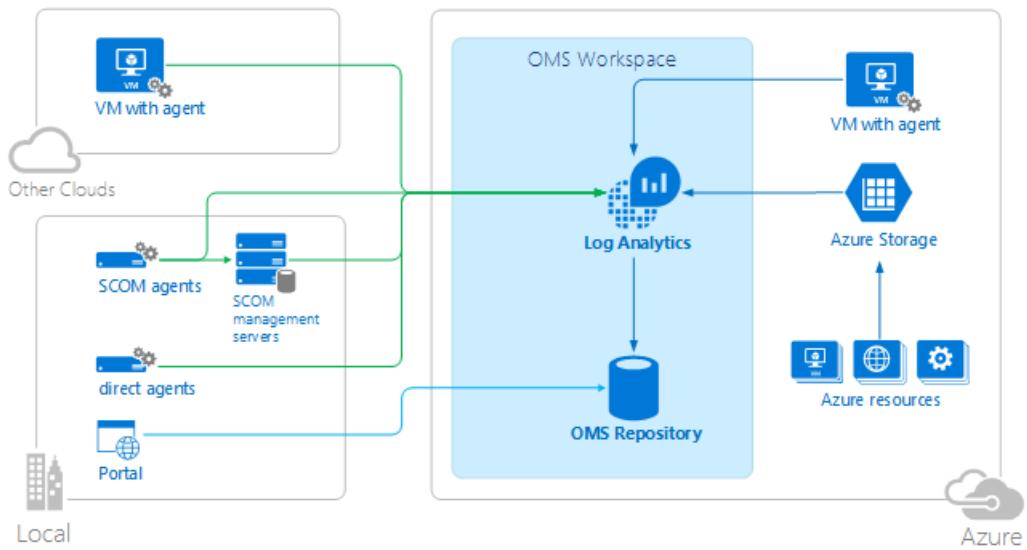
If you have custom requirements, then you can use the [HTTP Data Collector API](#) to write data to the repository from a REST API client.

Log Analytics architecture

The deployment requirements of Log Analytics are minimal since the central components are hosted in the Azure cloud. This includes the repository in addition to the services that allow you to correlate and analyze collected data. The portal can be accessed from any browser so there is no requirement for client software.

You must install agents on [Windows](#) and [Linux](#) computers, but there is no additional agent required for computers that are already members of a [connected SCOM management group](#). SCOM agents will continue to communicate with management servers which will forward their data to Log Analytics. Some solutions though will require agents to communicate directly with Log Analytics. The documentation for each solution will specify its communication requirements.

When you [sign up for Log Analytics](#), you will create an OMS workspace. You can think of the workspace as a unique Log Analytics environment with its own data repository, data sources, and solutions. You may create multiple workspaces in your subscription to support multiple environments such as production and test.



Next steps

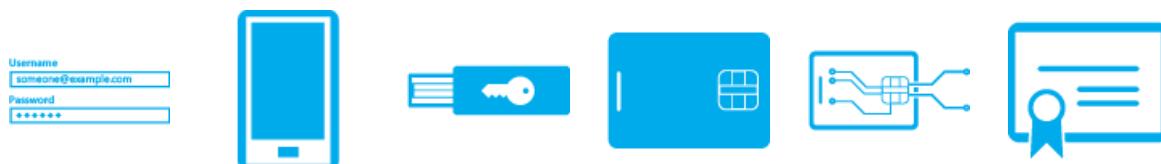
- [Sign up for a free Log Analytics account](#) to test in your own environment.
- View the different [Data Sources](#) available to collect data into the OMS repository.
- [Browse the available solutions in the Solutions Gallery](#) to add functionality to Log Analytics.

What is Azure Multi-Factor Authentication?

9/13/2017 • 1 min to read • [Edit Online](#)

Two-step verification is a method of authentication that requires more than one verification method and adds a critical second layer of security to user sign-ins and transactions. It works by requiring any two or more of the following verification methods:

- Something you know (typically a password)
- Something you have (a trusted device that is not easily duplicated, like a phone)
- Something you are (biometrics)



Azure Multi-Factor Authentication (MFA) is Microsoft's two-step verification solution. Azure MFA helps safeguard access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication via a range of verification methods, including phone call, text message, or mobile app verification.

Why use Azure Multi-Factor Authentication?

Today, more than ever, people are increasingly connected. With smart phones, tablets, laptops, and PCs, people have several different options on how they are going to connect and stay connected at any time. People can access their accounts and applications from anywhere, which means that they can get more work done and serve their customers better.

Azure Multi-Factor Authentication is an easy to use, scalable, and reliable solution that provides a second method of authentication so your users are always protected.

Easy to use	Scalable	Always Protected	Reliable

- **Easy to Use** - Azure Multi-Factor Authentication is simple to set up and use. The extra protection that comes with Azure Multi-Factor Authentication allows users to manage their own devices. Best of all, in many instances it can be set up with just a few simple clicks.
- **Scalable** - Azure Multi-Factor Authentication uses the power of the cloud and integrates with your on-premises AD and custom apps. This protection is even extended to your high-volume, mission-critical

scenarios.

- **Always Protected** - Azure Multi-Factor Authentication provides strong authentication using the highest industry standards.
- **Reliable** - We guarantee 99.9% availability of Azure Multi-Factor Authentication. The service is considered unavailable when it is unable to receive or process verification requests for the two-step verification.

Next steps

- Learn about [how Azure Multi-Factor Authentication works](#)
- Read about the different [versions and consumption methods for Azure Multi-Factor Authentication](#)

What is Azure Active Directory?

7/26/2017 • 5 min to read • [Edit Online](#)

Azure Active Directory (Azure AD) is Microsoft's multi-tenant, cloud based directory and identity management service. Azure AD combines core directory services, advanced identity governance, and application access management. Azure AD also offers a rich, standards-based platform that enables developers to deliver access control to their applications, based on centralized policy and rules.

For IT Admins, Azure AD provides an affordable, easy to use solution to give employees and business partners single sign-on (SSO) access to [thousands of cloud SaaS Applications](#) like Office365, Salesforce.com, DropBox, and Concur.

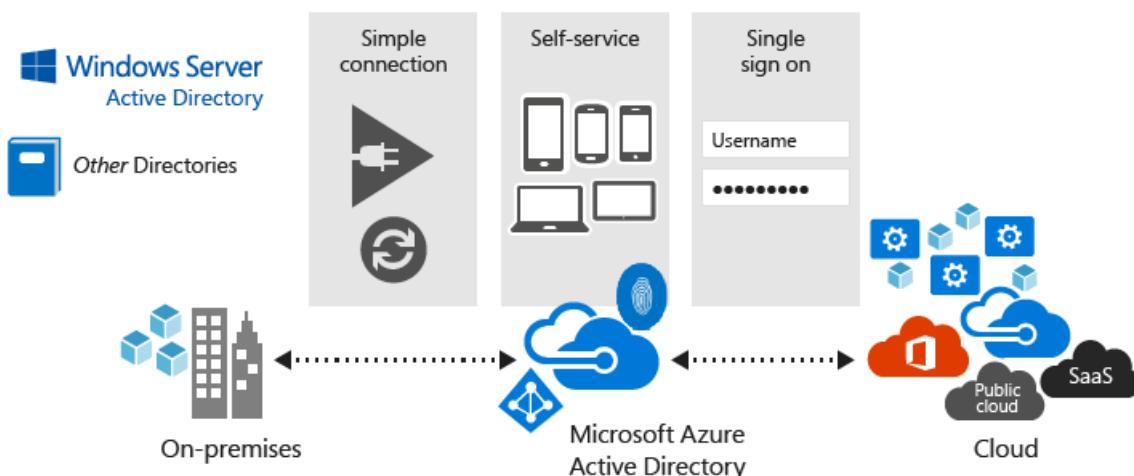
For application developers, Azure AD lets you focus on building your application by making it fast and simple to integrate with a world class identity management solution used by millions of organizations around the world.

Azure AD also includes a full suite of identity management capabilities including multi-factor authentication, device registration, self-service password management, self-service group management, privileged account management, role based access control, application usage monitoring, rich auditing and security monitoring and alerting. These capabilities can help secure cloud based applications, streamline IT processes, cut costs and help ensure that corporate compliance goals are met.

Additionally, with just [four clicks](#), Azure AD can be integrated with an existing Windows Server Active Directory, giving organizations the ability to leverage their existing on-premises identity investments to manage access to cloud based SaaS applications.

If you are an Office 365, Azure or Dynamics CRM Online customer, you might not realize that you are already using Azure AD. Every Office 365, Azure and Dynamics CRM tenant is actually already an Azure AD tenant.

Whenever you want you can start using that tenant to manage access to thousands of other cloud applications Azure AD integrates with!



How reliable is Azure AD?

The multi-tenant, geo-distributed, high availability design of Azure AD means that you can rely on it for your most critical business needs. Running out of 28 data centers around the world with automated failover, you'll have the comfort of knowing that Azure AD is highly reliable and that even if a data center goes down, copies of your directory data are live in at least two more regionally dispersed data centers and available for instant access.

For more details, see [Service Level Agreements](#).

Choose an edition

All Microsoft Online business services rely on Azure Active Directory (Azure AD) for sign-in and other identity needs. If you subscribe to any of Microsoft Online business services (for example, Office 365 or Microsoft Azure), you get Azure AD with access to all of the Free features. With the Azure Active Directory Free edition, you can manage users and groups, synchronize with on-premises directories, get single sign-on across Azure, Office 365, and thousands of popular SaaS applications like Salesforce, Workday, Concur, DocuSign, Google Apps, Box, ServiceNow, Dropbox, and more.

To enhance your Azure Active Directory, you can add paid capabilities using the Azure Active Directory Basic, Premium P1, and Premium P2 editions. Azure Active Directory paid editions are built on top of your existing free directory, providing enterprise class capabilities spanning self-service, enhanced monitoring, security reporting, Multi-Factor Authentication (MFA), and secure access for your mobile workforce.

NOTE

For the pricing options of these editions, see [Azure Active Directory Pricing](#). Azure Active Directory Premium P1, Premium P2, and Azure Active Directory Basic are not currently supported in China. Please contact us at the Azure Active Directory Forum for more information.

- **Azure Active Directory Basic** - Designed for task workers with cloud-first needs, this edition provides cloud centric application access and self-service identity management solutions. With the Basic edition of Azure Active Directory, you get productivity enhancing and cost reducing features like group-based access management, self-service password reset for cloud applications, and Azure Active Directory Application Proxy (to publish on-premises web applications using Azure Active Directory), all backed by an enterprise-level SLA of 99.9 percent uptime.
- **Azure Active Directory Premium P1** - Designed to empower organizations with more demanding identity and access management needs, Azure Active Directory Premium edition adds feature-rich enterprise-level identity management capabilities and enables hybrid users to seamlessly access on-premises and cloud capabilities. This edition includes everything you need for information worker and identity administrators in hybrid environments across application access, self-service identity and access management (IAM), identity protection and security in the cloud. It supports advanced administration and delegation resources like dynamic groups and self-service group management. It includes Microsoft Identity Manager (an on-premises identity and access management suite) and provides cloud write-back capabilities enabling solutions like self-service password reset for your on-premises users.
- **Azure Active Directory Premium P2** - Designed with advanced protection for all your users and administrators, this new offering includes all the capabilities in Azure AD Premium P1 as well as our new Identity Protection and Privileged Identity Management. Azure Active Directory Identity Protection leverages billions of signals to provide risk-based conditional access to your applications and critical company data. We also help you manage and protect privileged accounts with Azure Active Directory Privileged Identity Management so you can discover, restrict and monitor administrators and their access to resources and provide just-in-time access when needed.

NOTE

A number of Azure Active Directory capabilities are available through "pay as you go" editions:

- Active Directory B2C is the identity and access management solution for your consumer-facing applications. For more details, see [Azure Active Directory B2C](#)
- Azure Multi-Factor Authentication can be used through per user or per authentication providers. For more details, see [What is Azure Multi-Factor Authentication?](#)

How can I get started?

If you are an IT admin:

- [Try it out!](#) - you can sign up for a free 30 day trial today and deploy your first cloud solution in under 5 minutes using this link
- Read [Getting started with Azure AD](#) for tips and tricks on getting an Azure AD tenant up and running fast

If you are a developer:

- Check out our [Developers Guide](#) to Azure Active Directory
- [Start a trial](#) – sign up for a free 30 day trial today and start integrating your apps with Azure AD

Next steps

[Learn more about the fundamentals of Azure identity and access management](#)

Getting started with Operations Management Suite Security and Audit Solution

7/28/2017 • 11 min to read • [Edit Online](#)

This document helps you get started quickly with Operations Management Suite (OMS) Security and Audit solution capabilities by guiding you through each option.

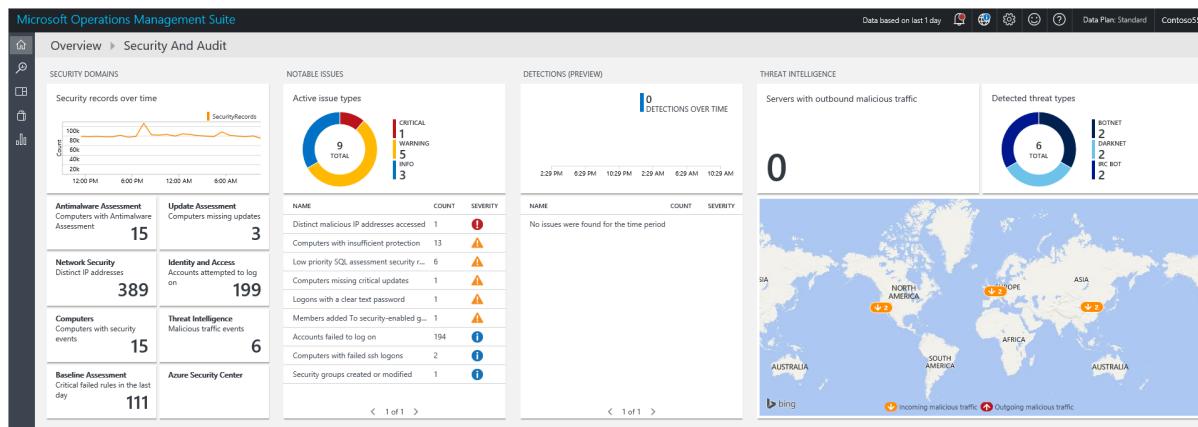
What is OMS?

Microsoft Operations Management Suite (OMS) is Microsoft's cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure. For more information about OMS, read the article [Operations Management Suite](#).

OMS Security and Audit dashboard

The OMS Security and Audit solution provides a comprehensive view into your organization's IT security posture with built-in search queries for notable issues that require your attention. The **Security and Audit** dashboard is the home screen for everything related to security in OMS. It provides high-level insight into the security state of your computers. It also includes the ability to view all events from the past 24 hours, 7 days, or any other custom time frame. To access the **Security and Audit** dashboard, follow these steps:

1. In the **Microsoft Operations Management Suite** main dashboard click **Settings** tile in the left.
2. In the **Settings** blade, under **Solutions** click **Security and Audit** option.
3. The **Security and Audit** dashboard appears:



If you are accessing this dashboard for the first time and you don't have devices monitored by OMS, the tiles will not be populated with data obtained from the agent. Once you install the agent, it can take some time to populate, therefore what you see initially may be missing some data as they are still uploading to the cloud. In this case, it is normal to see some tiles without tangible information. Read [Connect Windows computers directly to OMS](#) for more information on how to install OMS agent in a Windows system and [Connect Linux computers to OMS](#) for more information on how to perform this task in a Linux system.

NOTE

The agent collects the information based on the current events that are enabled, for instance computer name, IP address and user name. However no document/files, database name or private data are collected.

Solutions are a collection of logic, visualization, and data acquisition rules that address key customer challenges. Security and Audit is one solution, others can be added separately. Read the article [Add solutions](#) for more information on how to add a new solution.

The OMS Security and Audit dashboard is organized in four major categories:

- **Security Domains:** in this area you will be able to further explore security records over time, access malware assessment, update assessment, network security, identity and access information, computers with security events and quickly have access to Azure Security Center dashboard.
- **Notable Issues:** this option will allow you to quickly identify the number of active issues and the severity of these issues.
- **Detections (Preview):** enables you to identify attack patterns by visualizing security alerts as they take place against your resources.
- **Threat Intelligence:** enables you to identify attack patterns by visualizing the total number of servers with outbound malicious IP traffic, the malicious threat type and a map that shows where these IPs are coming from.
- **Common security queries:** this option provides you a list of the most common security queries that you can use to monitor your environment. When you click in one of those queries, it opens the **Search** blade with the results for that query.

NOTE

for more information on how OMS keeps your data secure, read [How OMS secures your data](#).

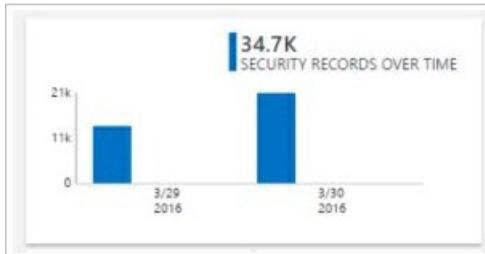
Security domains

When monitoring resources, it is important to be able to quickly access the current state of your environment. However it is also important to be able to track back events that occurred in the past that can lead to a better understanding of what's happening in your environment at certain point in time.

NOTE

data retention is according to the OMS pricing plan. For more information visit the [Microsoft Operations Management Suite pricing page](#).

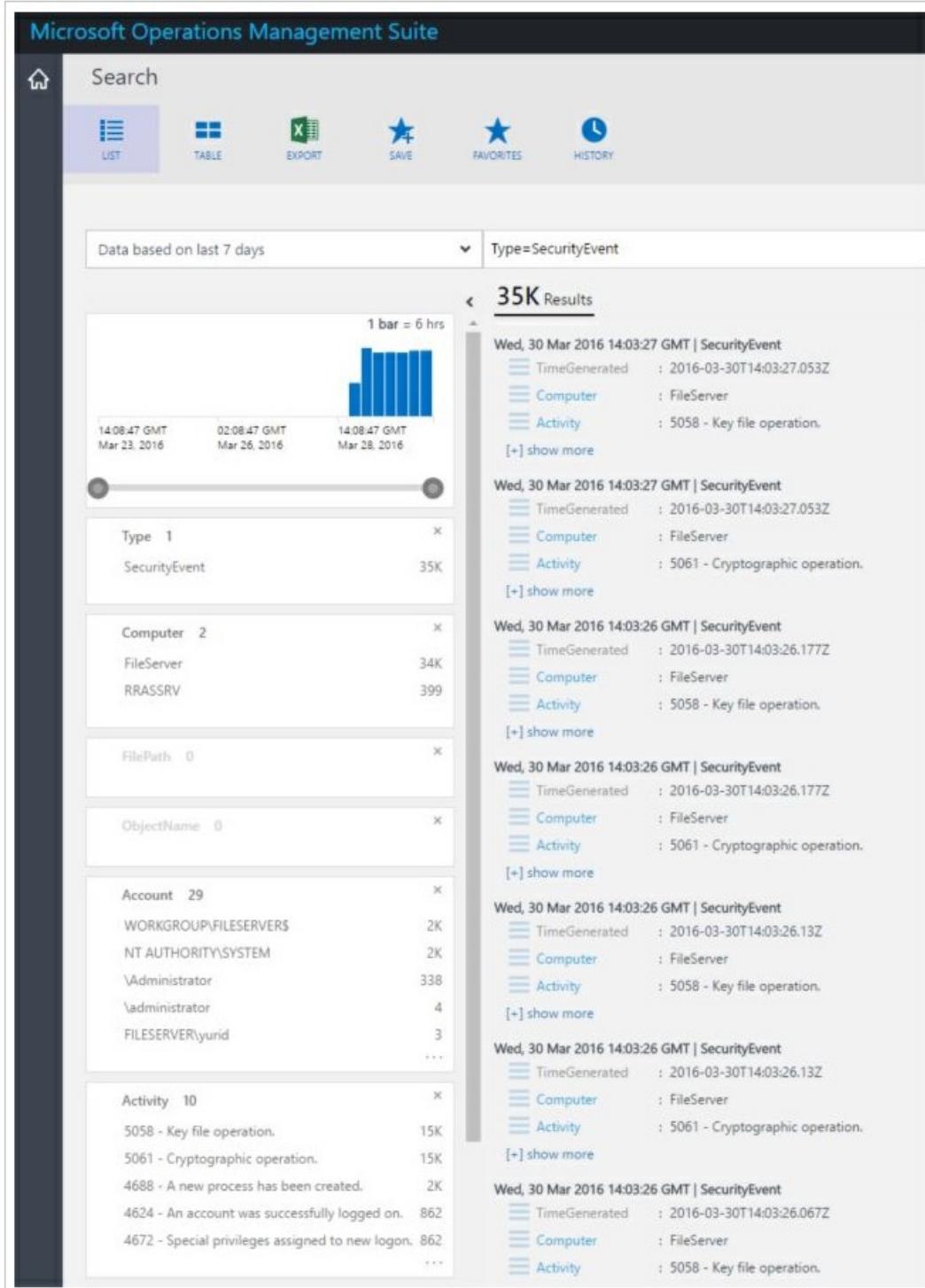
Incident response and forensics investigation scenarios will directly benefit from the results available in the **Security Records over Time** tile.



When you click on this tile, the **Search** blade will open, showing a query result for **Security Events** (Type=SecurityEvents) with data based on the last seven days, as shown below:

NOTE

If your workspace has been upgraded to the [new Log Analytics query language](#), then the following queries need to be converted. You can use the [language converter](#) to perform this translation.



The search result is divided in two panes: the left pane gives you a breakdown of the number of security events that were found, the computers in which these events were found, the number of accounts that were discovered in these computers and the types of activities. The right pane provides you the total results and a chronological view of the security events with the computer's name and event activity. You can also click **Show More** to view more details about this event, such as the event data, the event ID and the event source.

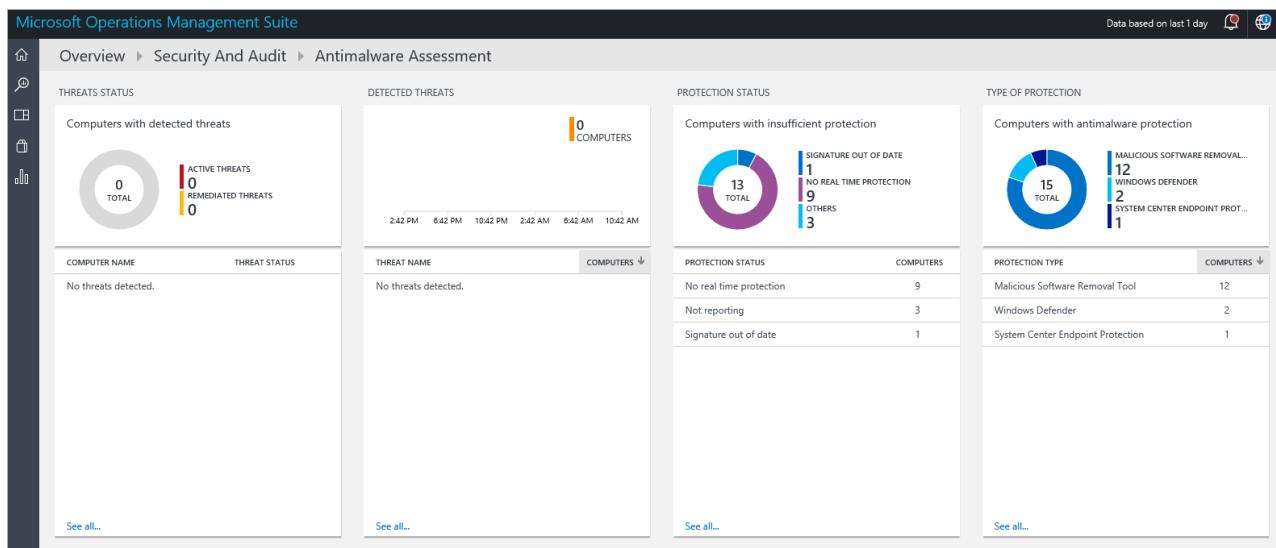
NOTE

for more information about OMS search query, read [OMS search reference](#).

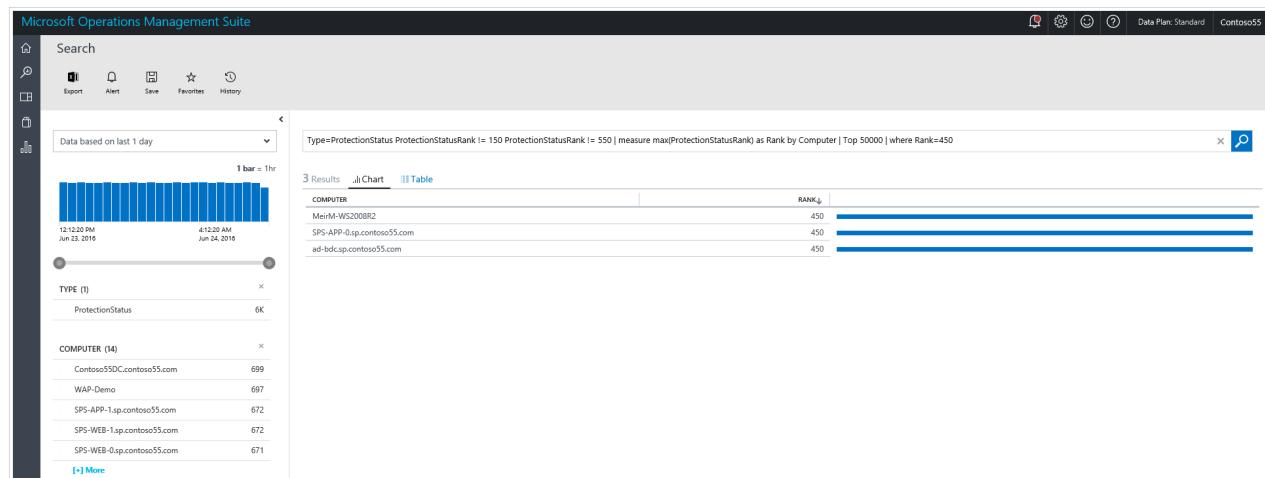
Antimalware assessment

This option enables you to quickly identify computers with insufficient protection and computers that are compromised by a piece of malware. Malware assessment status and detected threats on the monitored servers

are read, and then the data is sent to the OMS service in the cloud for processing. Servers with detected threats and servers with insufficient protection are shown in the malware assessment dashboard, which is accessible after you click in the **Antimalware Assessment** tile.



Just like any other live tile available in OMS Dashboard, when you click on it, the **Search** blade will open with the query result. For this option, if you click in the **Not Reporting** option under **Protection Status**, you will have the query result that shows this single entry that contains the computer's name and its rank, as shown below:



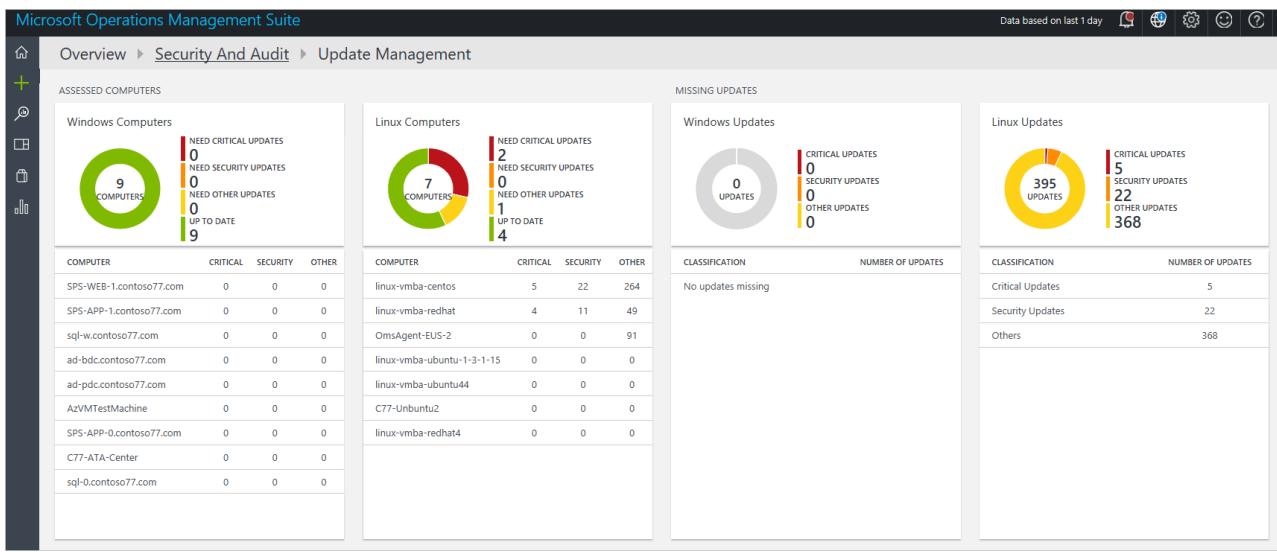
NOTE

rank is a grade giving to reflect the status of the protection (on, off, updated, etc.) and threats that are found. Having that as a number helps to make aggregations.

If you click in the computer's name, you will have the chronological view of the protection status for this computer. This is very useful for scenarios in which you need to understand if the antimalware was once installed and at some point it was removed.

Update assessment

This option enables you to quickly determine the overall exposure to potential security problems, and whether or how critical these updates are for your environment. OMS Security and Audit solution only provide the visualization of these updates, the real data comes from [Update Management Solutions](#), which is a different module within OMS. Here an example of the updates:



NOTE

For more information about Update Management solution, read [Update Management solution in OMS](#).

Identity and Access

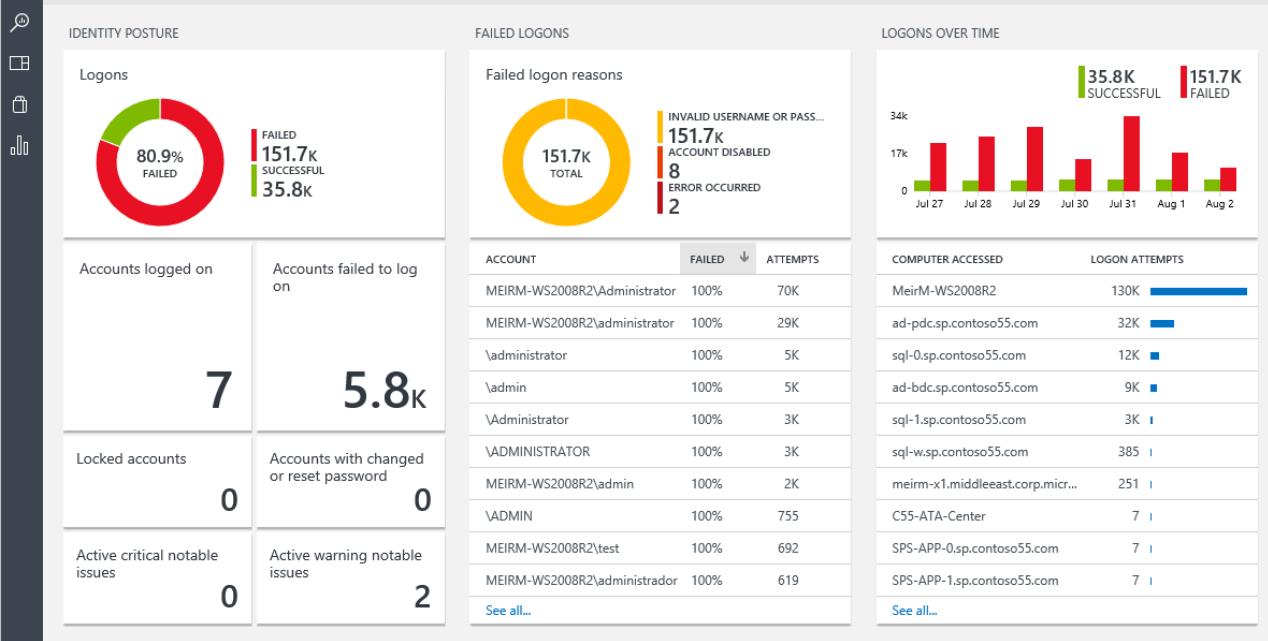
Identity should be the control plane for your enterprise, protecting your identity should be your top priority. While in the past there were perimeters around organizations and those perimeters were one of the primary defensive boundaries, nowadays with more data and more apps moving to the cloud the identity becomes the new perimeter.

NOTE

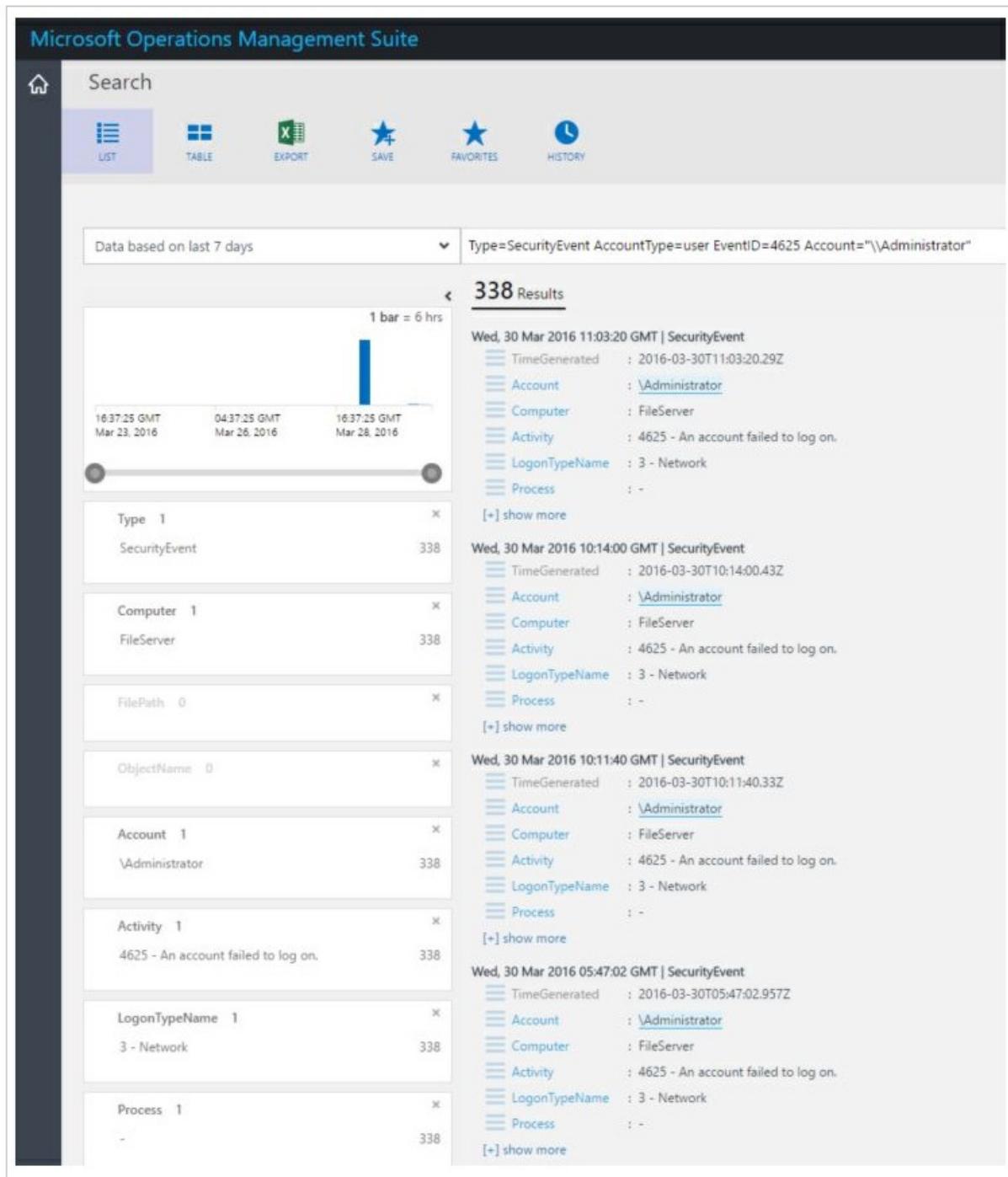
currently the data is based only on Security Events login data (event ID 4624) in the future Office365 logins and Azure AD data will also be included.

By monitoring your identity activities you will be able to take proactive actions before an incident takes place or reactive actions to stop an attack attempt. The **Identity and Access** dashboard provides you an overview of your identity state, including the number of failed attempts to log on, the user's account that were used during those attempts, accounts that were locked out, accounts with changed or reset password and currently number of accounts that are logged in.

When you click in the **Identity and Access** tile you will see the following dashboard:



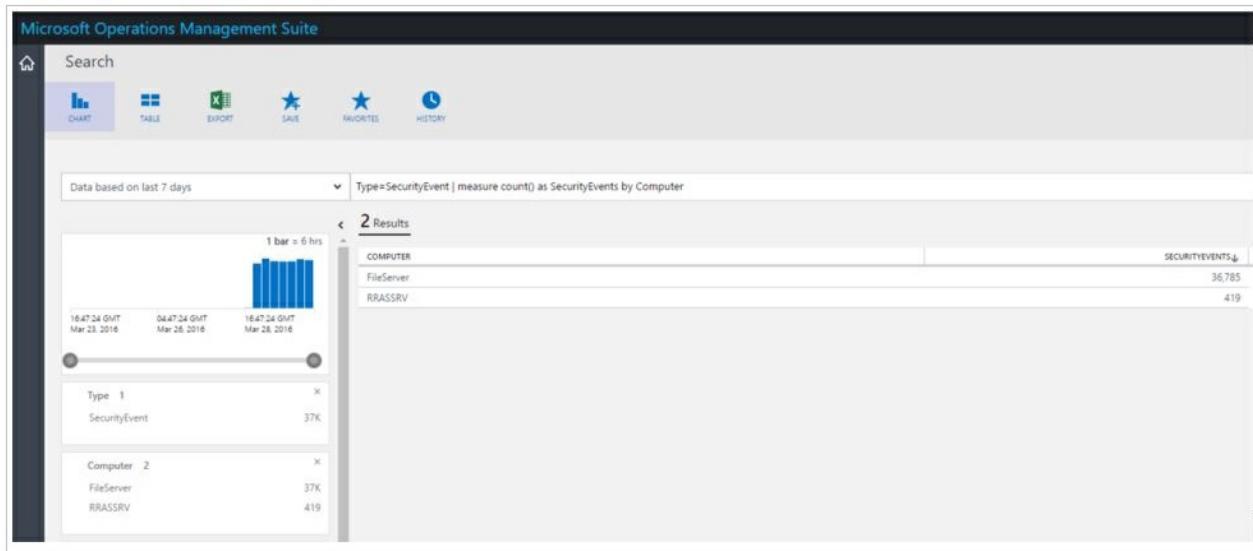
The information available in this dashboard can immediately assist you to identify a potential suspicious activity. For example, there are 338 attempts to log on as **Administrator** and 100% of these attempts failed. This can be caused by a brute force attack against this account. If you click on this account you will obtain more information that can assist you to determine the target resource for this potential attack:



The detailed report provides important information about this event, including: the target computer, the type of logon (in this case Network logon), the activity (in this case event 4625) and a comprehensive timeline of each attempt.

Computers

This tile can be used to access all computers that actively have security events. When you click in this tile you will see the list of computers with security events and the number of events on each computer:

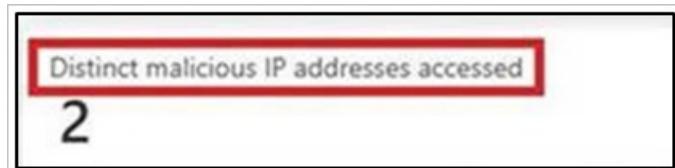


You can continue your investigation by clicking on each computer and review the security events that were flagged.

Threat Intelligence

By using the Threat Intelligence option available in OMS Security and Audit, IT administrators can identify security threats against the environment, for example, identify if a particular computer is part of a botnet. Computers can become nodes in a botnet when attackers illicitly install malware that secretly connects this computer to the command and control. It can also identify potential threats coming from underground communication channels, such as darknet. Learn more about Threat Intelligence by reading [Monitoring and responding to security alerts in Operations Management Suite Security and Audit Solution](#) article.

In some scenarios, you may notice a potential malicious IP that was accessed from one monitored computer:



This alert and others within the same category, are generated via OMS Security by leveraging [Microsoft Threat Intelligence](#). The Threat Intelligence data is collected by Microsoft as well as purchased from leading threat intelligence providers. This data is updated frequently and adapted to fast-moving threats. Due to its nature, it should be combined with other sources of security information while [investigating](#) a security alert.

Baseline Assessment

Microsoft, together with industry and government organizations worldwide, defines a Windows configuration that represents highly secure server deployments. This configuration is a set of registry keys, audit policy settings, and security policy settings along with Microsoft's recommended values for these settings. This set of rules is known as Security baseline. Read [Baseline Assessment in Operations Management Suite Security and Audit Solution](#) for more information about this option.

Azure Security Center

This tile is basically a shortcut to access Azure Security Center dashboard. Read [Getting started with Azure Security Center](#) for more information about this solution.

Notable issues

The main intent of this group of options is to provide a quick view of the issues that you have in your environment, by categorizing them in Critical, Warning and Informational. The Active issue type tile it's a visualization of these issues, but it doesn't allow you to explore more details about them, for that you need to use the lower part of this tile that has the name of the issue (NAME), how many objects had this happen (COUNT) and how critical it is

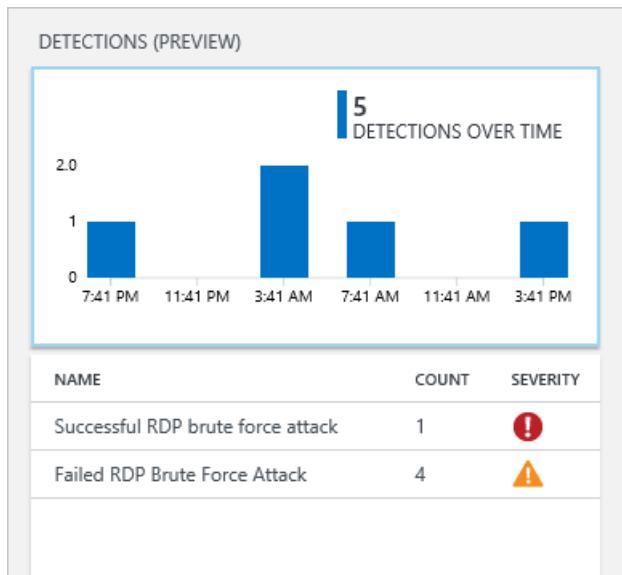
(SEVERITY).



You can see that these issues were already covered in different areas of the **Security Domains** group, which reinforces the intent of this view: visualize the most important issues in your environment from a single place.

Detections (Preview)

The main intent of this option is to allow IT to quickly identify potential threats to their environment via and the severity of this threat.



This option can also be used during an [incident response investigation](#) to perform the assessment and obtain more information about the attack.

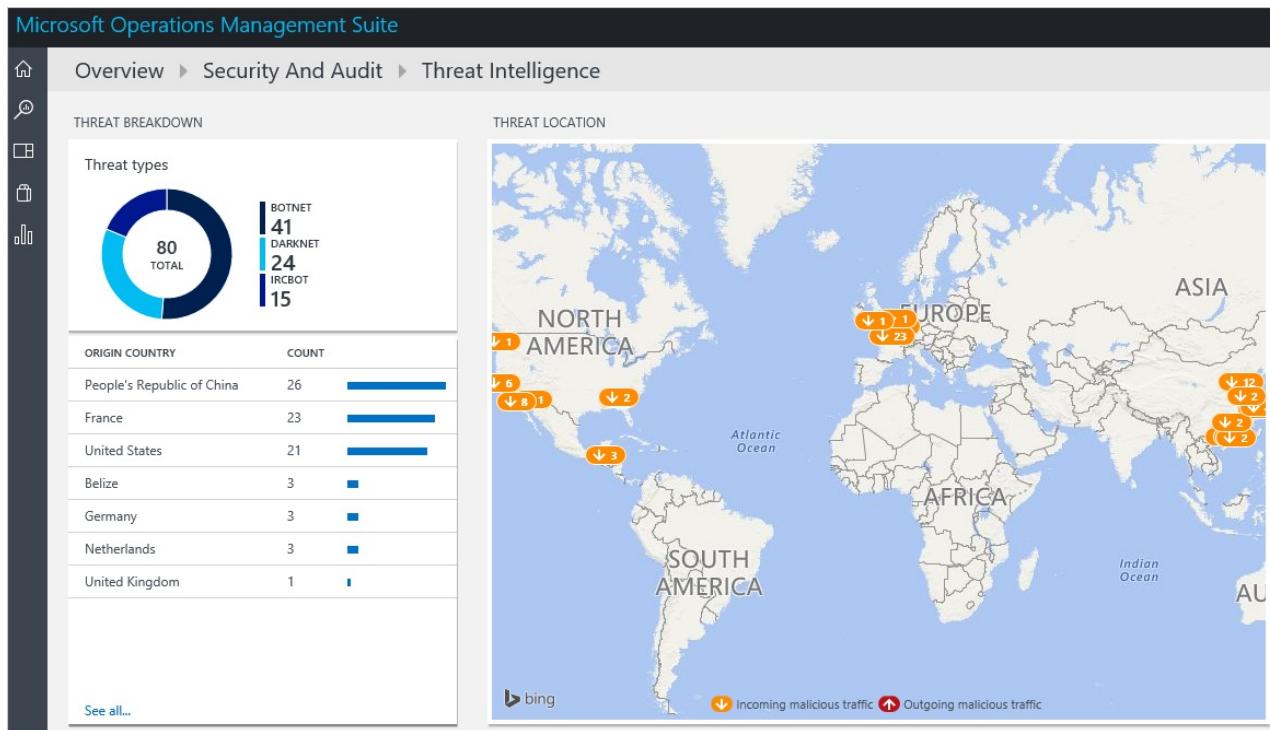
NOTE

For more information on how to use OMS for Incident Response, watch this video: [How to Leverage the Azure Security Center & Microsoft Operations Management Suite for an Incident Response](#).

Threat Intelligence

The new threat intelligence section of the Security and Audit solution visualizes the possible attack patterns in several ways: the total number of servers with outbound malicious IP traffic, the malicious threat type and a map that shows where these IPs are coming from. You can interact with the map and click on the IPs for more information.

Yellow pushpins on the map indicate incoming traffic from malicious IPs. It is not uncommon for servers that are exposed to the internet to see incoming malicious traffic, but we recommend reviewing these attempts to make sure none of them was successful. These indicators are based on IIS logs, WireData and Windows Firewall logs.



Common security queries

The list of common security queries available can be useful for you to rapidly access resource's information and customize it based on your environment's needs. These common queries are:

- All Security Activities
- Security Activities on the computer "computer01.contoso.com" (replace with your own computer name)
- Security Activities on the computer "computer01.contoso.com" for account "Administrator" (replace with your own computer and account names)
- Log on Activity by Computer
- Accounts who terminated Microsoft antimalware on any computer
- Computers where the Microsoft antimalware process was terminated
- Computers where "hash.exe" was executed (replace with different process name)
- All Process names that were executed
- Log on Activity by Account
- Accounts who remotely logged on the computer "computer01.contoso.com" (replace with your own computer name)

See also

In this document, you were introduced to OMS Security and Audit solution. To learn more about OMS Security, see the following articles:

- [Operations Management Suite \(OMS\) overview](#)
- [Monitoring and Responding to Security Alerts in Operations Management Suite Security and Audit Solution](#)
- [Monitoring Resources in Operations Management Suite Security and Audit Solution](#)

Azure Security MVP Program

6/27/2017 • 1 min to read • [Edit Online](#)

Microsoft Most Valuable Professionals (MVPs) are community leaders who've demonstrated an exemplary commitment to helping others get the most out of their experience with Microsoft technologies. They share their exceptional passion, real-world knowledge, and technical expertise with the community and with Microsoft.

We are happy to announce that Microsoft Azure now recognizes community experts with special expertise in Azure security. Microsoft MVPs can be awarded the MVP in Microsoft Azure in the Azure Security contribution area.

Award Category	Microsoft Azure	Windows Development	Office Development	Visual Studio and Development Technologies	Data Platform
Contribution Areas	<ul style="list-style-type: none">• Azure App Service• Azure Media Service & CDN• IoT on Azure and Azure Messaging (Event Hub and Service Bus)• Azure Cloud Service• Azure Service Fabric• Application Integration• Azure Virtual Machines (IaaS) and Batch• Azure Storage• Azure Networking• Azure Backup & Recovery• Azure Security• Linux and Docker on Azure• DevOps on Azure (Chef, Puppet, Salt, Ansible, Dev/Test Lab)• SDK support on Azure (.NET, Node.js, Java, PHP, Python, GO, Ruby)	<ul style="list-style-type: none">• Windows App Development• Classic Windows Development• Windows Bridges• Windows On Devices (IoT /Embedded)• Windows Hardware Engineering• Emerging Experiences (More Personal Computing)	<ul style="list-style-type: none">• Office Add-in Development• O365 API Development• SharePoint Add-in Development• Office Development for iOS• Office Development for Android• Office Development with PHP• Office Development with Node.js• Office Development with Angularjs	<ul style="list-style-type: none">• ASP.NET/IIS• .NET• Visual C++• Visual Studio ALM• Developer Security• Visual Studio Extensibility• Front End Web Dev• Node.js• PHP• Python• Java• Unity• Xamarin• Cordova• JavaScript/TypeScript• Grunt/Gulp• CSS3• Clang/LLVM	<ul style="list-style-type: none">• Analytics Platform System• Azure Data Lake• Azure DocumentDB• Azure HDInsight and Hadoop, Spark, & Storm on Azure• Azure Machine Learning• Azure Search• Azure SQL Data Warehouse• Azure SQL Database• Azure Stream Analytics• Cortana Analytics Suite• Information Management (ADF, SSIS, & Data Sync)• Power BI• SQL Server• SQL Server Reporting Services & Analysis Services

While there is no benchmark for becoming an MVP, in part because it varies by technology and its life-cycle, some of the criteria we evaluate include the impact of a nominee's contributions to online forums such as Microsoft Answers, TechNet and MSDN; wikis and online content; conferences and user groups; podcasts, Web sites, blogs and social media; and articles and books.

Are you an expert in Azure security? Do you know someone who is? Then [Nominate yourself or someone else](#) to become an Azure security MVP today!

Microsoft Services in Cybersecurity

6/27/2017 • 1 min to read • [Edit Online](#)

Microsoft Services provides a comprehensive approach to security, identity and cybersecurity. Microsoft Services provides an array of Security and Identity services across strategy, planning, implementation, and ongoing support which can help our Enterprise customers implement holistic security solutions that align with their strategic goals.

With direct access to product development teams, we can create solutions that integrate, and enhance the latest security and identity capabilities of our products to help protect our customer's business and drive innovation.

Entrusted with helping protect and enable the world's largest organizations, our diverse group of technical professionals consists of highly trained experts who offer a wealth of security and identity experience.

Learn more about services provided by Microsoft Services:

- [Security Risk Assessment](#)
- [Dynamic Identity Framework Assessment](#)
- [Offline Assessment for Active Directory Services](#)
- [Enhanced Security Administration Environment](#)
- [Azure AD Implementation Services](#)
- [Securing Against Lateral Account Movement](#)
- [Microsoft Threat Detection Services](#)
- [Incident Response and Recovery](#)

[Learn more](#) about Microsoft Services Security consulting services.

Manage personal data in Microsoft Azure

8/30/2017 • 4 min to read • [Edit Online](#)

This article provides guidance on how to correct, update, delete, and export personal data in Azure Active Directory and Azure SQL Database.

Scenario

A Dublin-based company provides one-stop shopping for high end destination weddings in Ireland and around the world for both a local and international customer base. They have offices, customers, employees, and vendors located around the world to fully service the venues they offer.

Among many other items, the company keeps track of RSVPs that include food allergies and dietary preferences. Wedding guests can register for various activities such as horseback riding, surfing, boat rides, etc., and even interact with one another on a central web page during the months leading up to the event. The company collects personal information from employees, vendors, customers, and wedding guests. Because of the international nature of the business the company must comply with multiple levels of regulation.

Problem statement

- Data admins must be able to correct inaccurate personal information and update incomplete or changing personal information.
- Data admins must be able to delete personal information upon the request of a data subject.
- Data admins need to export data and provide it to a data subject in a common, structured format upon his or her request.

Company goals

- Inaccurate or incomplete customer, wedding guest, employee, and vendor personal information must be corrected or updated in Azure Active Directory and Azure SQL Database.
- Personal information must be deleted in Azure Active Directory and Azure SQL Database upon the request of a data subject.
- Personal data must be exported in a common, structured format upon the request of a data subject.

Solutions

Azure Active Directory: rectify/correct inaccurate or incomplete personal data and erase/delete personal data/user profiles

[Azure Active Directory](#) is Microsoft's cloud-based, multi-tenant directory and identity management service. You can correct, update, or delete customer and employee user profiles and user work information that contain personal data, such as a user's name, work title, address, or phone number, in your [Azure Active Directory](#) (AAD) environment by using the [Azure portal](#).

You must sign in with an account that's a global admin for the directory.

How do I correct or update user profile and work information in Azure Active Directory?

1. Sign in to the [Azure portal](#) with an account that's a global admin for the directory.
2. Select **More services**, enter **Users and groups** in the text box, and then select **Enter**.

The screenshot shows the Microsoft Azure portal interface. On the left, there's a dark sidebar with various service icons and names. A red box highlights the 'More services >' option at the bottom of this sidebar. In the center, a search bar at the top has 'user' typed into it. Below the search bar, a red box highlights the 'Users and groups' link under the 'PREVIEW' section. The main content area shows a message 'No resources to display' and a 'Help + support' button.

3. On the **Users and groups** blade, select **Users**.

The screenshot shows the 'User management - Users' blade. The left sidebar lists various management options like Resource groups, All resources, Recent, App Services, etc., with 'User management' highlighted. Under 'User management', 'Users' is selected and highlighted with a red box. The main content area shows a table of users with columns for NAME and USER NAME. Each user row has a three-dot ellipsis icon on the far right. The table lists approximately 20 users, each with a small profile picture and their respective email addresses.

NAME	USER NAME	...
Aaron Nicholls	tnich@f128.info	...
Abby Brennan	abrennan@f128.info	...
Adam Carlton	acarlton@f128.info	...
Adam Steenwyk	adam@f128.info	...
Adam Steenwyk	adam@f128.onmicrosoft.com	...
Adam Steenwyk (admin f128.onmicrosoft.com)	admin@f128.onmicrosoft.com	...
Adam Steenwyk (admin)	admin@f128.info	...
Adam Steenwyk (MSA)	ajamess_gmail.com#EXT#@f128.onmicrosoft.com	...
Adrian Martin	amartin@f128.info	...
Aidan Mitzner	amitzner@f128.info	...
Ajay Mokashi	amokashi@f128.info	...
Alain Dubois	adubois@f128.info	...
Alberto Bassani	abassani@f128.info	...
Alejandro Ruiz	aruiz@f128.info	...
Alex Grossman	agrossman@f128.info	...
Alice Gartner	alice@f128.info	...
Allison Hunter	ahunter@f128.info	...
Alvin Hwang	ahwang@f128.info	...
Amanda Baker	abaker@f128.info	...
Amanda Mackenzie	amackenzie@f128.info	...
Amelia Casias	acasias@f128.info	...

4. On the **Users and groups - Users** blade, select a user from the list, and then, on the blade for the selected user, select **Profile** to view the user profile information that needs to be corrected or updated.

The screenshot shows the Azure Active Directory User - Profile blade for a user named Angie Shannon. On the left, there is a list of users with their names and email addresses. On the right, the profile details for Angie Shannon are displayed. The 'Profile' tab is selected. The profile includes fields for Name (Angie Shannon), User name (angie@litwarecorp.com), First name, Last name, Photo (a placeholder blue person icon), and an 'Upload a new photo' button. There are also sections for Object ID (3beb06c5 687e 4ae5 9c6b bdcb438f33), Source (Azure Active Directory), Settings (Block sign in set to No), and Authentication contact info (Authentication phone and Alternate authentication phone fields).

NAME	USER NAME
Abbie Spencer	aspencer@litwarecorp.com
AJ	tperkins_f128.info#EXT#@litware154.onmicrosoft.com
Al Vanover	avanover@litwarecorp.com
Alfred Borrego	alfred@litwarecorp.com
Alfred Geer	ageer@litwarecorp.com
Althea Spears	althea@litwarecorp.com
Amanda Dunlap	amanda@litwarecorp.com
Andy Pettis	andy@litwarecorp.com
Angie Shannon	angie@litwarecorp.com
Annabelle Ballard	annabelle@litwarecorp.com
Anne-Marie Johansen	annemarie@litwarecorp.com
Anthony Philip	anthony@litwarecorp.com
Arthur Theriot	arthur@litwarecorp.com
Audrey Bradley	audrey@litwarecorp.com
Aurel Cuza	aurel@litwarecorp.com
Austin Jin	austin@litwarecorp.com
Bhaanulata Mokkapati	bhaanulata@litwarecorp.com
Bryce Ault	bryce@litwarecorp.com
Cecill Noll	cnoll@litwarecorp.com
Cezar Spirlea	cezar@litwarecorp.com
Daniel Hernandez	daniel.hernandez@litwarecorp.com

5. Correct or update the information, and then, in the command bar, select **Save**.
6. On the blade for the selected user, select **Work Info** to view user work information that needs to be corrected or updated.

7. Correct or update the user work information, and then, in the command bar, select **Save**.

How do I delete a user profile in Azure Active Directory?

1. Sign in to the [Azure portal](#) with an account that's a global admin for the directory.

2. Select **More services**, enter **Users and groups** in the text box, and then select **Enter**.

3. On the **Users and groups** blade, select **Users**.

The screenshot shows the Azure portal interface for User management. On the left sidebar, there's a list of resources including Resource groups, All resources, Recent, App Services, Virtual machines (classic), Virtual machines, SQL databases, Cloud services (classic), Security Center, Subscriptions, User management, and Enterprise applications. A 'New' button is also present. The main content area is titled 'User management - Users' and shows a preview for 'f128 Photography'. It has sections for GENERAL (Overview, Audit) and RESOURCES (Users, Groups). The 'Users' section is highlighted with a red box. The main list displays user details:

NAME	USER NAME
Aaron Nicholls	tnich@f128.info
Abby Brennan	abrennan@f128.info
Adam Carlton	acarlton@f128.info
Adam Steenwyk	adam@f128.info
Adam Steenwyk	adam@f128.onmicrosoft.com
Adam Steenwyk (admin f128.onmicrosoft.com)	admin@f128.onmicrosoft.com
Adam Steenwyk (admin)	admin@f128.info
Adam Steenwyk (MSA)	ajamess_gmail.com#EXT#@f128.onmicrosoft.com
Adrian Martin	amartin@f128.info
Aidan Mitzner	amitzner@f128.info
Ajay Mokashi	amokashi@f128.info
Alain DuBois	adubois@f128.info
Alberto Bassani	abassani@f128.info
Alejandro Ruiz	aruiz@f128.info
Alex Grossman	agrossman@f128.info
Alice Gartner	alice@f128.info
Allison Hunter	ahunter@f128.info
Alvin Hwang	ahwang@f128.info
Amanda Baker	abaker@f128.info
Amanda Mackenzie	amackenzie@f128.info
Amelia Casias	acasias@f128.info

4. On the **Users and groups - Users** blade, select a user from the list.

The screenshot shows the Azure portal interface. On the left, there's a search bar and a list of users with their names, email addresses, and profile icons. On the right, the 'User - Work Info' blade is open for 'Angie Shannon - PREVIEW'. The 'Work Info' section is highlighted with a red box. Other sections include General, Audit, Profile, Access, and Contact info.

- On the blade for the selected user, select **Overview**, and then in the command bar, select **Delete**.

The screenshot shows the Azure portal interface with two windows open. On the left, the 'User management - Users' blade is shown with a list of users. On the right, the 'User - Andy Pettis' blade is open, showing the user's details, sign-in history, and a summary card for Andy Pettis.

SQL Database: rectify/correct inaccurate or incomplete personal data; erase/delete personal data; export personal data

Azure SQL Database is a cloud database that helps developers build and maintain applications.

Personal data can be updated in [Azure SQL Database](#) using standard SQL queries, and it can also be deleted. Additionally, personal data can be exported from SQL Database using a variety of methods, including the Azure SQL Server import and export wizard, and in a variety of formats, including a BACPAC file.

How do I correct, update, or erase personal data in SQL Database?

To learn how to correct or update personal data in SQL Database, visit the [Update \(Transact-SQL\)](#), [Update Text](#), [Update with Common Table Expression](#), or [Update Write Text](#) documentation.

To learn how to delete personal data in SQL Database, visit the [Delete \(Transact-SQL\)](#) documentation.

How do I export personal data to a BACPAC file in SQL Database?

A BACPAC file includes the SQL Database data and metadata and is a zip file with a BACPAC extension. This can be done using the [Azure portal](#), the SQLPackage command-line utility, SQL Server Management Studio (SSMS), or PowerShell.

To learn how to export data to a BACPAC file, visit the [Export an Azure SQL database to a BACPAC file](#) page, which includes detailed instructions for each method listed above.

How do I export personal data from SQL Database with the SQL Server Import and Export Wizard?

This wizard helps you copy data from a source to a destination. For an introduction to the wizard, including how to get it, permissions information, and how to get help with the tool, visit the [Import and Export Data with the SQL Server Import and Export Wizard](#) web page.

For an overview of steps for the wizard, visit the [Steps in the SQL Server Import and Export Wizard](#) web page.

Next Steps:

[Azure SQL Database](#)

[Azure Active Directory](#)

Discover, identify, and classify personal data in Microsoft Azure

8/25/2017 • 9 min to read • [Edit Online](#)

This article provides guidance on how to discover, identify, and classify personal data in several Azure tools and services, including using Azure Data Catalog, Azure Active Directory, SQL Database, Power Query for Hadoop clusters in Azure HDInsight, Azure Information Protection, Azure Search, and SQL queries for Azure Cosmos DB.

Scenario, problem statement, and goal

A U.S.-based sports company collects a variety of personal and other data. This includes customers and employee data. The company keeps it in multiple databases, and stores it in several different locations in their Azure environment. In addition to selling sports equipment, they also host and manage registration for elite athletic events around the world, including in the EU, and in some cases the customer data they collect includes medical information.

Since the company hosts many international bicycling tours every year and has contingent staff in locations around the globe, a couple of the data sets are quite large. The company also has developer-built applications that are used by both customers and employees.

The company wants to address the following problems:

- Customer and employee personal data must be classified/distinguished from the other data the company collects in order to ensure proper access and security.
- The data admin needs to easily discover the location of customer personal data across various areas of the Azure environment.
- Customer and employee personal data that appears in shared documents and email communications must be labeled to help ensure that it's kept secure.
- The company's app developers need a way to easily search for customer and employee personal data in their web and mobile apps.
- Developers also need to query their document database for personal data.

Company goals

- All customer and employee personal data must be tagged/annotated in Azure Data Catalog so it can be found easily. Ideally customer and employee personal data are tagged/annotated separately.
- Personal data from customer and employee user profiles and work information residing in Azure Active Directory must be easily located.
- Personal data residing in multiple SQL databases must be easily queried.
- Some of the company's large data sets are managed through Azure HDInsight and stored in Hadoop. They must be imported into Excel so they can be queried for personal data.
- Personal data shared in documents and email communications must be classified, labeled, and kept secure with Azure Information Protection.
- The company's app developers must be able to discover customer and employee personal data in the apps they've built, which they can do with Azure Search.
- Developers must be able to find personal data in their document database.

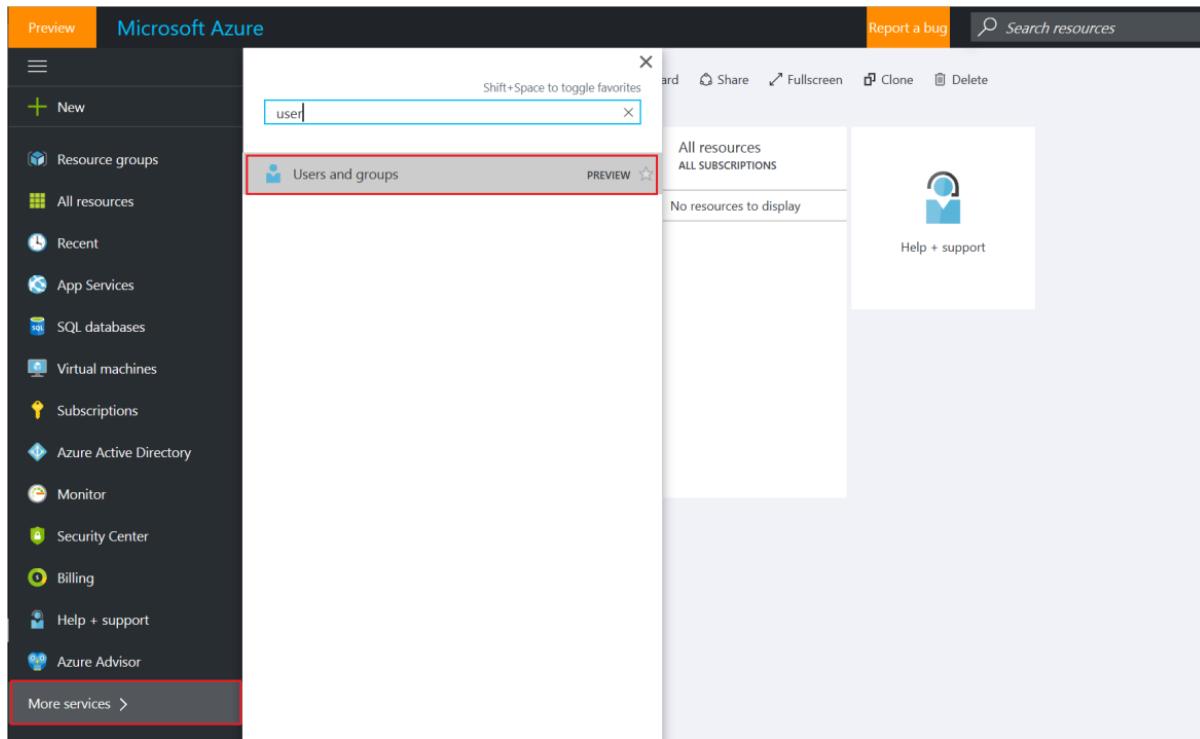
Azure Active Directory: Data discovery

Azure Active Directory is Microsoft's cloud-based, multi-tenant directory and identity management service. You can locate customer and employee user profiles and user work information that contain personal data in your Azure Active Directory (AAD) environment by using the [Azure portal](#).

This is particularly helpful if you want to find or change personal data for a specific user. You can also add or change user profile and work information. You must sign in with an account that's a global admin for the directory.

How do I locate or view user profile and work information?

1. Sign in to the [Azure portal](#) with an account that's a global admin for the directory.
2. Select **More services**, enter **Users and groups** in the text box, and then select **Enter**.



3. On the **Users and groups** blade, select **Users**.

The screenshot shows the Azure portal interface for User management. On the left, there's a sidebar with various service links like Resource groups, All resources, Recent, App Services, etc. The main content area is titled 'User management - Users' and shows a preview for 'f128 Photography'. At the top right are buttons for 'Add', 'Columns', and 'Multi-Factor Au..'. Below the title is a search bar labeled 'Search (Ctrl+ /)'. The left sidebar has sections for GENERAL (Overview, Audit), RESOURCES (Users, Groups), and CONFIGURATION (Domains, Azure AD Connect, Settings, Password reset). The 'Users' link under RESOURCES is highlighted with a red box. The main content area displays a table of users:

NAME	USER NAME
Aaron Nicholls	tnich@f128.info
Abby Brennan	abrennan@f128.info
Adam Carlton	acarlton@f128.info
Adam Steenwyk	adam@f128.info
Adam Steenwyk	adam@f128.onmicrosoft.com
Adam Steenwyk (admin f128.onmicrosoft.com)	admin@f128.onmicrosoft.com
Adam Steenwyk (admin)	admin@f128.info
Adam Steenwyk (MSA)	ajamess_gmail.com#EXT#@f128.onmicrosoft.com
Adrian Martin	amartin@f128.info
Aidan Mitzner	amitzner@f128.info
Ajay Mokashi	amokashi@f128.info
Alain DuBois	adubois@f128.info
Alberto Bassani	abassani@f128.info
Alejandro Ruiz	aruiz@f128.info
Alex Grossman	agrossman@f128.info
Alice Gartner	alice@f128.info
Allison Hunter	ahunter@f128.info
Alvin Hwang	ahwang@f128.info
Amanda Baker	abaker@f128.info
Amanda Mackenzie	amackenzie@f128.info
Amelia Casias	acasias@f128.info

4. On the **Users and groups - Users** blade, select a user from the list, and then, on the blade for the selected user, select **Profile** to view user profile information that might contain personal data.

The screenshot shows the Azure Active Directory User - Profile blade for a user named Angie Shannon. On the left, there is a list of users with their names and email addresses. On the right, the profile details for Angie Shannon are displayed. The 'Profile' section is selected in the navigation menu. The profile information includes the name (Angie Shannon), user name (angie@litwarecorp.com), first name, last name, and a placeholder photo. There are fields for uploading a new photo, object ID, source (Azure Active Directory), and settings for blocking sign-in (set to No). The 'Work Info' section is also visible.

5. If you need to add or change user profile information, you can do so, and then, in the command bar, select **Save**.
6. On the blade for the selected user, select **Work Info** to view user work information that may contain personal data.

The screenshot shows the Microsoft Azure portal interface. On the left, there is a list of users with their names and email addresses. On the right, there is a form for updating user work information. The 'Work Info' tab is selected, and its fields are highlighted with a red box. The fields include Job title, Department, Manager ID, Street address, City, State or province, Country or region, Office, and Mobile phone.

- If you need to add or change user work information, you can do so, and then, in the command bar, select **Save**.

Azure SQL Database: Data discovery

[Azure SQL Database](#) is a cloud database that helps developers build and maintain applications. Personal data can be found in [Azure SQL Database](#) using standard SQL queries. Azure SQL elastic query (preview) enables users to perform cross-database queries.

A detailed [SQL database](#) tutorial explains many aspects of using a SQL database, including how to build one and how to run data queries. The following is a summary of the information available in the tutorial with links to specific sections.

How do I build a SQL database?

There are three ways to do it:

- An Azure SQL database can be created in the [Azure portal](#). In the tutorial, you'll use a specific set of compute and storage resources within a resource group and logical server. You'll use sample data from a fictitious company called AdventureWorks. You'll also create a server-level firewall rule. To learn how to do this, visit the [Create an Azure SQL database in the Azure portal](#) tutorial.

- A SQL database can also be created in the [Azure Cloud Shell](#) CLI, a browser-based command-line tool. The tool is available in the Azure portal and can be run directly from there. In this tutorial, you launch the tool, define script variables, create a resource group and logical server, and configure a server firewall rule. Then you create a database with sample data. To learn how to create your database this way, visit the [Create a single Azure SQL database using the Azure CLI](#) tutorial.

```

Bash ▾ ⏎ ? Azure CLI Documentation
Requesting a Cloud Shell...succeeded.
Connecting terminal...
Welcome to Azure Cloud Shell (Preview)
Type "help" to learn about Cloud Shell
Type "az" to use Azure CLI 2.0
neil@Azure:-$ 
  
```

NOTE

Azure CLI is commonly used by Linux admins and developers. Some users find it easier and more intuitive than PowerShell, which is your third option.

- Finally, you can create a SQL database using PowerShell, which is a command line/script tool used to create and manage Azure and other resources. In this tutorial, you launch the tool, define script variables, create a resource group and logical server, and configure a server firewall rule. Then you'll create a database with sample data.

The tutorial requires the Azure PowerShell module version 4.0 or later. Run `Get-Module -ListAvailable AzureRM` to find your version. If you need to install or upgrade, see [Install Azure PowerShell module](#).

```
New-AzureRmSQLDatabase -ResourceGroupName $resourcegroupname ` 
-ServerName $servername ` 
-DatabaseName $databasename ` 
-RequestedServiceObjectiveName "S0"
```

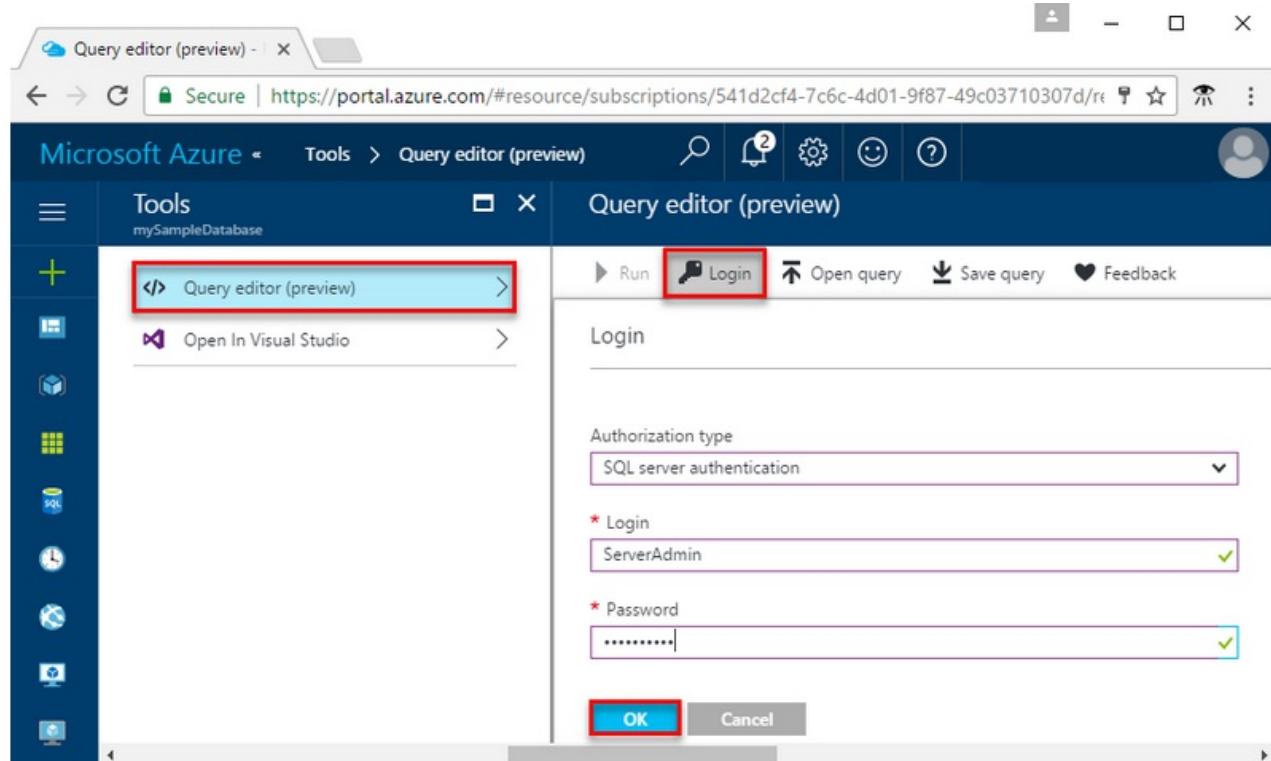
To learn how to create your database this way, visit the [Create a single Azure SQL database using Powershell](#) tutorial.

NOTE

Windows admins tend to use PowerShell, but some of them prefer Azure CLI.

How do I search for personal data in SQL database in the Azure portal?**

You can use the built-in query editor tool inside the Azure portal to search for personal data. You'll log in to the tool using your SQL server admin login and password, and then enter a query.



Step 5 of the tutorial shows an example query in the query editor pane, but it doesn't focus on personal or sensitive information(it also combines data from two tables and creates aliases for the source column in the data set being returned). The following screenshot shows the query from Step 5 as well as the results pane that's returned:

The screenshot shows the Microsoft Azure SQL Databases interface. In the top navigation bar, it says "Microsoft Azure SQL databases > mySampleDatabase > Tools > Query editor (preview)". The main area is titled "Query editor (preview)" with a sub-header "Authenticated as ServerAdmin". A red box highlights the "Run" button in the toolbar above the query editor. Another red box highlights the query code itself:

```
1 SELECT TOP 20 pc.Name as CategoryName, p.name as ProductName
2 FROM SalesLT.ProductCategory pc
3 JOIN SalesLT.Product p
4 ON pc.productcategoryid = p.productcategoryid;
```

The results pane is selected and highlighted with a red box. It displays a table with two columns: "CATEGORYNAME" and "PRODUCTNAME". The data is as follows:

CATEGORYNAME	PRODUCTNAME
Road Frames	HL Road Frame - Black, 58
Road Frames	HL Road Frame - Red, 58
Helmets	Sport-100 Helmet, Red
Helmets	Sport-100 Helmet, Black
Socks	Mountain Bike Socks, M
Socks	Mountain Bike Socks, L
Helmets	Sport-100 Helmet, Blue
Caps	AWC Logo Cap
Jerseys	Long-Sleeve Logo Jersey, S
Jerseys	Long-Sleeve Logo Jersey, M

If your database was called MyTable, a sample query for personal information might include name, Social Security number and ID number and would look like this:

"SELECT Name, SSN, ID number FROM MyTable"

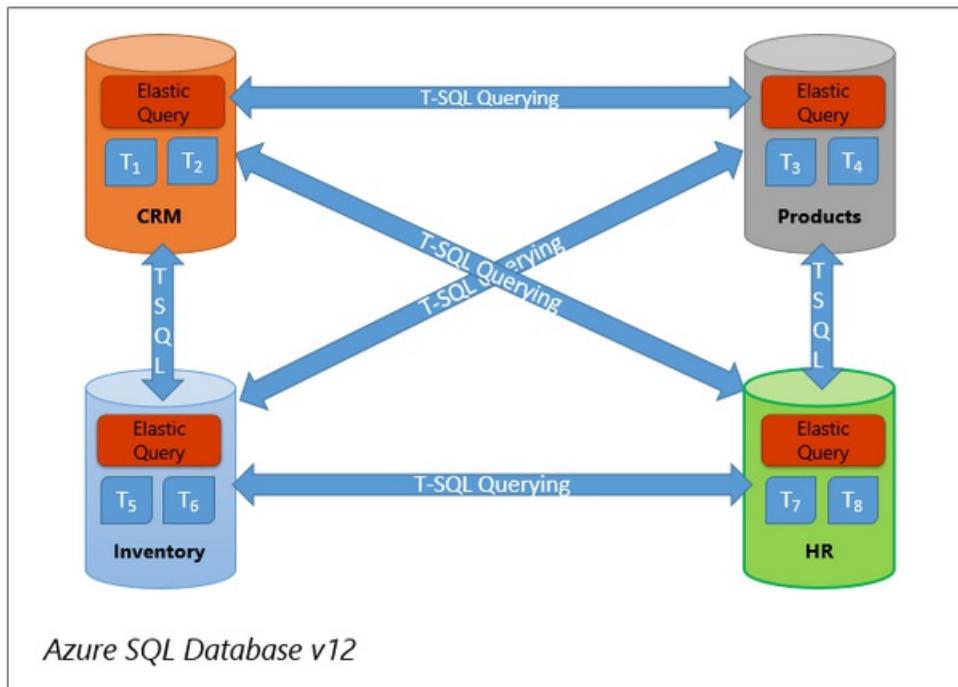
You'd run the query and then see the results in the **Results** pane.

For more information on how to query a SQL database in the Azure portal, visit the [Query the SQL database](#) section of the tutorial.

How do I search for data across multiple databases?

SQL elastic query (preview) enables you to perform cross-database and multiple database queries and return a single result. The [tutorial overview](#) includes a detailed description of scenarios and explains the difference between vertical and horizontal database partitioning. Horizontal partitioning is called "sharding."

Vertical partitioning - Using elastic query to query across various databases



Azure SQL Database v12

Horizontal partitioning - Using elastic query for reporting over sharded data tiers



To get started, visit the [Azure SQL Database elastic query overview \(preview\)](#) page.

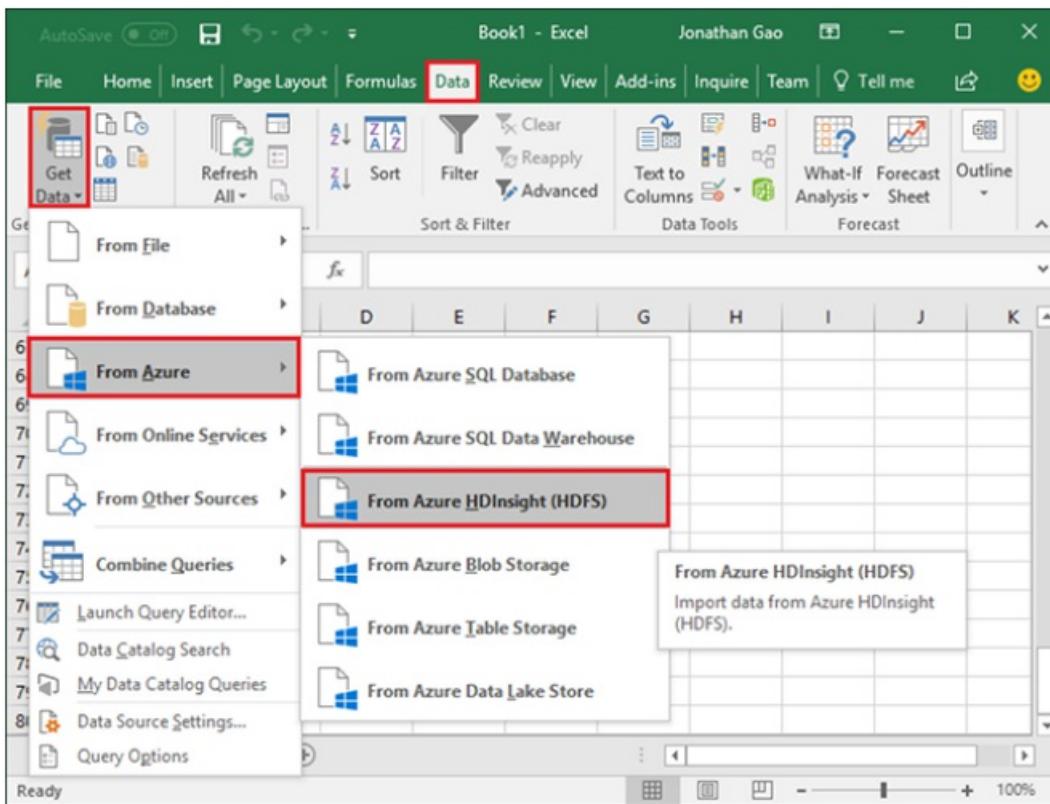
Power Query (for importing Azure HDInsight Hadoop clusters): data discovery for large data sets

Hadoop is an open source Apache storage and processing service for large data sets, which are analyzed and stored in Hadoop clusters. [Azure HDInsight](#) allows users to work with Hadoop clusters in Azure. Power Query is an Excel add-in that, among other things, helps users discover data from different sources.

Personal data associated with Hadoop clusters in Azure HDInsight can be imported to Excel with Power Query. Once the data is in Excel you can use a query to identify it.

How do I use Excel Power Query to import Hadoop clusters in Azure HDInsight into Excel?

An HDInsight tutorial will walk you through this entire process. It explains prerequisites, and includes a link to a [Get started with Azure HDInsight](#) tutorial. Instructions cover Excel 2016 as well as 2013 and 2010 (steps are slightly different for the older versions of Excel). If you don't have the Excel Power Query add-in, the tutorial shows you how to get it. You'll start the tutorial in Excel and will need to have an Azure Blob storage account associated with your cluster.



To learn how to do this, visit the [Connect Excel to Hadoop by using Power Query](#) tutorial.

Source: [Connect Excel to Hadoop by using Power Query](#)

Azure Information Protection: personal data classification for documents and email

[Azure Information Protection](#) can help Azure customers apply labels to classify and protect internally or externally shared documents and email communications. Some of these items may contain customer or employee personal information. Rules and conditions can be defined automatically or manually, by administrators or by users. For example, if a user is saving a document that includes credit card information, he or she would see a label recommendation that was configured by the administrator.

How do I try it?

If you'd like to give Azure Information Protection a try to see if it might be a fit for your organization, visit the [Quickstart tutorial](#). It walks you through five basic steps—from installation to configuring policy to seeing classification, labeling, and sharing in action—and should take less than a half hour.

How do I deploy it?

If you'd like to deploy Azure Information Protection for your organization, visit the [deployment roadmap for classification, labeling, and protection](#).

Is there anything else I should know?

For complementary information that will help you think through how to set it up, visit the [Ready, set, protect!](#) blog post. And check the Learn more links listed below for more on Azure Information Protection.

Azure Search: data discovery for developer apps

[Azure Search](#) is a cloud search solution for developers, and provides a rich data search experience for your applications. Azure Search allows you to locate data across user-defined indexes, sourced from Azure Cosmos DB, Azure SQL Database, Azure Blob Storage, Azure Table storage, or custom customer JSON data. You can also structure Lucene queries using the Azure Search REST API to search for personal data types or the personal data of

specific individuals. Features include full text search, simple query syntax, and Lucene query syntax.

How do I use SQL to query data?

To begin with the basics, visit the [Azure CosmosDB: How to query using SQL](#) tutorial. The tutorial provides a sample document and two sample SQL queries and results.

For more in-depth guidance on building SQL queries, visit [SQL queries for Azure Cosmos DB Document DB API](#).

If you're new to Azure Cosmos DB and would like to learn how to create a database, add a collection, and add data, visit the [Azure Cosmos DB: Build a DocumentDB API web app](#) Quickstart tutorial. If you'd like to do this in a language other than .NET, such as Java or Python, just choose your preferred language once you get to the site.

Next steps

[Azure SQL Database](#)

[What is SQL Database?](#)

[SQL Database Query Editor available in Azure portal](#)

[What is Azure Information Protection?](#)

[What is Azure Rights Management?](#)

[Azure Information Protection: Ready, set, protect!](#)

Protect personal data in Microsoft Azure

8/30/2017 • 1 min to read • [Edit Online](#)

This article introduces a series of articles that help you use Azure security technologies and services to protect personal data. This is a key requirement for many corporate and industry compliance and governance initiatives. The scenario, problem statement and company goals are included here.

Scenario and problem statement

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, Adriatic, and Baltic seas, as well as the British Isles. To support those efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark and the U.K.

The company uses Microsoft Azure to store corporate data in the cloud. This may include customer and/or employee information such as:

- addresses
- phone numbers
- tax identification numbers
- credit card information

The company must protect the privacy of customer and employee data while making data accessible to those departments that need it. (such as payroll and reservations departments)

Company goals

- Data sources that contain personal data are encrypted when residing in cloud storage.
- Personal data that is transferred from one location to another is encrypted while in-transit. This is true if the data is traveling across the virtual network or across the Internet between the corporate datacenter and the Azure cloud.
- Confidentiality and integrity of personal data is protected from unauthorized access by strong identity management and access control technologies.
- Personal data is protected from exposure via data breach via monitoring for vulnerabilities and threats.
- The security state of Azure services that store or transmit personal data is assessed to identify opportunities to better protect personal data.

Data protection guidance

The following articles contain technical how-to guidance that will help you attain the personal data protection goals listed above:

- [Azure encryption technologies](#)
- [Azure encryption technologies](#)
- [Azure identity and access technologies](#)
- [Azure network security technologies](#)
- [Azure Security Center](#)

Next steps

- [Azure Security Information Site](#)
- [Microsoft Trust Center](#)
- [Azure Security Center](#)
- [Azure Security Team Blog](#)
- [Azure.com Blog - Security](#)

Protect personal data from breaches and attacks: Azure Security Center

8/28/2017 • 8 min to read • [Edit Online](#)

This article will help you understand how to use Azure Security Center to protect personal data from breaches and attacks.

Scenario

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, and Baltic seas, as well as the British Isles. To help in those efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark, and the U.K.

The company uses Microsoft Azure to store corporate data in the cloud. This includes personal identifiable information such as names, addresses, phone numbers, and credit card information. It also includes Human Resources information such as:

- Addresses
- Phone numbers
- Tax identification numbers
- Medical information

The cruise line also maintains a large database of reward and loyalty program members. Corporate employees access the network from the company's remote offices and travel agents located around the world have access to some company resources. Personal data travels across the network between these locations and the Microsoft data center.

Problem statement

The company is concerned about the threat of attacks on their Azure resources. They want to prevent exposure of customers' and employees' personal data to unauthorized persons. They want guidance on both prevention and response/remediation, as well as an effective way to monitor the ongoing security of their cloud resources. They need a strong line of defense against today's sophisticated and organized attackers.

Company goal

One of the company's goals is to ensure the privacy of customers' and employees' personal data by protecting it from threats. One of their goals is to respond immediately to signs of breach to mitigate the impact. It requires a way to assess the current state of security, identify vulnerable configurations, and remediate them.

Solutions

Microsoft Azure Security Center (ASC) provides an integrated security monitoring and policy management solution. It delivers easy-to-use and effective threat prevention, detection, and response capabilities.

Prevention

ASC helps you prevent breaches by enabling you to set security policies, provide just-in-time access, and implement security recommendations.

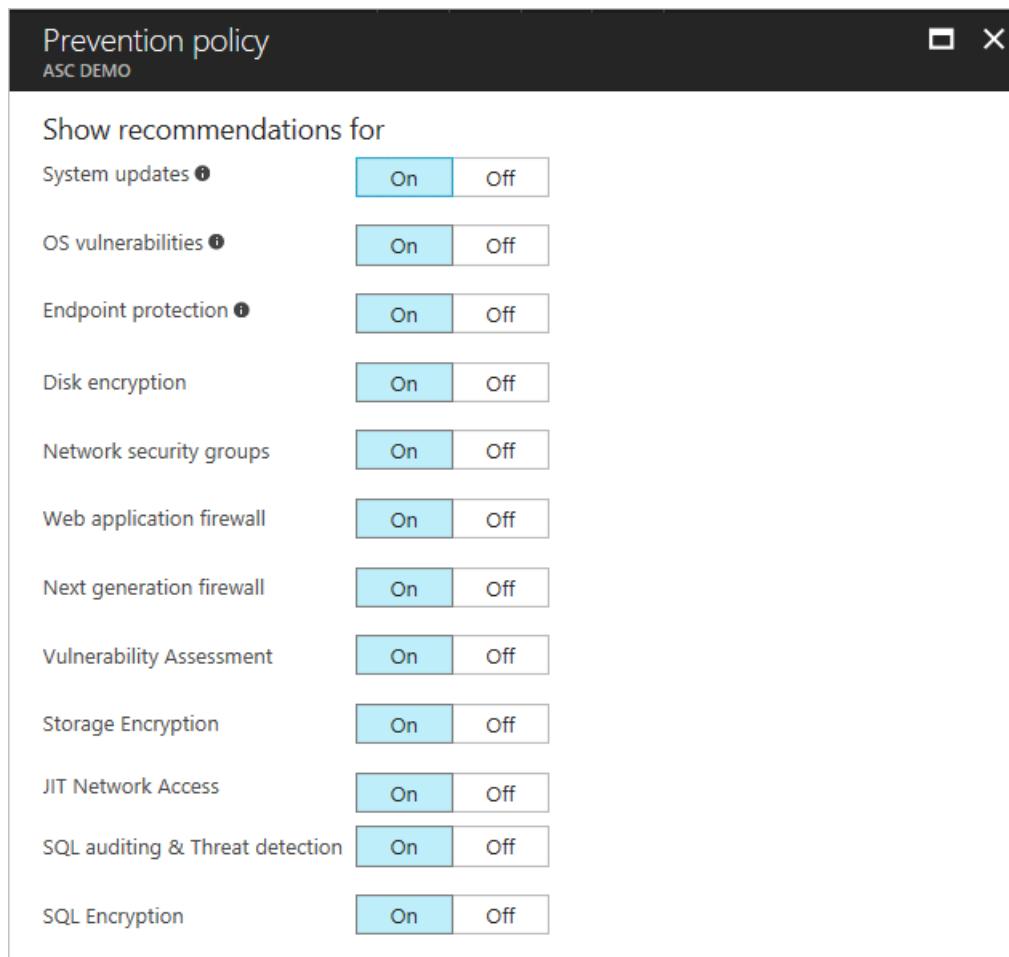
A security policy defines the set of controls recommended for resources within the specified subscription. Just in

time access can be used to lock down inbound traffic to your Azure VMs, reducing exposure to attacks. Security recommendations are created by ASC after analyzing the security state of your Azure resources.

How do I set security policies in ASC?

You can configure security policies for each subscription. To modify a security policy, you must be an owner or contributor of that subscription. In the Azure portal, do the following:

1. Select **Policy** in the ASC dashboard.
2. Select the subscription on which you want to enable the policy.
3. Choose **Prevention policy** to configure policies per subscription. **Collect data from virtual machines** should be set to **On**.
4. In the **Prevention policy** options, select **On** to enable the security recommendations that are relevant for the subscription.



For more detailed instructions and an explanation of each of the policy recommendations that can be enabled, see [Set security policies in Azure Security Center](#).

How do I configure Just in Time Access (JIT)?

When JIT is enabled, Security Center locks down inbound traffic to your Azure VMs by creating an NSG rule. You select the ports on the VM to which inbound traffic will be locked down. To use JIT access, do the following:

1. Select the **Just in time VM access tile** on the ASC blade.
2. Select the **Recommended** tab.
3. Under **VMs**, select the VMs that you want to enable. This puts a checkmark next to a VM.
4. Select **Enable JIT** on VMs.
5. Select **Save**.

Then you can see the default ports that ASC recommends being enabled for JIT. You can also add and configure a new port on which you want to enable the just in time solution. The **Just in time VM access** tile in the Security Center shows the number of VMs configured for JIT access. It also shows the number of approved access requests made in the last week.

The screenshot shows the Microsoft Azure Security Center - Overview page. The left sidebar contains navigation links for General, Prevention, Detection, and Advanced Cloud Defense. The main area has tabs for Power BI, Subscriptions, and Log Integration. The 'Just in time VM access' tile is highlighted with a red box. It displays the following information:

- Just in time VM access**: Last week
- What is just in time VM access?**: Just in time VM access enables you to lock down your VMs in the network level by blocking inbound traffic to specific ports. It enables you to control the access and reduce the attack surface to your VMs, by allowing access only upon a specific need.
- How does it work?**: Upon a user request, based on Azure BBAC, Security Center will decide whether to grant access. If a request is approved, Security Center automatically configures the NSGs to allow inbound traffic to these ports, for only 3 hours, after which it restores the NSGs to their previous states.
- For more information go to the documentation >**
- Virtual machines**: Configured, Recommended, No recommendation
- VMs**: 5 VMs

VIRTUAL MACHINE	APPROVED	LAST ACCESS	LAST USER
vm1	1 Requests	6/26/17 11:13 AM	bekliger@microsoft.com
vm2	1 Requests	6/26/17 11:13 AM	bekliger@microsoft.com
vm3	0 Requests	N/A	N/A
vm2WL	0 Requests	N/A	N/A
vm3WL	0 Requests	N/A	N/A

For instructions on how to do this, and additional information about Just in Time access, see [Manage virtual machine access using just in time](#).

How do I implement ASC security recommendations?

When Security Center identifies potential security vulnerabilities, it creates recommendations. The recommendations guide you through the process of configuring the needed controls.

1. Select the **Recommendations** tile on the ASC dashboard.
2. View the recommendations, which are shown in a table format where each line represents one recommendation.
3. To filter recommendations, select **Filter** and select the severity and state values you wish to see.
4. To dismiss a recommendation that is not applicable, you can right click and select **Dismiss**.
5. Evaluate which recommendation should be applied first.
6. Apply the recommendations in order of priority.

For a list of possible recommendations and walk-throughs on how to apply each, see [Managing security recommendations in Azure Security Center](#).

Detection and Response

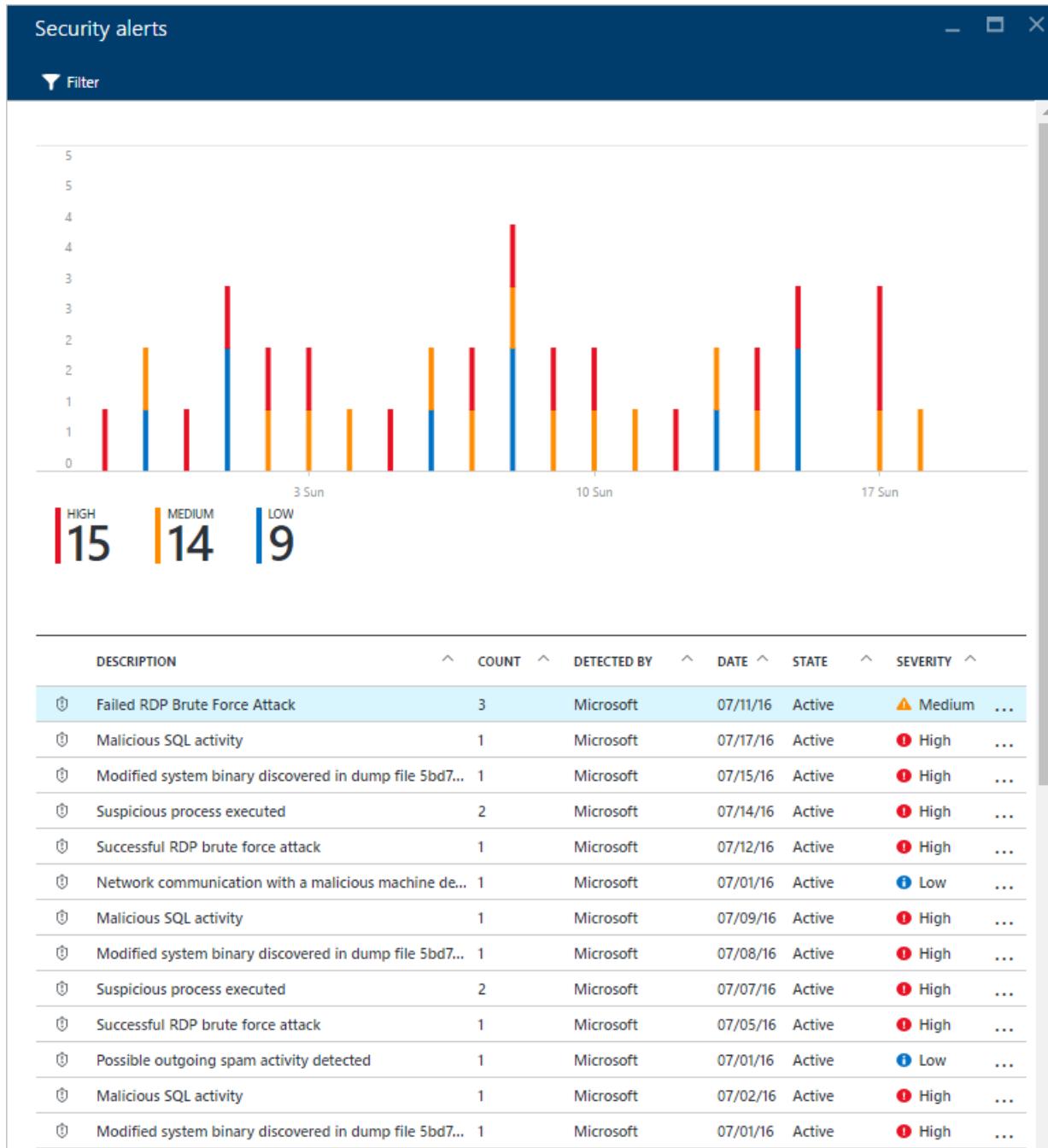
Detection and response go together, as you want to respond as quickly as possible after a threat is detected. ASC threat detection works by automatically collecting security information from your Azure resources, the network, and connected partner solutions. ASC can rapidly update its detection algorithms as attackers release new and increasingly sophisticated exploits. For more detailed information on how ASC's threat detection works, see [Azure Security Center detection capabilities](#).

How do I manage and respond to security alerts?

A list of prioritized security alerts is shown in Security Center along with the information you need to quickly investigate the problem. Security Center also includes recommendations for how to remediate an attack. To manage your security alerts, do the following:

1. Select the **Security alerts** tile in the ASC dashboard. This shows details for each alert.
2. To filter alerts based on date, state, and severity, select **Filter** and then select the values you want to see.

- To respond to an alert, select it and review the information, then select the resource that was attacked.
- In the **Description** field, you'll see details, including recommended remediation.



For more detailed instructions on responding to security alerts, see [Managing and responding to security alerts in Azure Security Center](#).

For further help in investigating security alerts, the company can integrate ASC alerts with its own SIEM solution, using [Azure Log Integration](#).

How do I manage security incidents?

In ASC, a security incident is an aggregation of all alerts for a resource that align with kill chain patterns. An Incident will reveal the list of related alerts, which enables you to obtain more information about each occurrence. Incidents appear in the Security Alerts tile and blade.

To review and manage security incidents, do the following:

- Select the **Security alerts** tile. If a security incident is detected, it will appear under the security alerts graph. It will have an icon that's different from other alerts.
- Select the incident to see more details about this security incident. Additional details include its full

description, its severity, its current state, the attacked resource, the remediation steps for the incident, and the alerts that were included in this incident.

You can filter to see **incidents only**, **alerts only**, or **both**.

How do I access the Threat Intelligence Report?

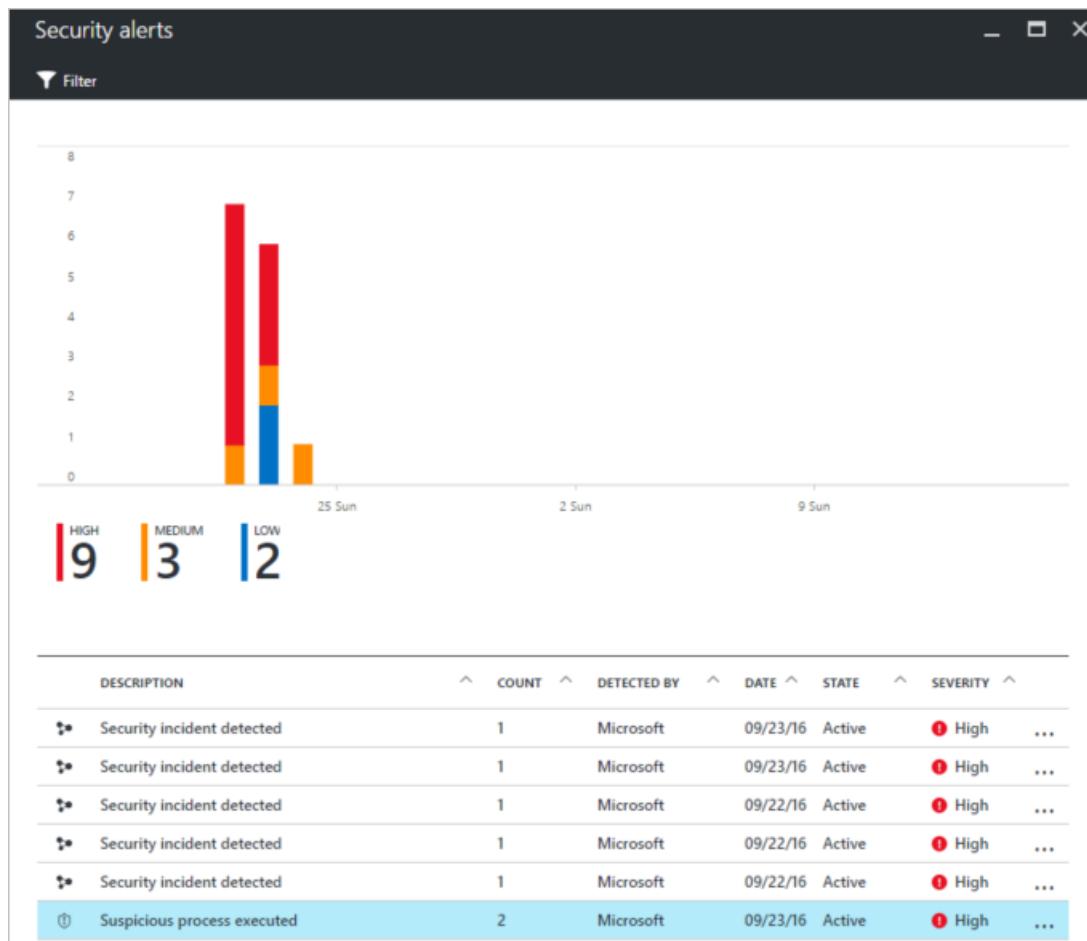
ASC analyzes information from multiple sources to identify threats. To assist incident response teams investigate and remediate threats, Security Center includes a threat intelligence report that contains information about the threat that was detected.

Security Center has three types of threat reports, which can vary per attack. The reports available are:

- Activity Group Report: provides deep dives into attackers, their objectives and tactics.
- Campaign Report: focuses on details of specific attack campaigns.
- Threat Summary Report: covers all items in the previous two reports.

This type of information is very useful during the incident response process, where there is an ongoing investigation to understand the source of the attack, the attacker's motivations, and what to do to mitigate this issue moving forward.

1. To access the threat intelligence report, do the following:
2. Select the **Security alerts** tile on the ASC dashboard.
3. Select the security alert for which you want to obtain more information.
4. In the **Reports** field, click the link to the threat intelligence report.
5. This will open the PDF file, which you can download.



For additional information about the ASC threat intelligence report, see [Azure Security Center Threat Intelligence](#)

Report.

Assessment

To help with testing, assessment and evaluation of your security posture, ASC provides for integrated vulnerability assessment with Qualys cloud agents, as a part of its virtual machine recommendations component.

The Qualys agent reports vulnerability data to the Qualys management platform, which then sends vulnerability and health monitoring data back to ASC. The recommendation to add a vulnerability assessment solution is displayed in the **Recommendations** blade on the ASC dashboard.

After the vulnerability assessment solution is installed on the target VM, Security Center scans the VM to detect and identify system and application vulnerabilities. Detected issues are shown under the **Virtual Machines Recommendations** option.

How do I implement a vulnerability assessment solution?

If a Virtual Machine does not have an integrated vulnerability assessment solution already deployed, Security Center recommends that it be installed.

1. In the ASC dashboard, on the **Recommendations** blade, select **Add a vulnerability assessment solution**.
2. Select the VMs where you want to install the vulnerability assessment solution.
3. Click on **Install on [number of] VMs**.
4. Select a partner solution in the Azure Marketplace, or under **Use existing solution**, select **Qualys**.
5. You can turn the auto update settings on or off in the **Partner Solutions** blade.

For further instructions on how to implement a vulnerability assessment solution, see [Vulnerability Assessment in Azure Security Center](#).

Next steps

- [Azure Security Center quick start guide](#)
- [Introduction to Azure Security Center](#)
- [Integrating Azure Security Center alerts with Azure log integration](#)
- [Boost Azure Security Center with Integrated Vulnerability Assessment](#)

Protect personal data with network security features: Azure Application Gateway and Network Security Groups

8/30/2017 • 6 min to read • [Edit Online](#)

This article provides information and procedures that will help you use Azure Application Gateway and Network Security Groups to protect personal data.

An important element in a multi-layered security strategy to protect the privacy of personal data is a defense against common vulnerability exploits such as SQL injection or cross-site scripting. Keeping unwanted network traffic out of your Azure virtual network helps protect against potential compromise of sensitive data, and Microsoft Azure gives you tools to help protect your data against attackers.

Scenario

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, Adriatic, and Baltic seas, as well as the British Isles. In furtherance of those efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark and the U.K.

The company uses Microsoft Azure to store corporate data in the cloud and run applications on virtual machines that process and access this data. This data includes personal identifiable information such as names, addresses, phone numbers, and credit card information of its global customer base. It also includes traditional Human Resources information such as addresses, phone numbers, tax identification numbers and other information about company employees in all locations. The cruise line also maintains a large database of reward and loyalty program members that includes personal information to track relationships with current and past customers.

Corporate employees access the network from the company's remote offices and travel agents located around the world have access to some company resources and use web-based applications hosted in Azure VMs to interact with it.

Problem statement

The company must protect the privacy of customers' and employees' personal data from attackers who exploit software vulnerabilities to run malicious code that could expose personal data stored or used by the company's cloud-based applications.

Company goal

The company's goal to ensure that unauthorized persons cannot access corporate Azure Virtual Networks and the applications and data that reside there by exploiting common vulnerabilities.

Solutions

Microsoft Azure provides security mechanisms to help prevent unwanted traffic from entering Azure Virtual Networks. Control of inbound and outbound traffic is traditionally performed by firewalls. In Azure, you can use the Application Gateway with the Web Application Firewall and Network Security Groups (NSG), which act as a simple distributed firewall. These tools enable you to detect and block unwanted network traffic.

Application Gateway/Web Application Firewall

The [Web Application Firewall](#) (WAF) component of the [Azure Application Gateway](#) protects web applications, which are increasingly targets of malicious attacks that exploit common known vulnerabilities. A centralized WAF both protects against web attacks and simplifies security management without requiring any application changes.

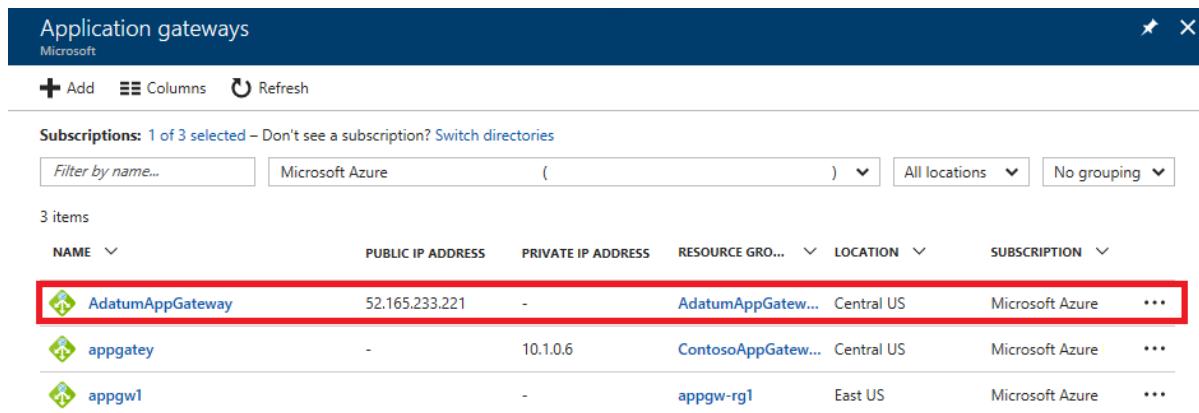
Azure WAF addresses various attack categories including SQL injection, cross site scripting, HTTP protocol violations and anomalies, bots, crawlers, scanners, common application misconfigurations, HTTP Denial of Service, and other common attacks such as command injection, HTTP request smuggling, HTTP response splitting, and remote file inclusion attacks.

You can create an application gateway with WAF, or add WAF to an existing application gateway. In either case, Azure Application Gateway requires its own subnet.

How do I create an application gateway with WAF?

To create a new application gateway with WAF enabled, do the following:

1. Log in to the Azure portal and in the **Favorites** pane of the portal, click **New**
2. In the **New** blade, click **Networking**.
3. Click **Application Gateway**.
4. Navigate to the Azure portal, [click New > Networking > Application Gateway](#).

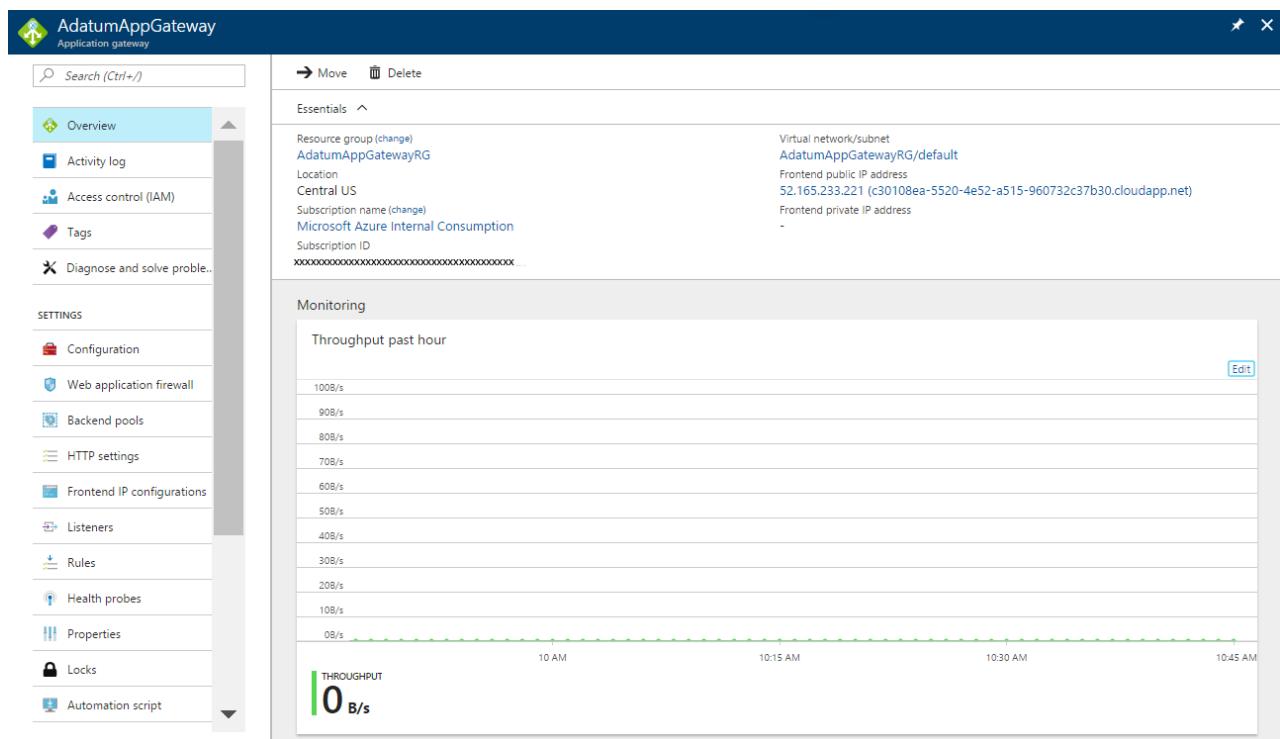


The screenshot shows the 'Application gateways' blade in the Azure portal. At the top, there's a header bar with a Microsoft logo, a search bar, and a refresh button. Below the header, it says 'Subscriptions: 1 of 3 selected - Don't see a subscription? Switch directories'. There are filter options for 'Filter by name...', 'Microsoft Azure', 'All locations', and 'No grouping'. The main table has columns: NAME, PUBLIC IP ADDRESS, PRIVATE IP ADDRESS, RESOURCE GRO..., LOCATION, and SUBSCRIPTION. There are three items listed: 'AdatumAppGateway' (selected), 'appgatay', and 'appgw1'. The 'AdatumAppGateway' row is highlighted with a red border.

NAME	PUBLIC IP ADDRESS	PRIVATE IP ADDRESS	RESOURCE GRO...	LOCATION	SUBSCRIPTION
AdatumAppGateway	52.165.233.221	-	AdatumAppGatew...	Central US	Microsoft Azure
appgatay	-	10.1.0.6	ContosoAppGatew...	Central US	Microsoft Azure
appgw1	-	-	appgw-rg1	East US	Microsoft Azure

5. In the **Basics** blade that appears, enter the values for the following fields: Name, Tier (Standard or WAF), SKU size (Small, Medium, or Large), Instance count (2 for high availability), Subscription, Resource group, and Location.
6. In the **Settings** blade that appears under **Virtual network**, click **Choose a virtual network**. This step opens enter the Choose virtual network blade.
7. Click **Create new** to open the **Create virtual network** blade.
8. Enter the following values: Name, Address space, Subnet name, Subnet address range. Click **OK**.
9. On the **Settings** blade under **Frontend IP configuration**, choose the IP address type.
10. Click **Choose a public IP address**, then **Create new**.
11. Accept the default value, and click **OK**.
12. On the **Settings** blade under **Listener configuration**, select to use HTTP or HTTPS under **Protocol**. To use HTTPS, a certificate is required.
13. Configure the WAF specific settings: **Firewall status (Enabled)** and **Firewall mode (Prevention)**. If you choose **Detection** as the mode, traffic is only logged.
14. Review the **Summary** page and click **OK**. Now the application gateway is queued up and created.

After the application gateway has been created, you can navigate to it in the portal and continue configuration of the application gateway.



How do I add WAF to an existing application?

To update an existing application gateway to support WAF in prevention mode, do the following:

1. In the Azure portal **Favorites** pane, click **All resources**.
2. Click the existing Application Gateway in the **All resources** blade.

NOTE

Note: If the subscription you selected already has several resources in it, you can enter the name in the Filter by name... box to easily access the DNS zone.

3. Click **Web application firewall** and update the application gateway settings: **Upgrade to WAF Tier** (checked), **Firewall status** (enabled), **Firewall mode** (Prevention). You also need to configure the rule set, and configure disabled rules.

For more detailed information on how to create a new application gateway with WAF and how to add WAF to an existing application gateway, see [Create an application gateway with web application firewall by using the portal](#).

Network Security Groups

A [network security group](#) (NSG) contains a list of security rules that allow or deny network traffic to resources connected to [Azure Virtual Networks](#) (VNet). NSGs can be associated to subnets or individual VMs. When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet. Traffic can further be restricted by also associating an NSG to a VM or NIC.

NSGs contain four properties: Name, Region, Resource group, and Rules.

NOTE

Although an NSG exists in a resource group, it can be associated to resources in any resource group, as long as the resource is part of the same Azure region as the NSG.

NSG rules contain nine properties: Name, Protocol (TCP, UDP, or *, which includes ICMP as well as UDP and TCP), Source port range, Destination port range, Source address prefix, Destination address prefix, Direction (inbound or outbound), Priority (between 100 and 4096) and Access type (allow or deny). All NSGs contain a set of default rules that can be deleted, or overridden by the rules you create.

How do I implement NSGs?

Implementing NSGs requires planning, and there are several design considerations you need to take into account. These include limits on the number of NSGs per subscription and rules per NSG; VNet and subnet design, special rules, ICMP traffic, isolation of tiers with subnets, load balancers, and more.

For more guidance in planning and implementing NSGs, and a sample deployment scenario, see [Filter network traffic with network security groups](#).

How do I create rules in an NSG?

To create inbound rules in an existing NSG, do the following:

1. Click **Browse**, and then **Network security groups**.
2. In the list of NSGs, click **NSG-FrontEnd**, and then **Inbound security rules**.
3. In the list of Inbound security rules, click **Add**.
4. Enter the values in the following fields: Name, Priority, Source, Protocol, Source range, Destination, Destination port range, and Action.

The new rule will appear in the NSG after a few seconds.

PRIORITY	NAME	SOURCE	DESTINATION	SERVICE	ACTION
100	sql-rule	192.168.1.0/24	Any	TCP/1433	Allow
200	web-rule	Any	Any	TCP/80	Allow

For more instructions on how to create NSGs in subnets, create rules, and associate an NSG with a front-end and back-end subnet, see [Create network security groups using the Azure portal](#).

Next steps

[Azure Network Security](#)

[Azure Network Security Best Practices](#)

[Get information about a network security group](#)

[Web application firewall \(WAF\)](#)

Azure Active Directory and Multi-Factor Authentication: Protect personal data with identity and access controls

8/30/2017 • 6 min to read • [Edit Online](#)

This article provides information and procedures you can use to protect personal data using Azure Active Directory and Multi-factor authentication security features and services.

Scenario

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, Adriatic, and Baltic seas, as well as the British Isles. To support those efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark and the U.K.

The company uses Microsoft Azure to store corporate data in the cloud. This includes personal identifiable information such as names, addresses, phone numbers, and credit card information of its global customer base. It also includes traditional Human Resources information such as addresses, phone numbers, tax identification numbers and other information about company employees in all locations. The cruise line also maintains a large database of reward and loyalty program members that includes personal information to track relationships with current and past customers.

Corporate employees access the network from the company's remote offices and travel agents located around the world have access to some company resources.

Problem statement

The company must protect the privacy of customers' and employees' personal data from attackers seeking to use compromised identities to gain access. They also must ensure that access to personal data by legitimate users is restricted to only those who need it to do their jobs.

Company goal

The company's goal is to ensure that access to personal data is strictly controlled. It is essential that identities of users with access to personal data be protected by strong authentication. A policy of [least privilege](#) must be enforced so that legitimate users have only the level of access they need, and no more.

Solutions

Microsoft Azure provides identity and access management tools to help companies control who has access to resources that contain personal data.

Azure Active Directory

[Azure Active Directory](#) (AAD) manages identities and controls access to Azure as well as other on-premises and other cloud resources, data, and applications. [Azure Active Directory Privileged Identity Management](#) helps Azure administrators to minimize the number of people who have access to certain information such as personal data. It enables them to discover, restrict, and monitor privileged identities and their access to resources, and to assign temporary, Just-In-Time (JIT) administrative rights to eligible users. It also provides insight into those who have AAD administrative privileges.

The activities involved in using AAD PIM include:

- Enabling Privileged Identity Management for your directory
- Using Privileged Identity Management admin dashboard to see important information at a glance
- Managing the privileged identities (administrators) by adding or removing permanent or eligible administrators to each role
- Configuring the role activation settings
- Activating roles
- Reviewing role activity

How do I enable AAD PIM?

To start using PIM for your directory, do the following:

1. Sign in to the Azure portal as a global administrator of your directory.
2. If your organization has more than one directory, select your username in the upper right-hand corner of the Azure portal. Select the directory where you will use Azure AD Privileged Identity Management.
3. Select **More services** and use the **Filter** textbox to search for Azure AD Privileged Identity Management.
4. Check **Pin to dashboard** and then click **Create**. The Privileged Identity Management application opens.

Once Azure AD Privileged Identity Management is set up, you see the navigation blade whenever you open the application.

Approvals and my audit history is now in preview

Search (Ctrl+ /)

QuickStart

Tasks

- My Roles
- Approve Requests (Preview)
- Pending Requests (Preview)
- Review Access

Manage

- Azure AD Directory Roles
- Azure Resources (Preview)

Activity

- My Audit History (Preview)

TROUBLESHOOTING + SUPPORT

- Troubleshoot
- New support request

What's New

Azure AD PIM Approvals are now in Preview!

Learn more

My Role Activations

Role Name	Previously assigned	Action required
Finance Administrator	No	No

Notifications

01:34

→ Introduction

Secure your organization by managing and restricting privileged access

Azure AD Privileged Identity Management

Azure AD Privileged Identity Management Powershell Module

For more information and instructions on getting started with AAD PIM, see [Start Using Azure AD Privileged Identity Management](#).

Azure Role-based Access Control

Azure Role-Based Access Control (RBAC) helps Azure administrators manage access to Azure resources by enabling the granting of access based on the user's assigned role. You can segregate duties within a team and grant only the amount of access to users, groups and applications that they need to perform their jobs.

Role-based access can be granted to users using the Azure portal, Azure Command-Line tools or Azure Management APIs.

For more information about Azure RBAC basics, see [Get started with Role-Based Access Control in the Azure Portal](#).

How do I manage Azure RBAC with PowerShell?

You can use PowerShell cmdlets to manage Azure RBAC, including the following management tasks:

- List roles
- See who has access
- Grant access
- Remove access
- Create a custom role
- Get Actions for a Resource Provider
- Modify a custom role
- Delete a custom role
- List custom roles

For instructions on how to manage Azure RBAC with PowerShell, see [Manage Role-based Access with Azure PowerShell](#).

Azure Multi-Factor Authentication

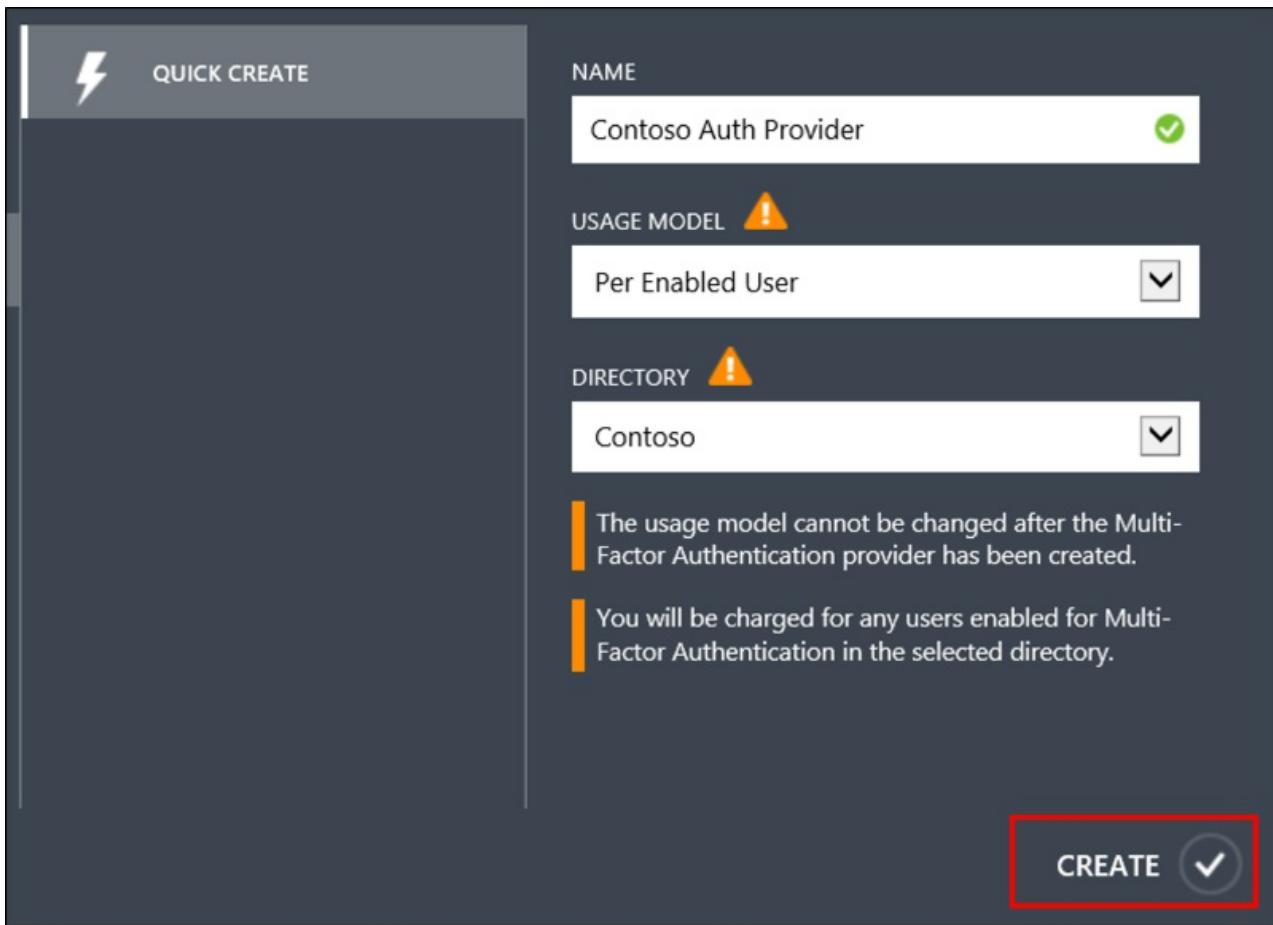
Azure Multi-Factor Authentication (MFA) is a two-step verification solution that helps safeguard access to data and applications, while meeting user demand for a simple sign-in process. It delivers strong authentication via a range of verification methods, including phone call, text message, or mobile app verification.

To deploy MFA in the Azure cloud, you need to first enable it and then turn on two-step verification for users.

How do I enable Azure to use MFA?

If your users have licenses that include Azure Multi-Factor Authentication, there's nothing that you need to do to turn on Azure MFA. If not, you need to create a Multi-Factor Auth provider in your directory. To do this, follow these steps:

1. Select **Active Directory** in the Azure classic portal (logged on as an administrator).
2. Select **Multi-Factor Authentication Providers**.
3. Select **New**, and then under **App Services**, select **Multi-Factor Auth Provider**.
4. Select **Quick Create**.
5. Fill in the name field and select a usage model (per authentication or per enabled user).
6. Designate a directory with which the MFA Provider is associated.
7. Click the **Create** button.



For more instructions on how to manage your Multi-Factor Auth Provider, see [Getting Started with an Azure Multi-Factor Auth Provider](#).

How do I turn on two-step verification for users?

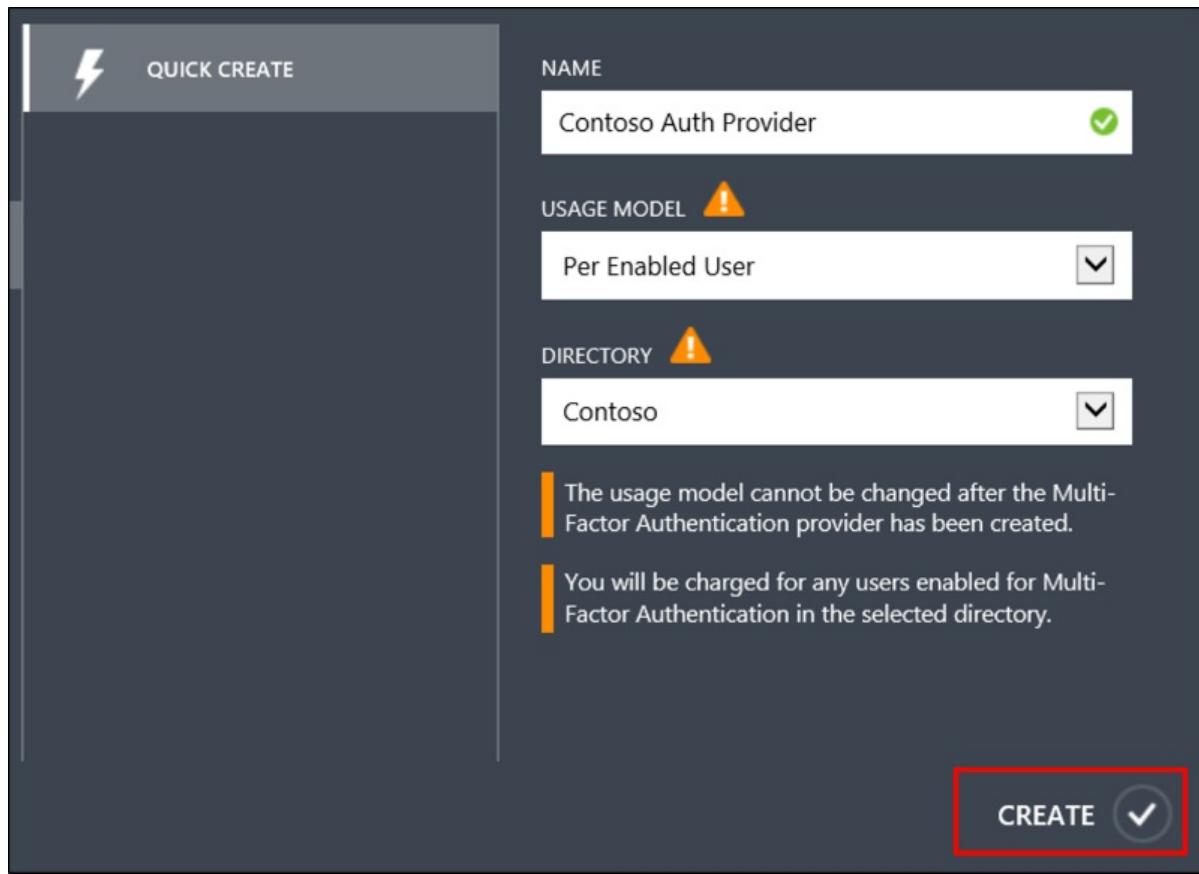
You can enforce two-step verification for all sign-ins, or you can create conditional access policies to require two-step verification only when specific conditions apply.

Enabling Azure MFA by changing user states is the traditional approach for requiring two-step verification. All the users that you enable will have the same requirement to perform two-step verification every time they sign in. Enabling a user overrides any conditional access policies that may affect that user.

Enabling Azure MFA with a conditional access policy is a more flexible approach for requiring two-step verification. You can create conditional access policies that apply to groups as well as individual users. High-risk groups can be given more restrictions than low-risk groups, or two-step verification can be required only for high-risk cloud apps and skipped for low-risk ones. However, conditional access is a paid feature of Azure Active Directory.

To enable MFA by changing user state, do the following:

1. Sign in to the Azure portal as an administrator.
2. Go to **Azure Active Directory > Users and groups > All users**.
3. Select **Multi-Factor Authentication**.
4. Find the user that you want to enable for Azure MFA. You may need to change the view at the top.
5. Check the box next to the user's name.
6. On the right, under quick steps, choose **Enable**.



7. Confirm your selection in the pop-up window that opens. Users for whom MFA has been enabled will be asked to register the next time they sign in.

To enable Azure MFA with a conditional access policy, do the following:

1. Sign in to the Azure portal as an administrator.
2. Go to **Azure Active Directory > Conditional access**.
3. Select **New policy**.
4. Under **Assignments**, select **Users and groups**. Use the **Include** and **Exclude** tabs to specify which users and groups will be managed by the policy.
5. Under **Assignments**, select **Cloud apps**. Choose to **include All cloud apps**.
6. Under **Access controls**, select **Grant**. Choose **Require multi-factor authentication**.
7. Turn **Enable policy** to **On** and then select **Save**.

For information on how to configure Azure MFA settings to set up fraud alerts, create a one-time bypass, use custom voice messages, configure caching, specify trusted IPs, create app passwords, enable remembering MFA for devices that users trust, and select verification methods, see [Configure Azure Multi-Factor Authentication Settings](#).

Next steps

- [Securing privileged access in Azure AD](#)
- [Frequently asked questions about Azure Multi-Factor Authentication](#)
- [Role-based Access Control troubleshooting](#)
- [Azure Active Directory Identity Protection](#)

title: Azure Protect personal data at rest with encryption | Microsoft Docs description: This article is part of a series helping you use Azure to protect personal data services: security documentationcenter: na author: Barclayn manager: MBaldwin editor: TomSh

ms.assetid: ms.service: security ms.devlang: na ms.topic: article ms.tgt_pltfrm: na ms.workload: na ms.date: 08/22/2017 ms.author: barclayn ms.custom:

Azure encryption technologies: Protect personal data at rest with encryption

This article helps you understand and use Azure encryption technologies to secure data at rest.

Encryption of data at rest is essential as a best practice to protect sensitive or personal data and to meet compliance and data privacy requirements. Encryption at rest is designed to prevent the attacker from accessing the unencrypted data by ensuring the data is encrypted when on disk.

Scenario

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, and Baltic seas, as well as the British Isles. To support those efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark, and the U.K.

The company uses Microsoft Azure to store corporate data in the cloud. This may include customer and/or employee information such as:

- addresses
- phone numbers
- tax identification numbers
- credit card information

The company must protect the privacy of customer and employee data while making data accessible to those departments that need it. (such as payroll and reservations departments)

The cruise line also maintains a large database of reward and loyalty program members that includes personal information to track relationships with current and past customers.

Problem statement

The company must protect the privacy of customers' and employees' personal data while making data accessible to those departments that need it (such as payroll and reservations departments). This personal data is stored outside of the corporate-controlled data center and is not under the company's physical control.

Company goal

As part of a multi-layered defense-in-depth security strategy, it is a company goal to ensure that all data sources that contain personal data are encrypted, including those residing in cloud storage. If unauthorized persons gain access to the personal data, it must be in a form that will render it unreadable. Applying encryption should be easy, or transparent – for users and administrators.

Solutions

Azure services provide multiple tools and technologies to help you protect personal data at rest by encrypting it.

Azure Key Vault

[Azure Key Vault](#) provides secure storage for the keys used to encrypt data at rest in Azure services and is the recommended key storage and management solution. Encryption key management is essential to securing stored data.

How do I use Azure Key Vault to protect keys that encrypt personal data?

To use Azure Key Vault, you need a subscription to an Azure account. You also need Azure PowerShell installed.

Steps include using PowerShell cmdlets to do the following:

1. Connect to your subscriptions
2. Create a key vault
3. Add a key or secret to the key vault
4. Register applications that will use the key vault with Azure Active Directory
5. Authorize the applications to use the key or secret

To create a key vault, use the `New-AzureRmKeyVault` PowerShell Cmdlet. You will assign a vault name, resource group name, and geographic location. You'll use the vault name when managing keys via other Cmdlets.

Applications that use the vault through the REST API will use the vault URI.

Azure Key Vault can provide a software-protected key for you, or you can import an existing key in a .PFX file. You can also store secrets (passwords) in the vault.

You can also generate a key in your local HSM and transfer it to HSMs in the Key Vault service, without the key leaving the HSM boundary.

For detailed instructions on using Azure Key Vault, follow the steps in [Get Started with Azure Key Vault](#).

For a list of PowerShell Cmdlets used with Azure Key Vault, see [AzureRM.KeyVault](#).

Azure Disk Encryption for Windows

[Azure Disk Encryption for Windows and Linux IaaS VMs](#) protects personal data at rest on Azure virtual machines and integrates with Azure Key Vault. Azure Disk Encryption uses [BitLocker](#) in Windows and [DM-Crypt](#) in Linux to encrypt both the OS and the data disks. Azure Disk Encryption is supported on Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, and on Windows 8 and Windows 10 clients.

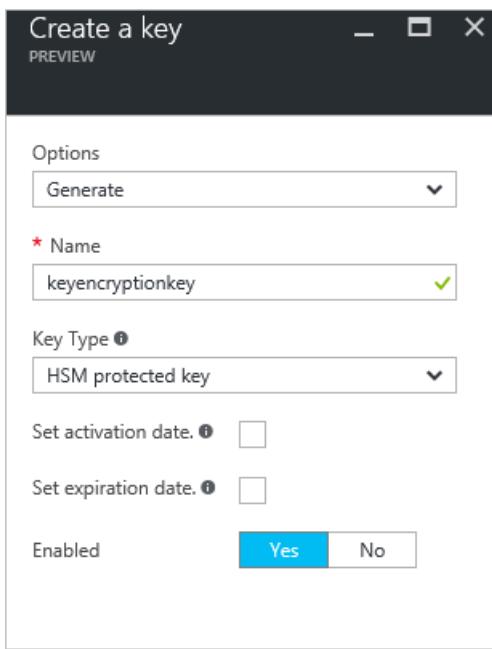
How do I use Azure Disk Encryption to protect personal data?

To use Azure Disk Encryption, you need a subscription to an Azure account. To enable Azure Disk Encryption for Windows and Linux VMs, do the following:

1. Use the Azure Disk Encryption Resource Manager template, PowerShell, or the command line interface (CLI) to enable disk encryption and specify the encryption configuration.
2. Grant access to the Azure platform to read the encryption material from your key vault.
3. Provide an Azure Active Directory (AAD) application identity to write the encryption key material to your key vault.

Azure will update the VM and the key vault configuration, and set up your encrypted VM.

When you set up your key vault to support Azure Disk Encryption, you can add a key encryption key (KEK) for added security and to support backup of encrypted virtual machines.



Detailed instructions for specific deployment scenarios and user experiences are included in [Azure Disk Encryption for Windows and Linux IaaS VMs](#).

Azure Storage Service Encryption

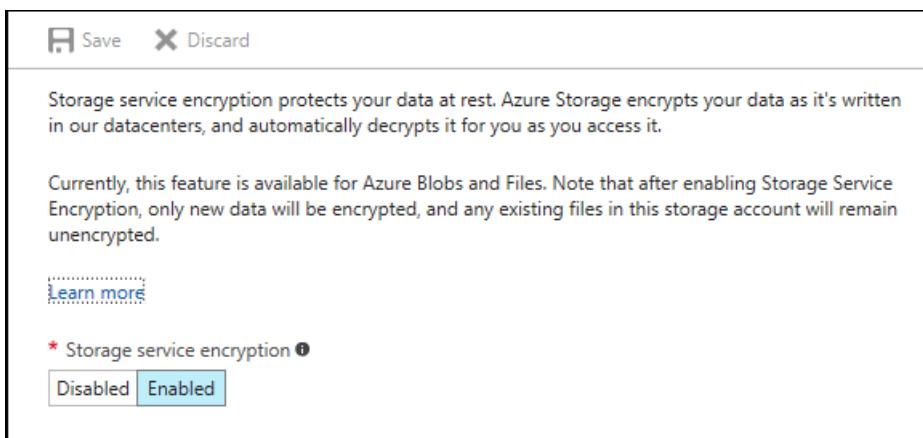
[Azure Storage Service Encryption \(SSE\) for Data at Rest](#) helps you protect and safeguard your data to meet your organizational security and compliance commitments. Azure Storage automatically encrypts your data using 256-bit AES encryption prior to persisting to storage, and decrypts it prior to retrieval. This service is available for Azure Blobs and Files.

How do I use Storage Service Encryption to protect personal data?

To enable Storage Service Encryption, do the following:

1. Log into the Azure portal.
2. Select a storage account.
3. In Settings, under the Blob Service section, select Encryption.
4. Under the File Service section, select Encryption.

After you click the Encryption setting, you can enable or disable Storage Service Encryption.



New data will be encrypted. Data in existing files in this storage account will remain unencrypted.

After enabling encryption, copy data to the storage account using one of the following methods:

1. Copy blobs or files with the [AzCopy Command Line utility](#).

2. Mount a file share using [SMB](#) so you can use a utility such as Robocopy to copy files.
3. Copy blob or file data to and from blob storage or between storage accounts using [Storage Client Libraries such as .NET](#).
4. Use a [Storage Explorer](#) to upload blobs to your storage account with encryption enabled.

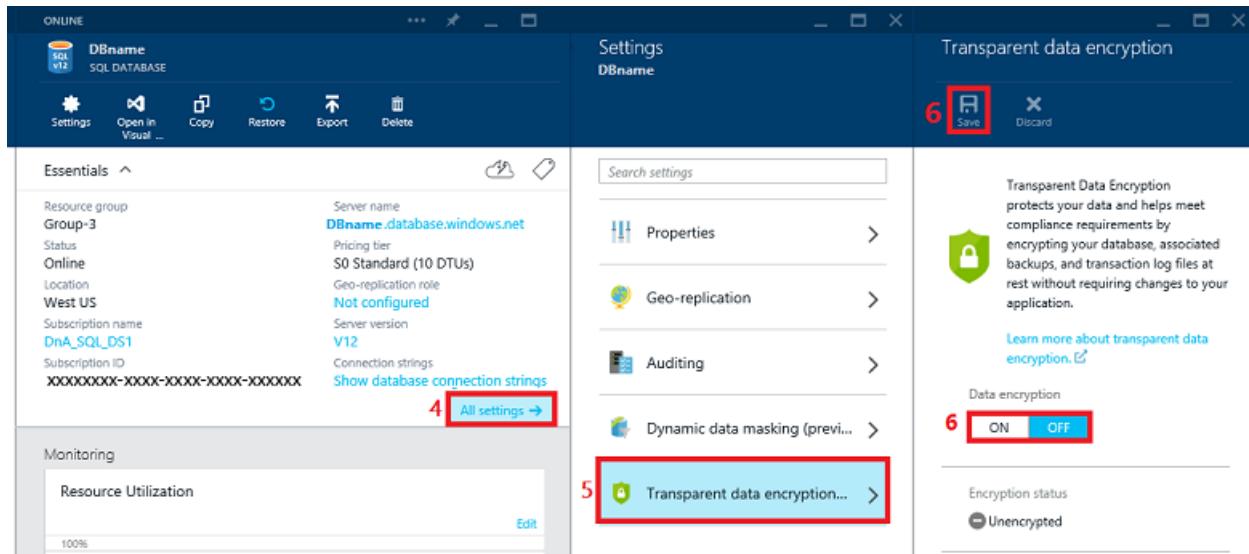
Transparent Data Encryption

Transparent Data Encryption (TDE) is a feature in SQL Azure by which you can encrypt data at both the database and server levels. TDE is now enabled by default on all newly created databases. TDE performs real-time I/O encryption and decryption of the data and log files.

How do I use TDE to protect personal data?

You can configure TDE through the Azure portal, by using the REST API, or by using PowerShell. To enable TDE on an existing database using the Azure Portal, do the following:

1. Visit the Azure portal at <https://portal.azure.com> and sign-in with your Azure Administrator or Contributor account.
2. On the left banner, click to BROWSE, and then click SQL databases.
3. With SQL databases selected in the left pane, click your user database.
4. In the database blade, click All settings.
5. In the Settings blade, click Transparent data encryption part to open the Transparent data encryption blade.
6. In the Data encryption blade, move the Data encryption button to On, and then click Save (at the top of the page) to apply the setting. The Encryption status will approximate the progress of the transparent data encryption.



Instructions on how to enable TDE and information on decrypting TDE-protected databases and more can be found in the article [Transparent Data Encryption with Azure SQL Database](#).

Summary

The company can accomplish its goal of encrypting personal data stored in the Azure cloud. They can do this by using Azure Disk Encryption to protect entire volumes. This may include both the operating system files and data files that hold personal identifiable information and other sensitive data. Azure Storage Service encryption can be used to protect personal data that is stored in blobs and files. For data that is stored in Azure SQL databases, Transparent Data Encryption provides protection from unauthorized exposure of personal information.

To protect the keys that are used to encrypt data in Azure, the company can use Azure Key Vault. This streamlines

the key management process and enables the company to maintain control of keys that access and encrypt personal data.

Next steps

- [Azure Disk Encryption Troubleshooting Guide](#)
- [Encrypt an Azure Virtual Machine](#)
- [Encryption of data in Azure Data Lake Store](#)
- [Azure Cosmos DB database encryption at rest](#)

Azure encryption technologies: Protect personal data in transit with encryption

8/30/2017 • 7 min to read • [Edit Online](#)

This article will help you understand and use Azure encryption technologies to secure data in transit.

Protecting the privacy of personal data as it travels across the network is an essential part of a multi-layered defense-in-depth security strategy. Encryption in transit is designed to prevent an attacker who intercepts transmissions from being able to view or use the data.

Scenario

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, Adriatic, and Baltic seas, as well as the British Isles. To support those efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark and the U.K.

The company uses Microsoft Azure to store corporate data in the cloud. This includes personal identifiable information such as names, addresses, phone numbers, and credit card information of its global customer base. It also includes traditional Human Resource information such as addresses, phone numbers, tax identification numbers and other information about company employees in all locations. The cruise line also maintains a large database of reward and loyalty program members that includes personal information to track relationships with current and past customers.

Personal data of customers is entered in the database from the company's remote offices and from travel agents located around the world. Documents containing customer information are transferred across the network to Azure storage.

Problem statement

The company must protect the privacy of customers' and employees' personal data while it is in transit to and from Azure services.

Company goal

The company goal to ensure that personal data is encrypted when off disk. If unauthorized persons intercept the off-disk personal data, it must be in a form that will render it unreadable. Applying encryption should be easy, or completely transparent, for users and administrators.

Solutions

Azure services provide multiple tools and technologies to help you protect personal data in transit.

Azure Storage

Data that is stored in the cloud must travel from the client, which can be physically located anywhere in the world, to the Azure data center. When that data is retrieved by users, it travels again, in the opposite direction. Data that is in transit over the public Internet is always at risk of interception by attackers. It is important to protect the privacy of personal data by using transport-level encryption to secure it as it moves between locations.

The HTTPS protocol provides a secure, encrypted communications channel over the Internet. HTTPS should be used to access objects in Azure Storage and when calling REST APIs. You enforce use of the HTTPS protocol when using

Shared Access Signatures (SAS) to delegate access to Azure Storage objects. There are two types of SAS: Service SAS and Account SAS.

How do I construct a Service SAS?

A Service SAS delegates access to a resource in just one of the storage services (blob, queue, table or file service). To construct a Service SAS, do the following:

1. Specify the Signed Version Field
2. Specify the Signed Resource (Blob and File Service Only)
3. Specify Query Parameters to Override Response Headers (Blob Service and File Service Only)
4. Specify the Table Name (Table Service Only)
5. Specify the Access Policy
6. Specify the Signature Validity Interval
7. Specify Permissions
8. Specify IP Address or IP Range
9. Specify the HTTP Protocol
10. Specify Table Access Ranges
11. Specify the Signed Identifier
12. Specify the Signature

For more detailed instructions, see [Constructing a Service SAS](#).

How do I construct an Account SAS?

An Account SAS delegates access to resources in one or more of the storage services. You can also delegate access to read, write, and delete operations on blob containers, tables, queues, and file shares that are not permitted with a service SAS. Construction of an Account SAS is similar to that of a Service SAS. For detailed instructions, see [Constructing an Account SAS](#).

How do I enforce HTTPS when calling REST APIs?

To enforce the use of HTTPS when calling REST APIs to access objects in storage accounts, you can enable Secure Transfer Required for the storage account.

1. In the Azure portal, select **Create Storage Account**, or for an existing storage account, select **Settings** and then **Configuration**.
2. Under **Secure Transfer Required**, select **Enabled**.

Create storage account □ X

The cost of your storage account depends on the usage and the options you choose below.
[Learn more](#)

* Name ?
requiresecurexfer ✓
.core.windows.net

Deployment model ?
 Resource manager Classic

Account kind ?

Performance ?

Replication ?

* Storage service encryption (blobs and files) ?
 Disabled Enabled

* Secure transfer required ?
 Disabled Enabled

* Subscription

* Resource group ?
 Create new Use existing

* Location

Pin to dashboard

Create [Automation options](#)

For more detailed instructions, including how to enable Secure Transfer Required programmatically, see [Require Secure Transfer](#).

How do I encrypt data in Azure File Storage?

To encrypt data in transit with [Azure File Storage](#), you can use SMB 3.x with Windows 8, 8.1, and 10 and with Windows Server 2012 R2 and Windows Server 2016. When you are using the Azure Files service, any connection without encryption fails when "Secure transfer required" is enabled. This includes scenarios using SMB 2.1, SMB 3.0 without encryption, and some flavors of the Linux SMB client.

Azure Client-Side Encryption

Another option for protecting personal data while it's being transferred between a client application and Azure Storage is [Client-side Encryption](#). The data is encrypted before being transferred into Azure Storage and when you retrieve the data from Azure Storage, the data is decrypted after it is received on the client side.

Azure Site-to-Site VPN

An effective way to protect personal data in transit between a corporate network or user and the Azure virtual network is to use a [site-to-site](#) or [point-to-site](#) Virtual Private Network (VPN). A VPN connection creates a secure encrypted tunnel across the Internet.

How do I create a site-to-site VPN connection?

A site-to-site VPN connects multiple users on the corporate network to Azure. To create a site-to-site connection in

the Azure portal, do the following:

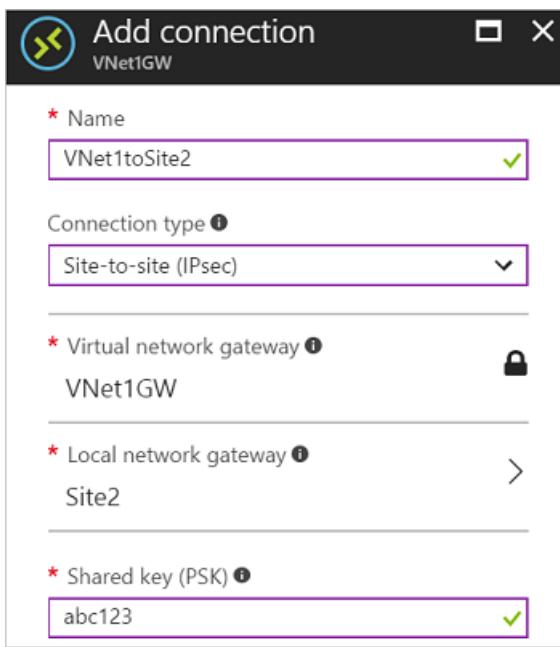
1. Create a virtual network.
2. Specify a DNS server.
3. Create the gateway subnet.
4. Create the VPN gateway.

This screenshot shows the first step of the 'Create virtual network gateway' wizard. It includes fields for Name (VNet1GW), Gateway type (VPN selected), VPN type (Route-based selected), SKU (dropdown menu), Virtual network (TestVNet1), and Public IP address (new VNet1GWIP).

5. Create the local network gateway.

This screenshot shows the first step of the 'Create local network gateway' wizard. It includes fields for Name (Site2), IP address (40.71.184.214), Address space (192.168.3.0/24), Subscription (Windows Azure Internal Consumption), Resource group (TestRG1 selected), and Location (East US).

6. Configure your VPN device.
7. Create the VPN connection.



8. Verify the VPN connection.

For more detailed instructions on how to create a site-to-site connection in the Azure portal, see [Create a Site-to-Site connection in the Azure Portal](#).

How do I create a point-to-site VPN connection?

A Point-to-Site VPN creates a secure connection from an individual client computer to a virtual network. This is useful when you want to connect to Azure from a remote location, such as from home or a hotel or conference center. To create a point-to-site connection in the Azure portal,

1. Create a virtual network.
2. Add a gateway subnet.
3. Specify a DNS server. (optional)
4. Create a virtual network gateway.
5. Generate certificates.
6. Add the client address pool.
7. Upload the root certificate public certificate data.
8. Generate and install the VPN client configuration package.
9. Install an exported client certificate.
10. Connect to Azure.
11. Verify your connection.

For more detailed instructions, see [Configure a Point-to-Site connection to a VNet using certificate authentication: Azure Portal](#).

SSL/TLS

Microsoft recommends that you always use SSL/TLS protocols to exchange data across different locations.

Organizations that choose to use [ExpressRoute](#) to move large data sets over a dedicated high-speed WAN link can also encrypt the data at the application-level using SSL/TLS or other protocols for added protection.

Encryption by default

Microsoft uses encryption to protect data in transit between customers and Azure cloud services. If you are

interacting with Azure Storage through the Azure Portal, all transactions occur via HTTPS.

[Transport Layer Security](#) (TLS) is the protocol that Microsoft data centers will attempt to negotiate with client systems that connect to Microsoft cloud services. TLS provides strong authentication, message privacy, and integrity (enables detection of message tampering, interception, and forgery), interoperability, algorithm flexibility, ease of deployment and use.

[Perfect Forward Secrecy](#) (PFS) is also employed so that each connection between customers' client systems and Microsoft's cloud services use unique keys. Connections to Microsoft cloud services also take advantage of RSA based 2,048-bit encryption key lengths. The combination of TLS, RSA 2,048-bit key lengths, and PFS makes it much more difficult for someone to intercept and access data that is in transit between Microsoft cloud services and customers.

Data in transit is always encrypted in [Data Lake Store](#). In addition to encrypting data prior to storing to persistent media, the data is also always secured in transit by using HTTPS. HTTPS is the only protocol that is supported for the Data Lake Store REST interfaces.

Summary

The company can accomplish its goal of protecting personal data and the privacy of such data by enforcing HTTPS connections to Azure Storage, using Shared Access Signatures and enabling Secure Transfer Required on the storage accounts. They can also protect personal data by using SMB 3.0 connections and implementing client-side encryption. Site-to-site VPN connections from the corporate network to the Azure virtual network and point-to-site VPN connections from individual users will create a secure tunnel through which personal data can securely travel. Microsoft's default encryption practices will further protect the privacy of personal data.

Next steps

- [Azure Data Security and Encryption Best Practices](#)
- [Planning and design for VPN Gateway](#)
- [VPN Gateway FAQ](#)
- [Buy and configure an SSL Certificate for your Azure App Service](#)

Document protection of personal data with Azure reporting tools

8/30/2017 • 12 min to read • [Edit Online](#)

This article will discuss how to use Azure reporting services and technologies to help protect privacy of personal data.

Scenario

A large cruise company, headquartered in the United States, is expanding its operations to offer itineraries in the Mediterranean, Adriatic, and Baltic seas, as well as the British Isles. To help these efforts, it has acquired several smaller cruise lines based in Italy, Germany, Denmark and the U.K.

The company uses Microsoft Azure for processing and storage of corporate data. This includes personal identifiable information such as names, addresses, phone numbers, and credit card information of its global customer base. It also includes traditional Human Resources information such as addresses, phone numbers, tax identification numbers and other information about company employees in all locations. The cruise line also maintains a large database of reward and loyalty program members that includes personal information to track relationships with current and past customers.

Corporate employees access the network from the company's remote offices and travel agents located around the world have access to some company resources.

Problem statement

The company must protect the privacy of customers' and employees' and personal data through a multi-layered security strategy that uses Azure management and security features to impose strict controls on access to and processing of personal data, and must be able to demonstrate its protective measures to internal and external auditors.

Company goal

As part of its defense-in-depth security strategy, it is a company goal to track all access to and processing of personal data, and ensure that documentation of adequate privacy protections for personal data are in place and working.

Solutions

Microsoft Azure provides comprehensive monitoring, logging, and diagnostics tools to help track and record activities and events associated with accessing and processing personal data, geographic flow of data, and third-party access to personal data. Because security of personal data in the cloud is a shared responsibility, Microsoft also provides customers with:

- Detailed information about its own processing of customers' data
- Security measures administered by Microsoft
- Where and how it sends customers' data
- Details of Microsoft's own privacy reviews process

Azure Active Directory

Azure Active Directory is Microsoft's cloud-based, multi-tenant directory and identity management service. The service's sign-in and audit reporting capabilities provide you with detailed sign-in and application usage activity information to help you monitor and ensure proper access to customers' and employees' personal data.

There are two types of activity reports:

- The [audit activity reports/logs](#) provide a detailed record of system activities/tasks
- The [sign-ins activity report/log](#) shows you who has performed each activity listed in the audit report

Using the two together, you can track the history of every task performed and who performed each. Both types of reports are customizable and can be filtered.

How do I access the audit and security logs?

The audit and security logs can be accessed from the Active Directory portal in three different ways: through the **Activity** section (select either **Audit logs** or **Sign-ins**), or from **Users and groups** or **Enterprise applications** under **Manage** in Active Directory. Reports can also be accessed through the Azure Active Directory reporting API.

1. In the Azure portal, select **Azure Active Directory**.

2. In the **Activity** section, select **Audit logs**.

DATE	INITIATED BY (ACTOR)	ACTIVITY	TARGET(S)
03-04-2017 12:26:17	Azure AD Cloud Sync	Import	ServicePrincipal : a36af6e8-ae03-402
03-04-2017 12:26:16	Azure AD Cloud Sync	Process escrow	ServicePrincipal : a36af6e8-ae03-402
03-04-2017 12:13:32	Azure AD Cloud Sync	Export	ServicePrincipal : 792a68a0-73fa-445

3. Customize the list view by clicking **Columns** in the toolbar.

4. Select an item in the list view to see all available details about it.

Activity Details: Audit log

Activity

Date : 4/3/2017 12:26:16
Name : Process escrow
CorrelationId : f65b338b-591d-489b-bfac-9840fc4b04f8
Category : Account Provisioning

Activity Status

Status : Failure
Reason : We will attempt to retry an operation that previously failed on Group 'My Demo Apps';
Error: This object is to be re-synchronized: . We will retry this operation on the next synchronization attempt

Initiated By (Actor)

Type : Application
Name : Azure AD Cloud Sync
ObjectId :

Target(s)

Target
Type : ServicePrincipal
ObjectId : a36af6e8-ae03-402b-8582-b3d7a811f8aa

Target
Type : Group
Name : My Demo Apps

Additional Details

Details : Error was originally encountered at 2017-03-30 07:39:39Z.
ErrorCode : Retry
EventName : EntryEscrowProcess
JoiningProperty : My Demo Apps
SourceAnchor : 5dfa2046-6169-4589-bc44-0c517b388686

Azure Active Directory reporting also includes two types of security reports, **users flagged for risk** and **risky sign-ins**, which can help you monitor potential risks in your Azure environment.

For more information about the reporting service, see [Azure Active Directory reporting](#)

Visit [Azure Active Directory activity reports](#) for more specifics about the reports available in Azure Active Directory. This site includes more details about how to access and use [audit logs activity reports](#) and [sign-in activity reports](#) in the portal. It also includes information about [users flagged for risk](#) and [risky sign-in](#) security reports.

Visit the [Azure Active Directory audit API reference](#) site for more information on how to connect to Azure Directory reporting programmatically.

Log Analytics

[Log Analytics](#) can collect data from [Azure Monitor](#) to correlate it with other data and provide additional analysis. Azure Monitor collects and analyzes monitoring data for your Azure environment.

Analysis tools in Log Analytics such as log searches, views, and solutions work against all collected data, providing you with centralized analysis of your entire environment. Log Analytics can aggregate and analyze Windows Event logs, IIS logs, and Syslogs, which can help detect potential personal data breaches that could expose personal data to unauthorized users.

How do I use Log Analytics?

You can access Log Analytics through the OMS portal or the Azure portal, from any web browser. Log Analytics includes a query language to quickly retrieve and consolidate data in the repository. You can create and save Log Searches to directly analyze data in the portal.

To create a Log Analytics workspace in the Azure portal, do the following:

1. Select **Log Analytics** from the list of services in the Marketplace.
2. Select **Create**, then specify the name of your OMS workspace, select your subscription, resource group, location, and pricing tier.
3. Click **OK** to display a list of your workspaces.
4. Select a workspace to see its details.

The screenshot shows the Azure Log Analytics workspace details page for 'bandersworkspace3'. The page has a left sidebar with navigation links like 'Search (Ctrl+)', 'OMS Workspace', 'Activity log', 'Access control (IAM)', 'Tags', 'SETTINGS', 'Locks', 'Automation script', 'GENERAL', 'Quick Start', 'Saved searches', 'Log Search', 'Solutions', 'Pricing tier', 'Log Analytics usage', and 'Properties'. The main content area is titled 'Essentials' and displays workspace details: Resource group 'banderserver1', Workspace Name 'bandersworkspace3', Status 'Active', Location 'East US', Subscription name 'Visual Studio Enterprise', Subscription ID '2894fb4b-6011-4122-8f7e-8042d4000001', Pricing tier 'Free', Management services 'Operations logs', and a 'Management' section with 'Overview' and 'Log Search' buttons. Below this is a 'Pricing tier' section showing 'Free' and 'BANDERSWORKSPACE3' with icons for CPU, Memory, Storage, and Network. At the bottom is a 'F' icon indicating '500 MB daily limit' and 'Data retention 7 days'.

Visit the [Log Analytics documentation](#) to learn more about the service.

Visit the [Get started with a Log Analytics workspace](#) tutorial to create an evaluation workspace and learn the basics of how to use the service.

Visit the following web pages for more specific information on how to connect to use Log Analytics with the logs described above:

[Windows event logs data sources in Log Analytics](#)

[IIS logs in Log Analytics](#)

[Syslog data sources in Log Analytics](#)

Azure Monitor/Azure Activity Log

[Azure Monitor](#) provides base level infrastructure metrics and logs for most services in Microsoft Azure. Monitoring can help you to gain deep insights about your Azure applications. Azure Monitor relies on the Azure diagnostics extension (Windows or Linux) to collect most application level metrics and logs. [The Azure Activity Log](#) is one of the resources you can view with Azure Monitor. It tracks every API call, and provides a wealth of information about activities that occur in [Azure Resource Manager](#). You can search the Activity Log (previously called Operational or Audit Logs) for information about your resource as seen by the Azure infrastructure.

Although much of the information recorded in the Activity log pertains to performance and service health, there is also information that is related to protection of data. Using the Activity Log, you can determine the "what, who, and when" for any write operations (PUT, POST, DELETE) taken on the resources in your Azure subscription.

For example, it provides a record when an administrator deletes a network security group, which could impact the protection of personal data. Activity log entries are stored in Azure Monitor for 90 days.

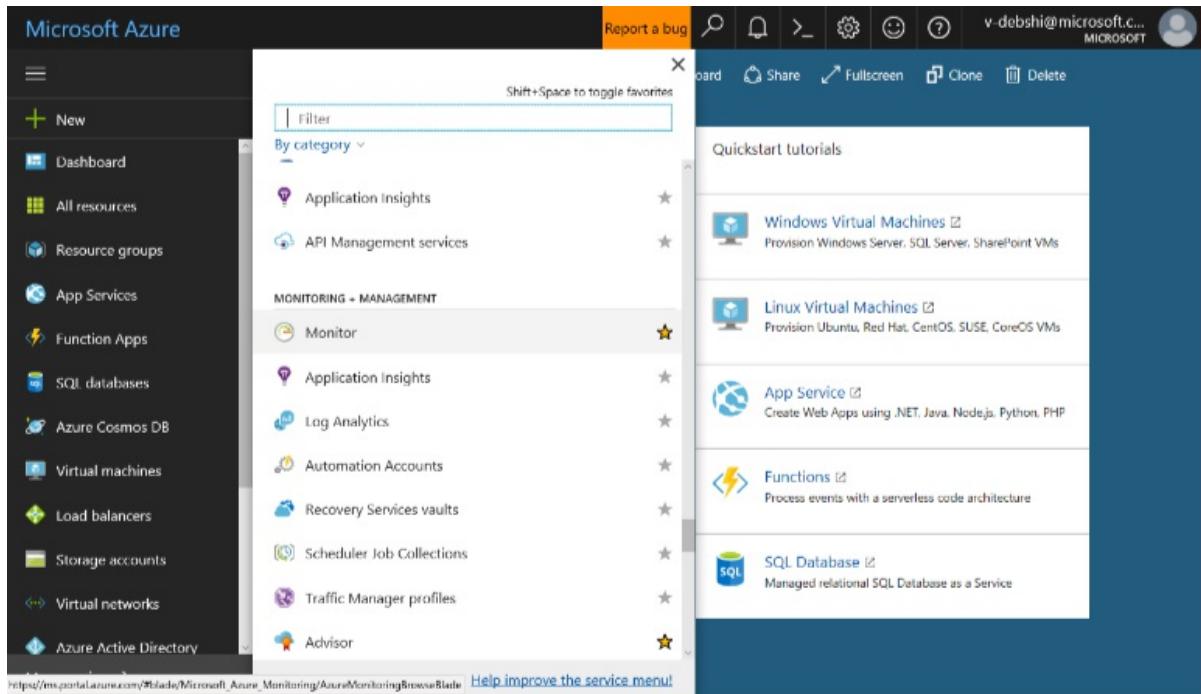
How do I use the data collected by Azure Monitor?

There are a number of ways to use the data in the Activity log and other Azure Monitor resources.

- You can stream the data to other locations in real time.
- You can store the data for longer time periods than the defaults, using an [Azure storage account](#) and setting a retention policy.
- You can visualize the data in graphics and charts, using the [Azure portal](#), [Azure Application Insights](#), [Microsoft PowerBI](#), or third-party visualization tools.
- You can query the data using the Azure Monitor REST API, CLI commands, [PowerShell cmdlets](#), or the .NET SDK.

To get started with Azure Monitor, select **More Services** in the Azure portal.

1. Scroll down to **Monitor** in the **Monitoring and Managing** section.



2. Monitor opens in the **Activity Log** view.

Monitor - Activity log

Microsoft - PREVIEW

Columns Export Log search

Search (Ctrl+ /)

EXPLORE

- Activity log
- Metrics
- Diagnostics logs
- Log search
- Service notifications

MANAGE

- Alerts
- Notification groups

ADVANCED

- Application Insights
- Management solutions

Try out Log Analytics to get activity log solution p

Select query ...

* Subscription Microsoft Azure Inter... Resource group All resource groups All resource groups

Timespan Last 1 hour Event category * Event

All categories 4 sele

Apply Reset

Query returned 40 items. [Click here to download all the items as csv](#)

OPERATION NAME	STATUS	TIME
ListKeys	Succeeded	Just now
ListKeys	Succeeded	3 min ago
ListKeys	Succeeded	3 min ago
ListKeys	Succeeded	6 min ago
ListKeys	Succeeded	8 min ago
ListKeys	Succeeded	8 min ago

You can create and save queries for common filters, then pin the most important queries to a portal dashboard so you'll always know if events that meet your criteria have occurred.

1. You can filter the view by resource group, timespan, and event category.

Try out Log Analytics to get activity log solution p

Select query ...

* Subscription Visual Studio Ultimat... Resource group JohnTest

Timespan Last week Event category All categories

Apply Reset

2. You can then pin queries to a portal dashboard by clicking the **Pin** button. This helps you create a single source of information for operational data on your services. The query name and number of results will be displayed on the dashboard.

You can also use the Monitor to view metrics for all Azure resources, configure diagnostics settings and alerts, and search the log. For more information on how to use the Azure Monitor and Activity Log, see [Get Started with Azure Monitor](#).

Azure Diagnostics

The diagnostics capability in Azure enables collection of data from several sources. The Windows Event logs, which include the Security log, can be especially useful in tracking and documenting protection of personal data. The security log tracks logon success and failure events, as well as permissions changes, detection of patterns indicating certain types of attacks, changes to security-related policies, security group membership changes, and much more.

For example, Event ID 4695 alerts you to the attempted unprotection of auditable protected data. This pertains to

the Data Protection API (DPAPI), which helps to protect data such as private keys, stored credentials, and other confidential information.

How do I enable the diagnostics extension for Windows VMs?

You can use PowerShell to enable the diagnostics extension for a Windows VM, so as to collect log data. The steps for doing so depend on which deployment model you use (Resource Manager or Classic). To enable the diagnostics extension on an existing VM that was created through the Resource Manager deployment model, you can use the [Set-AzureRmVMDiagnosticExtension PowerShell cmdlet](#).



```
$vm_resourcegroup = "myvmresourcegroup"
$vm_name = "myvm"
$diagnosticsconfig_path = "DiagnosticsPubConfig.xml"

Set-AzureRmVMDiagnosticExtension -ResourceGroupName $vm_resourcegroup -VMName $vm_name -DiagnosticsConf
```

\\$diagnosticsconfig_path is the path to the file that contains the diagnostics configuration in XML. For more detailed instructions on enabling Azure Diagnostics on a VM, see [Use PowerShell to enable Azure Diagnostics in a virtual machine running Windows](#).

The Azure diagnostics extension can transfer the collected data to an Azure storage account or send it to services such as Application Insights. You can then use the data for auditing.

How do I store and view diagnostic data?

It's important to remember that diagnostic data is not permanently stored unless you transfer it to the Microsoft Azure storage emulator or to Azure storage. To store and view diagnostic data in Azure Storage, follow these steps:

1. Specify a storage account in the ServiceConfiguration.cscfg file. Azure Diagnostics can use either the Blob service or the Table service, depending on the type of data. Windows Event logs are stored in Table format.
2. Transfer the data. You can request to transfer the diagnostic data through the configuration file. For SDK 2.4 and previous, you can also make the request programmatically.
3. View the data, using [Azure Storage Explorer](#), [Server Explorer](#) in Visual Studio, or [Azure Diagnostics Manager](#) in Azure Management Studio.

For more information on how to perform each of these steps, see [Store and view diagnostic data in Azure Storage](#).

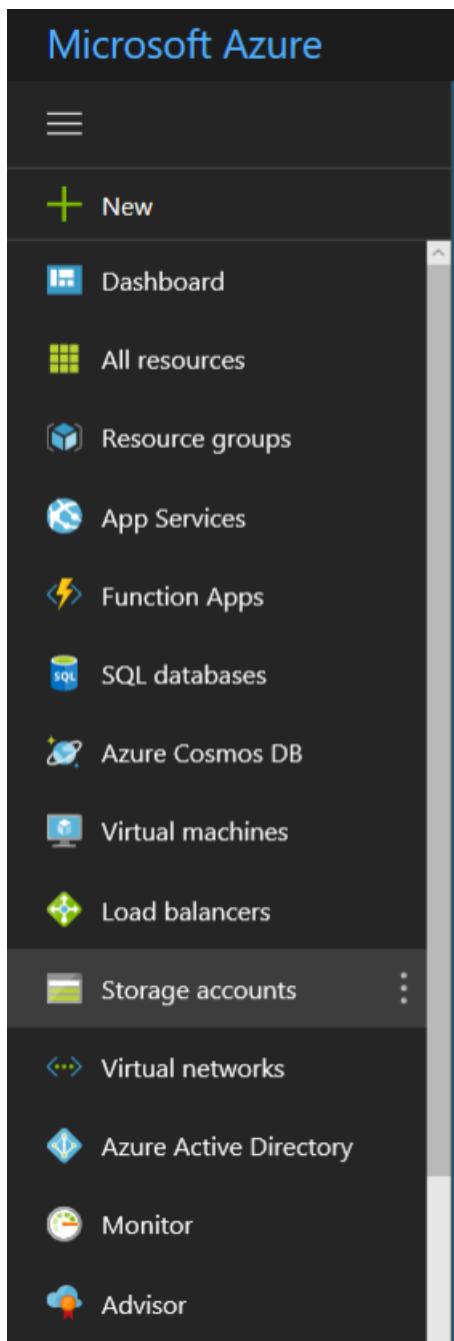
Azure Storage Analytics

Storage Analytics logs detailed information about successful and failed requests to a storage service. This information can be used to monitor individual requests, which can help in documenting access to personal data stored in the service. However, Storage Analytics logging is not enabled by default for your storage account. You can enable it in the Azure portal.

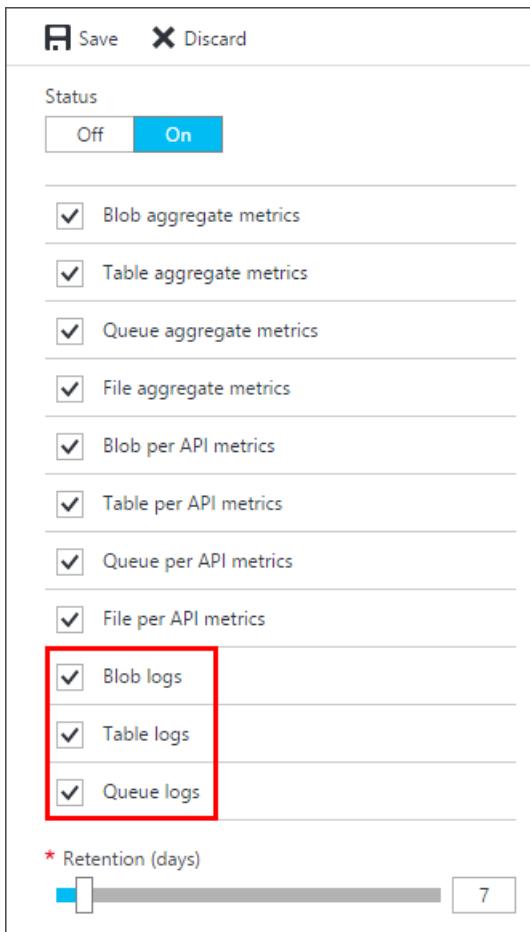
How do I configure monitoring for a storage account?

To configure monitoring for a storage account, do the following:

1. Select **Storage accounts** in the Azure portal, then select the name of the account you want to monitor.



2. In the **Monitoring** section, select **Diagnostics**.
3. Select the **type** of metrics data you want to monitor for each service (Blob, Table, File). To instruct Azure Storage to save diagnostics logs for read, write, and delete requests for the blob, table, and queue services, select **Blob logs**, **Table logs** and **Queue logs**.



4. Using the slider at the bottom, set the **retention** policy in days (value of 1 – 365). Seven days is the default.
5. Select **Save** to apply the configuration settings.

Storage Logging log entries contain the following information about individual requests:

- Timing information such as start time, end-to-end latency, and server latency.
- Details of the storage operation such as the operation type, the key of the storage object the client is accessing, success or failure, and the HTTP status code returned to the client.
- Authentication details such as the type of authentication the client used.
- Concurrency information such as the ETag value and last modified timestamp.
- The sizes of the request and response messages.

For more detailed instructions on how to enable Storage Analytics logging, see [Monitor a storage account in the Azure portal](#).

Azure Security Center

[Azure Security Center](#) monitors the security state of your Azure resources in order to prevent and detect threats, and provide recommendations for responding. It provides several ways to help document your security measures that protect the privacy of personal data.

Security health monitoring helps you ensure compliance with your security policies. Security monitoring is a proactive strategy that audits your resources to identify systems that do not meet organizational standards or best practices. You can monitor the security state of the following resources:

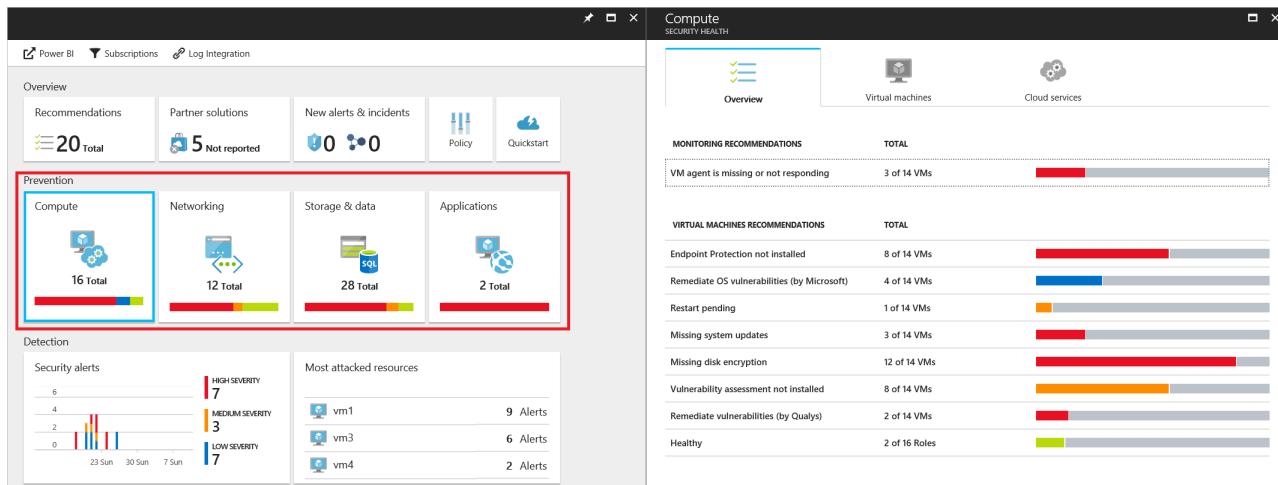
- Compute (virtual machines and cloud services)
- Networking (virtual networks)

- Storage and data (server and database auditing and threat detection, TDE, storage encryption)
- Applications (potential security issues)

Security issues in any of these categories could pose a threat to the privacy of personal data.

How do I view the security state of my Azure resources?

Security Center periodically analyzes the security state of your Azure resources. You can view any potential security vulnerabilities it identifies in the **Prevention** section of the dashboard.



1. In the **Prevention** section, select the **Compute** tile. You'll see here an **Overview**, along with the **Virtual machines** listing of all VMs and their security states, and the **Cloud services** list of web and worker roles monitored by Security Center.
2. On the **Overview** tab, second a recommendation to view more information.
3. On the **Virtual machines** tab, select a VM to view additional details.

When data collection is enabled in Azure Security Center, the Microsoft Monitoring Agent is automatically provisioned on all existing and any new supported virtual machines that are deployed. Data collected from this agent is stored in either an existing [Log Analytics](#) workspace associated with your subscription or a new workspace.

[Threat Intelligence Reports](#) are provided by Security Center. These give you useful information to help discern the attacker's identity, objectives, current and historical attack campaigns, and tactics, tools and procedures used. Mitigation and remediation information is also included.

The primary purpose of these threat reports is to help you to respond effectively to the immediate threat and help take measures afterward to mitigate the issue. The information in the reports can also be useful when you document your incident response for reporting and auditing purposes.

The Threat Intelligence Reports are presented in .PDF format, accessed via a link in the **Reports** field of the **Suspicious process executed** blade for each security alert in Azure Security Center.

For more information on how to view and use the Threat Intelligence Report, see [Azure Security Center Threat Intelligence Report](#).

Next Steps:

[Getting Started with the Azure Active Directory reporting API](#)

[What is Log Analytics?](#)

[Overview of Monitoring in Microsoft Azure](#)

[Introduction to the Azure Activity Log \(video\)](#)

Azure security courses from Microsoft Virtual Academy

6/27/2017 • 3 min to read • [Edit Online](#)

Microsoft Virtual Academy provides free, online training to help Developers, IT and Data Professionals, and students learn the latest technology, build their skills, and advance their careers.

On this page, you find a curated collection of Azure security-related courses. Visit the [Microsoft Virtual Academy](#) to see all the courses they have available.

[Dev/Test in the Cloud](#)

Are you a developer who needs to deliver faster and better applications? Moving your development and testing environments to the cloud can help you achieve exactly that! Learn how to get it done, and find out the benefits of making the move. Plus, see demonstrations and presentations that show you how Microsoft Azure can support your development and testing needs. Included are lesson on security development and deployment practices.

[Common Tasks for Linux on Azure](#)

If you have questions about using Linux on the Microsoft Azure platform, this detailed course has answers for you. Explore some common tasks with the experts from [Linux Academy](#). Learn about creating a Linux virtual machine (VM) in Azure, accessing the Linux VM using remote desktop software, and running virtual hosts. Many security technologies and configurations are addressed in this course.

[Secure the Cloud](#)

In this session, learn how Microsoft can help you meet global compliance requirements, such as ISO 27001 / 27018, FedRAMP, PCI, and HIPAA, with new security controls. These controls range from at-rest data encryption, key management, VM protection, logging and monitoring, to anti-malware services, identity management, access controls, and more.

[Design and Implement Cloud Data Platform Solutions](#)

Learn the features and capabilities of Microsoft cloud data platform solutions. Get a platform overview and hear about security features, options for high availability, techniques for monitoring and managing cloud data, and more. Plus, get guidance on how to identify tradeoffs and make decisions for designing public and hybrid cloud solutions using Microsoft cloud data platform features.

[Manage and Secure Identities in a Cloud and Mobile World](#)

In this session, learn how Azure Active Directory and Microsoft Advanced Threat Analytics helps you secure and manage user identity, identify security breaches before they cause damage, and provide your users a single identity for accessing all corporate resources. Explore the technologies used to discover Shadow IT, manage application access, and monitor suspicious activity through advanced security reporting, user behavioral analytics, auditing, and alerting.

[Security in a Cloud-Enabled World](#)

Experts lead you through the customer responsibility roadmap in the [Microsoft Cloud Security for Enterprise Architects](#) poster. The experts also provide recommendations for modernizing each part of your security posture, including governance, containment strategies, security operations, high-value asset protection, information protection, and user and device security, with a particular emphasis on protecting administrative control. Learn from the same framework that the Microsoft cybersecurity team uses to assess customers' cloud security and to

build them a security roadmap.

[Microsoft Azure IaaS Deep Dive](#)

Learn how to use Microsoft Azure infrastructure capabilities. If you are an IT Pro, no need to have previous experience with Azure. This course walks you through creating and configuring Azure VMs, Azure Virtual Networks, and cross-premises connectivity to get things up and running on the cloud. Security features and considerations are included throughout the course.

[Getting Started with Azure Security for the IT Professional](#)

In this demo-filled course, a team of security experts and Azure engineers takes you beyond the basic certifications and explores what's possible inside Azure. See how to design and use various technologies to ensure that you have the security and architecture you need to successfully launch your projects in the cloud. Dive into datacenter operations, VM configuration, network architecture, and storage infrastructure.

[Deep Dive into Azure Resource Manager Scenarios and Patterns](#)

Explore Azure Resource Manager with a team of experts, who show you scripts and tools that make it easy to spin up or spin down elements of your application infrastructure. Explore the use of role-based access control (RBAC) to implement security with Azure Resource Manager.

[Azure Rights Management Services Core Skills](#)

Find out why information protection is a "must have" requirement in your organization and how rights management protects your organization's intellectual property, wherever it travels across devices and the cloud. Get hands-on experience and technical know-how from Microsoft experts.

Azure security videos on Channel 9

6/27/2017 • 3 min to read • [Edit Online](#)

Channel 9 is a community that brings forward the people behind our products and connects them with customers.

They think there is a great future in software and they're excited about it. Channel 9 is a community to participate in the ongoing conversation.

The following is a curated list of Azure security presentations on Channel 9. Make sure to check this page monthly for new videos.

[Accelerating Azure Consumption with Barracuda Security](#)

See how you can use Barracuda security to secure your Azure deployments.

[Azure Security Center - Threat Detection](#)

With Azure Security Center, you get a central view of the security state of all your Azure resources. At a glance, verify that the appropriate security controls are in place and configured correctly. Scott talks to Sarah Fender who explains how Security Center integrates Threat Detection.

[Azure Security Center Overview](#)

With Azure Security Center, you get a central view of the security state of all your Azure resources. At a glance, verify that the appropriate security controls are in place and configured correctly. Scott talks to Sara Fender who explains it all!

[Live Demo: Protecting against, Detecting and Responding to Threats](#)

Join this session to see the Microsoft security platform in action. General Manager for Cloud & Enterprise, Julia White, demonstrates the security features of Windows 10, Azure, and Office 365 that can help you keep your organization secure.

[Encryption in SQL Server Virtual Machines in Azure for better security](#)

Jack Richins teaches [Scott](#) how to easily encrypt his SQL Server databases on Virtual Machine Azure instances. It's easier than you'd think!

Areas covered in this video:

- Understanding encryption and SQL Server
- Understanding the Data Protection API, master keys, and certificates
- Using SQL commands to create the master key and certificates, and encrypt the database

[How to set security in DevTest Labs](#)

As an owner of your lab, you can secure lab access by via two lab roles: Owner and DevTest Labs User. A person in the Owner role has complete access in the lab whereas a person in the DevTest Labs User role has limited access. In this video, we show you how to add a person in either of these roles to a lab.

[Managing Secrets for Azure Apps](#)

Every serious app you deploy on Azure has critical secrets – connection strings, certificates, keys. Silly mistakes in managing these secrets leads to fatal consequences – leaks, outages, compliance violations. As multiple recent surveys point out, silly mistakes cause four times more data breaches than adversaries. In this session, we go over some best practices to manage your important app secrets. These best practices may seem like common sense, yet

many developers neglect them. We also go over how to use Azure Key Vault to implement those best practices. As an added benefit, following these practices helps you demonstrate compliance with standards such as SOC. The first 10 minutes of the session are level 100 and they apply to any cloud app you develop on any platform. The remainder is level 200-300 and focuses on apps you build on the Azure platform.

[Securing your Azure Virtual Network using Network Security Groups with Narayan Annamalai](#)

Senior Program Manager Narayan Annamalai teaches Scott how to use Network Security Groups within an Azure Virtual Network. You can create control access to objects within Azure by subnet and network! You learn how to control access and create groups within Azure using PowerShell.

[Azure AD Privileged Identity Management: Security Wizard, Alerts, Reviews](#)

Azure Active Directory (AD) Privileged Identity Management is a premium functionality that allows you to discover, restrict, and monitor privileged identities and their access to resources. It also enforces on-demand, just in time administrative access when needed. Learn about:

- Managing protection for Office 365 workload-specific administrative roles
- Configuring Azure Multi-Factor Authentication(MFA) for privileged role activations
- Measuring and improving your tenant security posture
- Monitoring and fixing security findings
- Reviewing who needs to remain in privileged roles for periodic recertification workflows

[Azure Key Vault with Amit Bapat](#)

Amit Bapat introduces Scott to Azure Key Vault. With Azure Key Vault, you can encrypt keys and small secrets like passwords using keys stored in hardware security modules (HSMs). It's cloud-based, hardware-based secret management for Microsoft Azure!

Microsoft Threat Modeling Tool

8/25/2017 • 1 min to read • [Edit Online](#)

The Threat Modeling Tool is a core element of the Microsoft Security Development Lifecycle (SDL). It allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. As a result, it greatly reduces the total cost of development. Also, we designed the tool with non-security experts in mind, making threat modeling easier for all developers by providing clear guidance on creating and analyzing threat models.

The tool enables anyone to:

- Communicate about the security design of their systems
- Analyze those designs for potential security issues using a proven methodology
- Suggest and manage mitigations for security issues

Here are some tooling capabilities and innovations, just to name a few:

- **Automation:** Guidance and feedback in drawing a model
- **STRIDE per Element:** Guided analysis of threats and mitigations
- **Reporting:** Security activities and testing in the verification phase
- **Unique Methodology:** Enables users to better visualize and understand threats
- **Designed for Developers and Centered on Software:** many approaches are centered on assets or attackers. We are centered on software. We build on activities that all software developers and architects are familiar with -- such as drawing pictures for their software architecture
- **Focused on Design Analysis:** The term "threat modeling" can refer to either a requirements or a design analysis technique. Sometimes, it refers to a complex blend of the two. The Microsoft SDL approach to threat modeling is a focused design analysis technique

Next steps

The table below contains important links to get you started with the Threat Modeling Tool:

STEP	DESCRIPTION
1	Download the Threat Modeling Tool
2	Read Our getting started guide
3	Get familiar with the features
4	Learn about generated threat categories
5	Find mitigations to generated threats

Resources

Here are a few older articles still relevant to threat modeling today:

- [Article on the Importance of Threat Modeling](#)
- [Training Published by Trustworthy Computing](#)

Check out what a few Threat Modeling Tool experts have done:

- [Threats Manager](#)
- [Simone Curzi Security Blog](#)

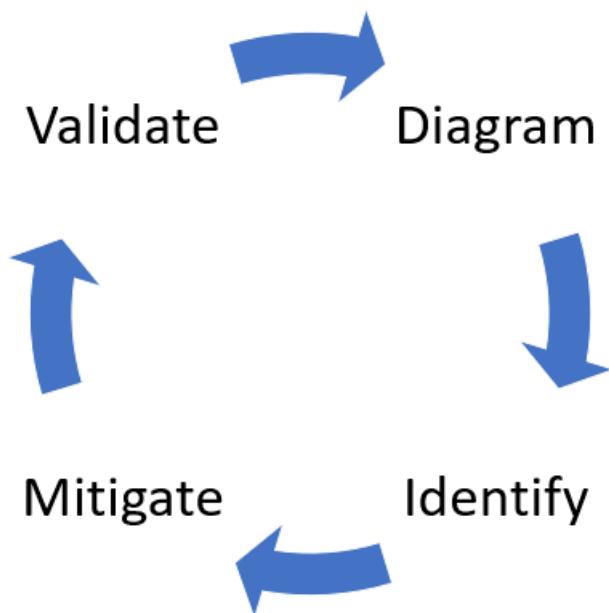
Getting started with the Threat Modeling Tool

8/25/2017 • 7 min to read • [Edit Online](#)

The Cloud and Enterprise Security Tools team released the Threat Modeling Tool Preview earlier this year as a free [click-to-download](#). The change in delivery mechanism allows us to push the latest improvements and bug fixes to customers each time they open the tool, making it easier to maintain and use. This article takes you through the process of getting started with the Microsoft SDL threat modeling approach and shows you how to use the tool to develop great threat models as a backbone of your security process.

This article builds on existing knowledge of the SDL threat modeling approach. For a quick review, refer to [Threat Modeling Web Applications](#) and an archived version of [Uncover Security Flaws Using the STRIDE Approach](#) MSDN article published in 2006.

To quickly summarize, the approach involves creating a diagram, identifying threats, mitigating them and validating each mitigation. Here's a diagram that highlights this process:



Starting the threat modeling process

When you launch the Threat Modeling Tool, you'll notice a few things, as seen in the picture:

MICROSOFT MICROSOFT THREAT MODELING TOOL (PREVIEW)

Threat Model:

[Feedback, Suggestions and Issues](#)

Create A Model

Model your system by drawing diagram (s). Make sure you capture important details.

Open A Model

Open an existing model and analyze threats against your system; do not worry, the tool will help you identify them.

Getting Started Guide

A step-by-step guide to help you get up and running now.

Template For New Models

[Azure Threat Model Template\(1.0.0.20\)](#) [Browse...](#)

Recently Opened Models

[Basic Web App NEW.tm7](#)
[New Threat Model.tm7](#)
[Library Sample.tm7](#)
[Basic Web App Sample.tm7](#)
[QPP_complete19_filtered.tm7](#)
[CloudMobileThreatModel_April2017.tm7](#)

Threat Modeling Workflow

1. Select your template.
2. Create your data flow diagram model.
3. Analyze the model for potential threats.
4. Determine mitigations.

Template:

Create New Template

Define stencils, threat types and custom threat properties for your threat model from scratch.

Open Template

Open an existing Template and make modifications to better suit your specific threat analysis.

Template Workflow

Use templates to define threats that applications should look for.

1. Define stencils
2. Define categories
3. Define threat properties
4. Define threat
5. Share your template

Threat model section

COMPONENT	DETAILS
Feedback, Suggestions and Issues Button	Takes you to the MSDN Forum for all things SDL. It gives you an opportunity to read through what other users are doing, along with workarounds and recommendations. If you still can't find what you're looking for, email tmtextsupport@microsoft.com for our support team to help you
Create a Model	Opens a blank canvas for you to draw your diagram. Make sure to select which template you'd like to use for your model
Template for New Models	You must select which template to use before creating a model. Our main template is the Azure Threat Model Template, which contains Azure-specific stencils, threats and mitigations. For generic models, select the SDL TM Knowledge Base from the drop-down menu. Want to create your own template or submit a new one for all users? Check out our Template Repository GitHub Page to learn more
Open a Model	Opens previously saved threat models. The Recently Opened Models feature is great if you need to open your most recent files. When you hover over the selection, you'll find 2 ways to open models: <ul style="list-style-type: none"> • Open From this Computer – classic way of opening a file using local storage • Open from OneDrive – teams can use folders in OneDrive to save and share all their threat models in a single location to help increase productivity and collaboration

COMPONENT	DETAILS
Getting Started Guide	Opens the Microsoft Threat Modeling Tool main page

Template section

COMPONENT	DETAILS
Create New Template	Opens a blank template for you to build on. Unless you have extensive knowledge in building templates from scratch, we recommend you to build from existing ones
Open Template	Opens existing templates for you to make changes to

The Threat Modeling Tool team is constantly working to improve tool functionality and experience. A few minor changes might take place over the course of the year, but all major changes require rewrites in the guide. Refer to it often to ensure you get the latest announcements.

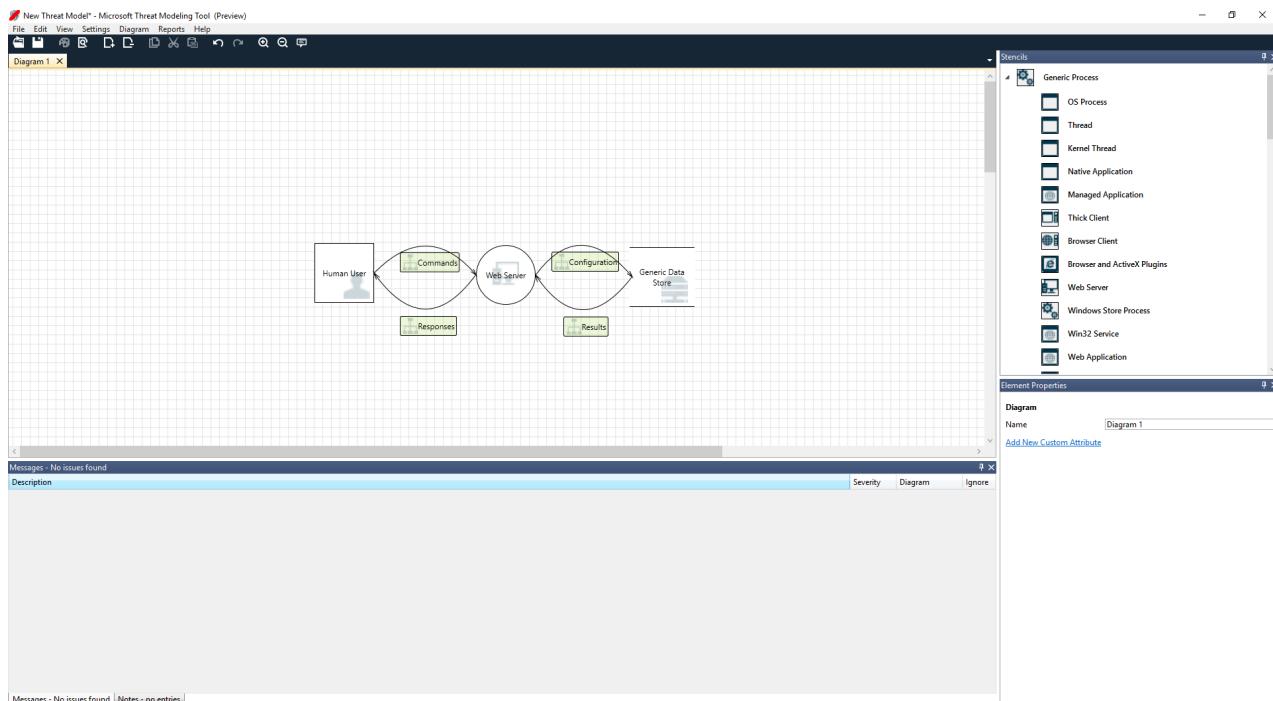
Building a model

In this section, we follow:

- Cristina (a developer)
- Ricardo (a program manager) and
- Ashish (a tester)

They are going through the process of developing their first threat model.

Ricardo: Hi Cristina, I worked on the threat model diagram and wanted to make sure we got the details right. Can you help me look it over? Cristina: Absolutely. Let's take a look. Ricardo opens the tool and shares his screen with Cristina.



Cristina: Ok, looks straightforward, but can you walk me through it? Ricardo: Sure! Here is the breakdown:

- Our human user is drawn as an outside entity—a square

- They're sending commands to our Web server—the circle
- The Web server is consulting a database (two parallel lines)

What Ricardo just showed Cristina is a DFD, short for **Data Flow Diagram**. The Threat Modeling Tool allows users to specify trust boundaries, indicated by the red dotted lines, to show where different entities are in control. For example, IT administrators require an Active Directory system for authentication purposes, so the Active Directory is outside of their control.

Cristina: Looks right to me. What about the threats? Ricardo: Let me show you.

Analyzing threats

Once he clicks on the analysis view from the icon menu selection (file with magnifying glass), he is taken to a list of generated threats the Threat Modeling Tool found based on the default template, which uses the SDL approach called **STRIDE (Spoofing, Tampering, Info Disclosure, Denial of Service and Elevation of Privilege)**. The idea is that software comes under a predictable set of threats, which can be found using these 6 categories.

This approach is like securing your house by ensuring each door and window has a locking mechanism in place before adding an alarm system or chasing after the thief.

The screenshot shows the Microsoft Threat Modeling Tool interface. At the top is a menu bar with File, Edit, View, Settings, Diagram, Reports, Help. Below the menu is a toolbar with icons for New, Open, Save, Print, Undo, Redo, Cut, Copy, Paste, Find, and Help. The main area is divided into two sections: a Data Flow Diagram on the left and a Threat List on the right.

Data Flow Diagram: This section shows a flow between a "Human User" (represented by a person icon) and a "Web Server" (represented by a server icon). The "Human User" sends "Commands" to the "Web Server". The "Web Server" sends "Responses" back to the "Human User". The "Web Server" also interacts with a "Generic Data Store" (represented by a database icon) via "Configuration" and "Results" flows.

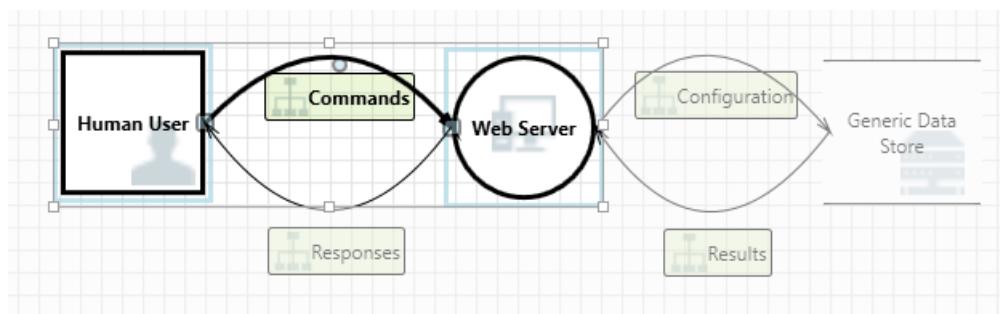
Threat List: This section is titled "Threat List" and contains a table with 9 rows of threat data. The columns are: ID, Diagram, Changed By, Last Modified, State, Title, Category, Description, Justification, Interaction, and Priority.

ID	Diagram	Changed By	Last Modified	State	Title	Category	Description	Justification	Interaction	Priority
0	Diagram 1	Generated	Not Started	Spoofing the...	Spoofing	Human User...	Human User...	Commands	High	
1	Diagram 1	Generated	Not Started	Cross Site Scr...	Tampering	The web serv...	The web serv...	Commands	High	
2	Diagram 1	Generated	Not Started	Elevation Usr...	Elevation Of...	Web Server...	Web Server...	Commands	High	
3	Diagram 1	Generated	Not Started	Spoofing of D...	Spoofing	Generic Data...	Generic Data...	Configuration	High	
4	Diagram 1	Generated	Not Started	Potential Exc...	Denial Of Ser...	Does Web Se...	Does Web Se...	Configuration	High	
5	Diagram 1	Generated	Not Started	Spoofing of S...	Spoofing	Generic Data...	Generic Data...	Results	High	
6	Diagram 1	Generated	Not Started	Cross Site Scr...	Tampering	The web serv...	The web serv...	Results	High	
7	Diagram 1	Generated	Not Started	Persistent Cr...	Tampering	The web serv...	The web serv...	Results	High	
8	Diagram 1	Generated	Not Started	Weak Access...	Information...	Improper dat...	Improper dat...	Results	High	

Below the Threat List is a message: "9 Threats Displayed, 9 Total". The Threat Properties window is open at the bottom, showing the message "No threats are selected".

Ricardo begins by selecting the first item on the list. Here's what happens:

First, the interaction between the two stencils is enhanced



Second, additional information about the threat appears in the Threat Properties window

Threat Properties	
ID: 0	Diagram: Diagram 1
Status: Not Started	
Title:	Spoofing the Human User External Entity
Category:	Spoofing
Description:	Human User may be spoofed by an attacker and this may lead to unauthorized access to Web Server. Consider using a standard authentication mechanism to identify the external entity.
Justification:	
Interaction:	Commands
Priority:	High

The generated threat helps him understand potential design flaws. The STRIDE categorization gives him an idea on potential attack vectors, while the additional description tells him exactly what's wrong, along with potential ways to mitigate it. He can use editable fields to write notes in the justification details or change priority ratings depending on his organization's bug bar.

Azure templates have additional details to help users understand not only what's wrong, but also how to fix it by adding descriptions, examples and hyperlinks to Azure-specific documentation.

The description made him realize the importance of adding an authentication mechanism to prevent users from being spoofed, revealing the first threat to be worked on. A few minutes into the discussion with Cristina, they understood the importance of implementing access control and roles. Ricardo filled in some quick notes to make sure these were implemented.

As Ricardo went into the threats under Information Disclosure, he realized the access control plan required some read-only accounts for audit and report generation. He wondered whether this should be a new threat, but the mitigations were the same, so he noted the threat accordingly. He also thought about information disclosure a bit more and realized that the backup tapes were going to need encryption, a job for the operations team.

Threats not applicable to the design due to existing mitigations or security guarantees can be changed to "Not Applicable" from the Status drop-down. There are three other choices: Not Started – default selection, Needs Investigation – used to follow up on items and Mitigated – once it's fully worked on.

Reports & sharing

Once Ricardo goes through the list with Cristina and adds important notes, mitigations/justifications, priority and status changes, he selects Reports -> Create Full Report -> Save Report, which prints out a nice report for him to go through with colleagues to ensure the proper security work is implemented.

Threat Modeling Report

Created on 7/31/2017 12:35:42 PM

Threat Model Name:

Owner:

Reviewer:

Contributors:

Description:

Assumptions:

External Dependencies:

Threat Model Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Diagram: Diagram 1

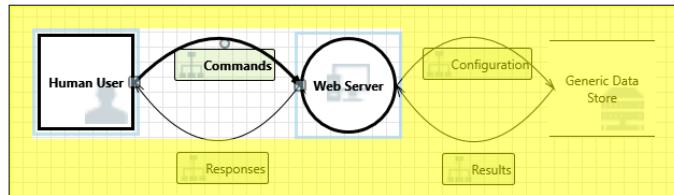
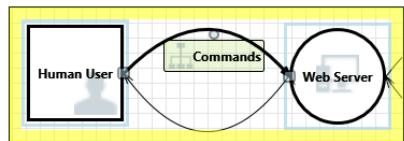


Diagram 1 Diagram Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Interaction: Commands



1. Spoofing the Human User External Entity [State: Not Started] [Priority: High]

Category: Spoofing
Description: Human User may be spoofed by an attacker and this may lead to unauthorized access to Web Server. Consider using a standard authentication mechanism to identify the external entity.
Justification: <no mitigation provided>
Possible Mitigation(s):
SDL Phase: Design

2. Cross Site Scripting [State: Not Started] [Priority: High]

Category: Tampering
Description: The web server 'Web Server' could be a subject to a cross-site scripting attack because it does not sanitize untrusted input.
Justification: <no mitigation provided>
Possible Mitigation(s):
SDL Phase: Design

If Ricardo wants to share the file instead, he can easily do so by saving in his organization's OneDrive account. Once he does that, he can copy the document link and share it with his colleagues.

Threat modeling meetings

When Ricardo sent his threat model to his colleague using OneDrive, Ashish, the tester, was underwhelmed. Seemed like Ricardo and Cristina missed quite a few important corner cases, which could be easily compromised. His skepticism is a complement to threat models.

In this scenario, after Ashish took over the threat model, he called for two threat modeling meetings: one meeting to synchronize on the process and walk through the diagrams and then a second meeting for threat review and sign-off.

In the first meeting, Ashish spent 10 minutes walking everyone through the SDL threat modeling process. He then pulled up the threat model diagram and started explaining it in detail. Within five minutes, an important missing component had been identified.

A few minutes later, Ashish and Ricardo got into an extended discussion of how the Web server was built. It was not the ideal way for a meeting to proceed, but everyone eventually agreed that discovering the discrepancy early was going to save them time in the future.

In the second meeting, the team walked through the threats, discussed some ways to address them, and signed off on the threat model. They checked the document into source control and continued with development.

Thinking about assets

Some readers who have threat modeled may notice that we haven't talked about assets at all. We've discovered that many software engineers understand their software better than they understand the concept of assets and what assets an attacker may be interested in.

If you're going to threat model a house, you might start by thinking about your family, irreplaceable photos or valuable artwork. Perhaps you might start by thinking about who might break in and the current security system. Or you might start by considering the physical features, like the pool or the front porch. These are analogous to thinking about assets, attackers, or software design. Any of these three approaches work.

The approach to threat modeling we've presented here is substantially simpler than what Microsoft has done in the past. We found that the software design approach works well for many teams. We hope that include yours.

Next Steps

Send your questions, comments and concerns to tmtextsupport@microsoft.com. [Download](#) the Threat Modeling Tool to get started.

Threat Modeling Tool feature overview

9/7/2017 • 5 min to read • [Edit Online](#)

The Threat Modeling Tool can help you with your threat modeling needs. For a basic introduction to the tool, see [Get started with the Threat Modeling Tool](#).

NOTE

The Threat Modeling Tool is updated frequently, so check this guide often to see our latest features and improvements.

To open a blank page, select **Create A Model**.

The screenshot shows the Microsoft Threat Modeling Tool (Preview) interface. At the top, there's a header bar with the title "MICROSOFT THREAT MODELING TOOL (PREVIEW)". Below the header, there's a "Threat Model:" section with a "Create A Model" button. To the right of this is a "Feedback, Suggestions and Issues" button. The main area contains three large cards:

- Create A Model**: Model your system by drawing diagram(s). Make sure you capture important details.
- Open A Model**: Open an existing model and analyze threats against your system; do not worry, the tool will help you identify them.
- Getting Started Guide**: A step-by-step guide to help you get up and running now.

Below these cards, there are sections for "Template For New Models" (with a dropdown menu showing "Azure Threat Model Template(1.0.0.20)" and a "Browse..." button) and "Recently Opened Models" (listing files like "Basic Web App NEW.tm7", "New Threat Model.tm7", etc.). On the right side, there's a "Threat Modeling Workflow" section with a list of steps: 1. Select your template. 2. Create your data flow diagram model. 3. Analyze the model for potential threats. 4. Determine mitigations.

Template:

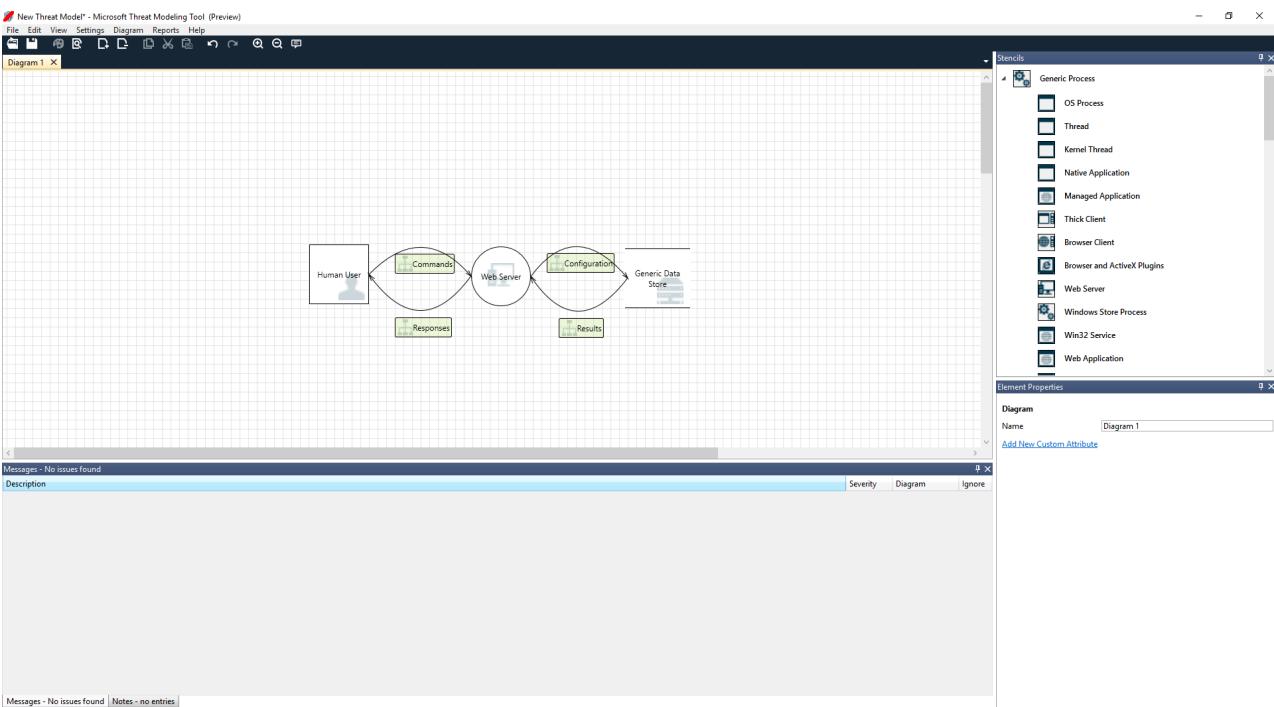
Below the main cards, there are two more cards under the "Template:" heading:

- Create New Template**: Define stencils, threat types and custom threat properties for your threat model from scratch.
- Open Template**: Open an existing Template and make modifications to better suit your specific threat analysis.

Template Workflow: Use templates to define threats that applications should look for.

1. Define stencils
2. Define categories
3. Define threat properties
4. Define threat
5. Share your template

To see the features currently available in the tool, use the threat model created by our team in the [Get started example](#).



Navigation

Before we discuss the built-in features, let's review the main components found in the tool.

Menu items

The experience is similar to other Microsoft products. Let's review the top-level menu items.



LABEL	DETAILS
File	<ul style="list-style-type: none"> Open, save, and close files Sign in and sign out of OneDrive accounts. Share links (view and edit). View file information. Apply a new template to existing models.
Edit	Undo and redo actions, as well as copy, paste, and delete.
View	<ul style="list-style-type: none"> Switch between Analysis and Design views. Open closed windows (for example, stencils, element properties, and messages). Reset layout to default settings.
Diagram	Add and delete diagrams, and move through tabs of diagrams.
Reports	Create HTML reports to share with others.
Help	Find guides to help you use the tool.

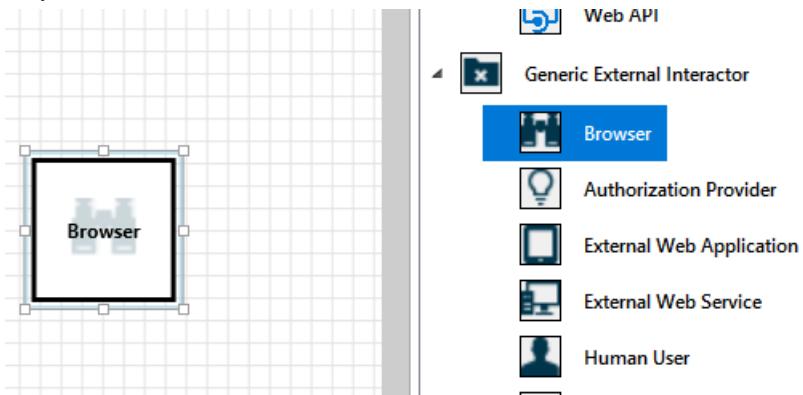
The symbols are shortcuts for the top-level menus:

SYMBOL	DETAILS
Open	Opens a new file.
Save	Saves the current file.
Design	Opens the Design view, where you can create models.
Analyze	Shows generated threats and their properties.
Add diagram	Adds a new diagram (similar to new tabs in Excel).
Delete diagram	Deletes the current diagram.
Copy/Cut/Paste	Copies, cuts, and pastes elements.
Undo/Redo	Undoes and redoes actions.
Zoom in/Zoom out	Zooms in and out of the diagram for a better view.
Feedback	Opens the MSDN Forum.

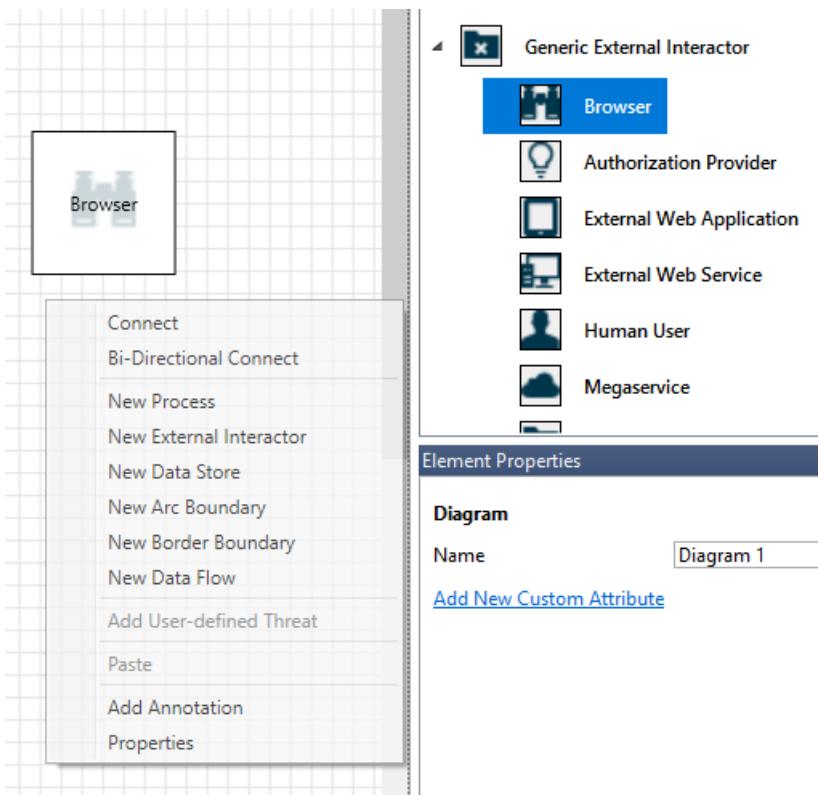
Canvas

The canvas is the space where you drag and drop elements. Drag and drop is the quickest and most efficient way to build models. You can also right-click and select items from the menu to add generic versions of elements, as shown:

Drop the stencil on the canvas



Select the stencil



Stencils

Based on the template you select, you can find all the stencils available to use. If you can't find the right elements, use another template. Or you can modify a template to fit your needs. Generally, you can find a combination of categories like these:

STENCIL NAME	DETAILS
Process	Applications, browser plug-ins, threads, virtual machines
External interactor	Authentication providers, browsers, users, web applications
Data store	Cache, storage, configuration files, databases, registry
Data flow	Binary, ALPC, HTTP, HTTPS/TLS/SSL, IOCTL, IPSec, named pipe, RPC/DCOM, SMB, UDP
Trust line/Border boundary	Corporate networks, internet, machine, sandbox, user/kernel mode

Notes/messages

COMPONENT	DETAILS
Messages	Internal tool logic that alerts users whenever there's an error, such as no data flows between elements.
Notes	Manual notes are added to the file by engineering teams throughout the design and review process.

Element properties

Element properties vary by the elements you select. Apart from trust boundaries, all other elements contain three general selections:

ELEMENT PROPERTY	DETAILS
Name	Useful for naming your processes, stores, interactors, and flows so that they're easily recognized.
Out of scope	If selected, the element is taken out of the threat-generation matrix (not recommended).
Reason for out of scope	Justification field to let users know why out of scope was selected.

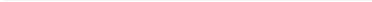
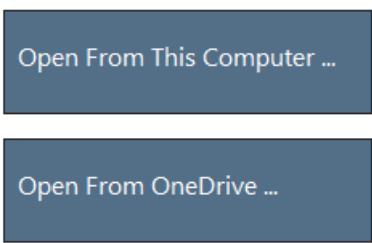
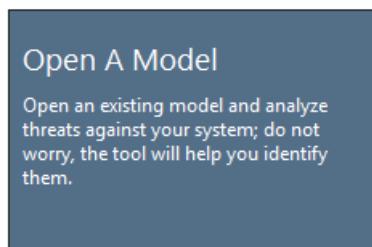
Properties are changed under each element category. Select each element to inspect the available options. Or you can open the template to learn more. Let's review the features.

Welcome screen

When you open the app, you see the **Welcome** screen.

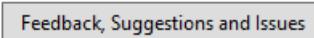
Open a model

Hover over **Open A Model** to reveal two options: **Open From This Computer** and **Open From OneDrive**. The first option opens the **File Open** screen. The second option takes you through the sign-in process for OneDrive. After successful authentication, you can select folders and files.



Feedback, suggestions, and issues

When you select **Feedback, Suggestions and Issues**, you go to the MSDN Forum for SDL Tools. You can read what other people are saying about the tool, including workarounds and new ideas.



Design view

When you open or create a new model, the **Design** view opens.

Add elements

You can add elements on the grid in two ways:

- **Drag and drop:** Drag the desired element to the grid. Then use the element properties to provide additional information.
- **Right-click:** Right-click anywhere on the grid, and select items from the drop-down menu. A generic

representation of the element you select appears on the screen.

Connect elements

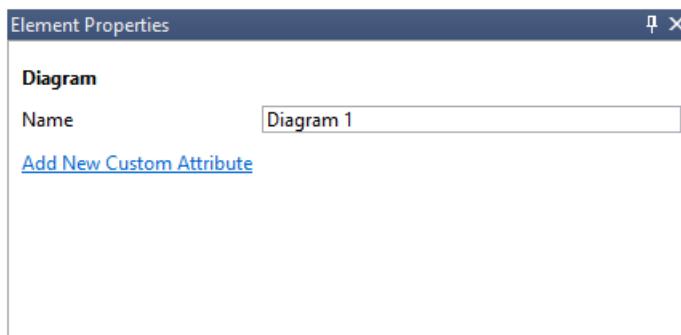
You can connect elements in two ways:

- **Drag and drop:** Drag the desired dataflow to the grid, and connect both ends to the appropriate elements.
- **Click + Shift:** Click the first element (sending data), press and hold the Shift key, and then select the second element (receiving data). Right-click, and select **Connect**. If you use a bi-directional data flow, the order is not as important.

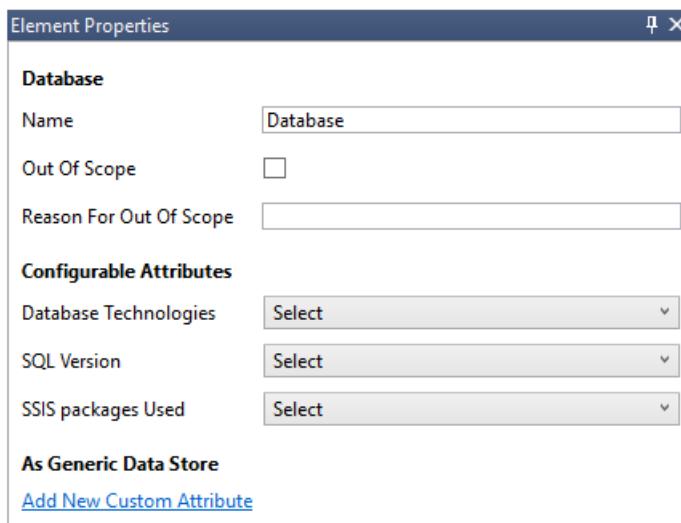
Properties

To see the properties that can be modified on the stencils, select the stencil and the information populates accordingly. The following example shows before and after a **Database** stencil is dragged onto the diagram:

Before

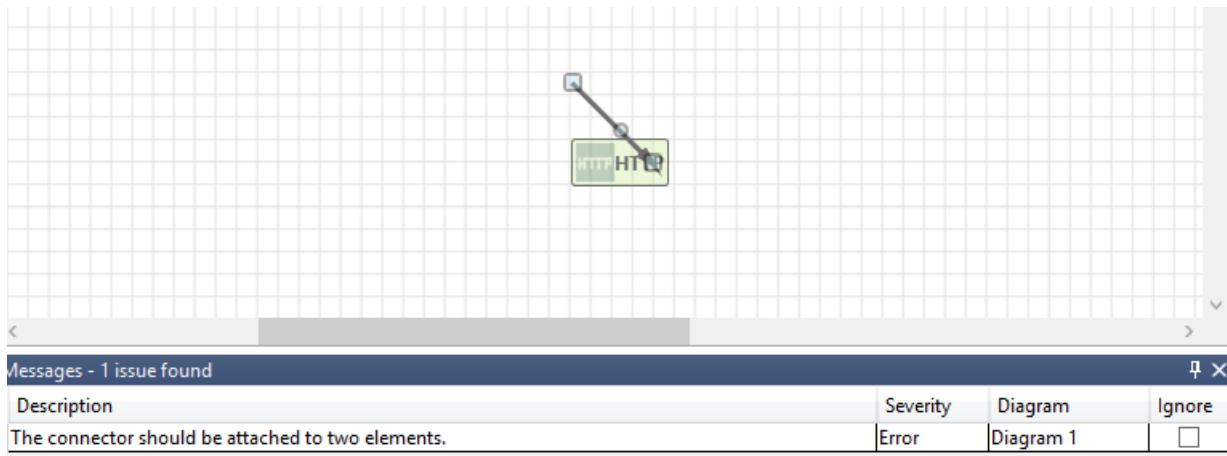


After



Messages

If you create a threat model and forget to connect data flows to elements, you get a notification. You can ignore the message, or you can follow the instructions to fix the issue.



Notes

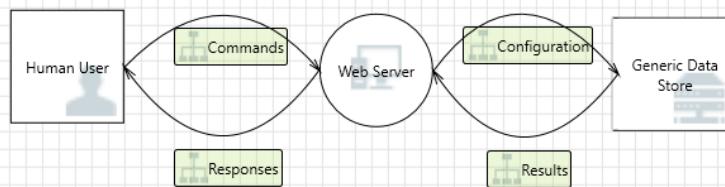
To add notes to your diagram, switch from the **Messages** tab to the **Notes** tab.

Analysis view

After you build your diagram, select the **Analysis** symbol (the magnifying glass) on the shortcuts toolbar to switch to the **Analysis** view.



Diagram 1 X



Threat List

ID	Diagram	Changed By	Last Modified	State	Title	Category	Description	Justification	Interaction	Priority
1	Diagram 1		Generated	Not Started	Cross Site Scr...	Tampering	The web serv...		Commands	High
2	Diagram 1		Generated	Not Started	Elevation Usi...	Elevation Of...	Web Server...		Commands	High
3	Diagram 1		Generated	Not Started	Spoofing of D...	Spoofing	Generic Data...		Configuration	High
4	Diagram 1		Generated	Not Started	Potential Exc...	Denial Of Ser...	Does Web Se...		Configuration	High
5	Diagram 1		Generated	Not Started	Spoofing of S...	Spoofing	Generic Data...		Results	High
6	Diagram 1		Generated	Not Started	Cross Site Scr...	Tampering	The web serv...		Results	High
7	Diagram 1		Generated	Not Started	Persistent Cr...	Tampering	The web serv...		Results	High
8	Diagram 1		Generated	Not Started	Weak Access...	Information...	Improper dat...		Results	High
9	Diagram 1		Generated	Not Started	Spoofing the...	Spoofing	Human User...		Commands	High

9 Threats Displayed, 9 Total

Threat Properties

No threats are selected

Generated threat selection

When you select a threat, you can use three distinct functions:

FEATURE

Read indicator

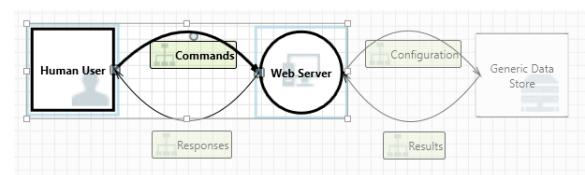
INFORMATION

The threat is marked as read, which helps you keep track of the items you reviewed.

ID	Diagram	Changed By	Last Modified	State	Title	Category	Description	Justification	Interaction	Priority
1	Diagram 1		Generated	Not Started	Cross Site Scr...	Tampering	The web serv...		Commands	High
2	Diagram 1		Generated	Not Started	Elevation Usi...	Elevation Of...	Web Server...		Commands	High
3	Diagram 1		Generated	Not Started	Spoofing of D...	Spoofing	Generic Data...		Configuration	High

Interaction focus

Interaction in the diagram that belongs to a threat is highlighted.



FEATURE	INFORMATION
Threat properties	Additional information about the threat appears in the Threat Properties window. 

Priority change

You can change the priority level of each generated threat. Different colors make it easy to identify high-, medium-, and low-priority threats.

ID	Diagram	Changed By	Last Modified	Status	Title	Category	Description	Justification	Interaction	Priority
1	Diagram 1	REDMOND\ro...	8/16/2017 3:13...	Not Started	Cross Site Scri...	Tampering	The web server...		Commands	Medium
2	Diagram 1	REDMOND\ro...	8/16/2017 3:13...	Not Started	Elevation Usin...	Elevation Of Pr...	Web Server ma...		Commands	Low
3	Diagram 1	REDMOND\ro...	8/16/2017 3:13...	Not Started	Spoofing of De...	Spoofing	Generic Data S...		Configuration	Medium
4	Diagram 1	REDMOND\ro...	8/16/2017 3:13...	Not Started	Potential Exces...	Denial Of Servi...	Does Web Serv...		Configuration	Medium
5	Diagram 1		Generated	Not Started	Spoofing of So...	Spoofing	Generic Data S...		Results	High
6	Diagram 1		Generated	Not Started	Cross Site Scri...	Tampering	The web server...		Results	High
7	Diagram 1	REDMOND\ro...	8/16/2017 3:13...	Not Started	Persistent Cros...	Tampering	The web server...		Results	Low
8	Diagram 1		Generated	Not Started	Weak Access...	Information...	Improper dat...		Results	High
9	Diagram 1		Generated	Not Started	Spoofing the...	Spoofing	Human User...		Commands	High

Threat properties editable fields

As seen in the preceding image, you can change the information generated by the tool. You can also add information to certain fields, such as justification. These fields are generated by the template. If you need more information for each threat, you can make modifications.

Threat Properties	
ID: 2	Diagram: Diagram 1
Status: Not Started	
Last Modified: Generated	
Title: Elevation Using Impersonation	
Category: Elevation Of Privilege	
Description: Web Server may be able to impersonate the context of Human User in order to gain additional privilege.	
Justification:	
Interaction: Commands	
Priority: High	

Reports

After you finish changing priorities and updating the status of each generated threat, you can save the file and/or print out a report. Go to **Report > Create Full Report**. Name the report, and you should see something similar to the following image:

Threat Modeling Report

Created on 7/31/2017 12:35:42 PM

Threat Model Name:

Owner:

Reviewer:

Contributors:

Description:

Assumptions:

External Dependencies:

Threat Model Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Diagram: Diagram 1

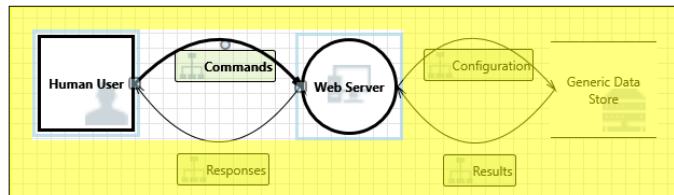
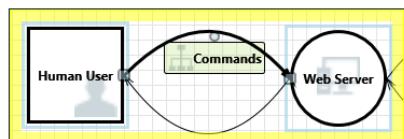


Diagram 1 Diagram Summary:

Not Started	9
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	9
Total Migrated	0

Interaction: Commands



1. Spoofing the Human User External Entity [State: Not Started] [Priority: High]

Category:	Spoofing
Description:	Human User may be spoofed by an attacker and this may lead to unauthorized access to Web Server. Consider using a standard authentication mechanism to identify the external entity.
Justification:	<no mitigation provided>
Possible Mitigation(s):	
SDL Phase:	Design

2. Cross Site Scripting [State: Not Started] [Priority: High]

Category:	Tampering
Description:	The web server 'Web Server' could be a subject to a cross-site scripting attack because it does not sanitize untrusted input.
Justification:	<no mitigation provided>
Possible Mitigation(s):	
SDL Phase:	Design

Next steps

- To contribute a template for the community, go to our [GitHub](#) page.
- To get started with the tool, go to the [Download](#) page.

Microsoft Threat Modeling Tool threats

8/25/2017 • 2 min to read • [Edit Online](#)

The Threat Modeling Tool is a core element of the Microsoft Security Development Lifecycle (SDL). It allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. As a result, it greatly reduces the total cost of development. Also, we designed the tool with non-security experts in mind, making threat modeling easier for all developers by providing clear guidance on creating and analyzing threat models.

Visit the [Threat Modeling Tool](#) to get started today!

The Threat Modeling Tool helps you answer certain questions, such as the ones below:

- How can an attacker change the authentication data?
- What is the impact if an attacker can read the user profile data?
- What happens if access is denied to the user profile database?

STRIDE model

To better help you formulate these kinds of pointed questions, Microsoft uses the STRIDE model, which categorizes different types of threats and simplifies the overall security conversations.

CATEGORY	DESCRIPTION
Spoofing	Involves illegally accessing and then using another user's authentication information, such as username and password
Tampering	Involves the malicious modification of data. Examples include unauthorized changes made to persistent data, such as that held in a database, and the alteration of data as it flows between two computers over an open network, such as the Internet
Repudiation	Associated with users who deny performing an action without other parties having any way to prove otherwise—for example, a user performs an illegal operation in a system that lacks the ability to trace the prohibited operations. Non-Repudiation refers to the ability of a system to counter repudiation threats. For example, a user who purchases an item might have to sign for the item upon receipt. The vendor can then use the signed receipt as evidence that the user did receive the package
Information Disclosure	Involves the exposure of information to individuals who are not supposed to have access to it—for example, the ability of users to read a file that they were not granted access to, or the ability of an intruder to read data in transit between two computers
Denial of Service	Denial of service (DoS) attacks deny service to valid users—for example, by making a Web server temporarily unavailable or unusable. You must protect against certain types of DoS threats simply to improve system availability and reliability

CATEGORY	DESCRIPTION
Elevation of Privilege	An unprivileged user gains privileged access and thereby has sufficient access to compromise or destroy the entire system. Elevation of privilege threats include those situations in which an attacker has effectively penetrated all system defenses and become part of the trusted system itself, a dangerous situation indeed

Next steps

Proceed to [Threat Modeling Tool Mitigations](#) to learn the different ways you can mitigate these threats with Azure.

Microsoft Threat Modeling Tool mitigations

8/25/2017 • 2 min to read • [Edit Online](#)

The Threat Modeling Tool is a core element of the Microsoft Security Development Lifecycle (SDL). It allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. As a result, it greatly reduces the total cost of development. Also, we designed the tool with non-security experts in mind, making threat modeling easier for all developers by providing clear guidance on creating and analyzing threat models.

Visit the [Threat Modeling Tool](#) to get started today!

Mitigation categories

The Threat Modeling Tool mitigations are categorized according to the Web Application Security Frame, which consists of the following:

CATEGORY	DESCRIPTION
Auditing and Logging	Who did what and when? Auditing and logging refer to how your application records security-related events
Authentication	Who are you? Authentication is the process where an entity proves the identity of another entity, typically through credentials, such as a user name and password
Authorization	What can you do? Authorization is how your application provides access controls for resources and operations
Communication Security	Who are you talking to? Communication Security ensures all communication done is as secure as possible
Configuration Management	Who does your application run as? Which databases does it connect to? How is your application administered? How are these settings secured? Configuration management refers to how your application handles these operational issues
Cryptography	How are you keeping secrets (confidentiality)? How are you tamper-proofing your data or libraries (integrity)? How are you providing seeds for random values that must be cryptographically strong? Cryptography refers to how your application enforces confidentiality and integrity
Exception Management	When a method call in your application fails, what does your application do? How much do you reveal? Do you return friendly error information to end users? Do you pass valuable exception information back to the caller? Does your application fail gracefully?

CATEGORY	DESCRIPTION
Input Validation	How do you know that the input your application receives is valid and safe? Input validation refers to how your application filters, scrubs, or rejects input before additional processing. Consider constraining input through entry points and encoding output through exit points. Do you trust data from sources such as databases and file shares?
Sensitive Data	How does your application handle sensitive data? Sensitive data refers to how your application handles any data that must be protected either in memory, over the network, or in persistent stores
Session Management	How does your application handle and protect user sessions? A session refers to a series of related interactions between a user and your Web application

This helps you identify:

- Where are the most common mistakes made
- Where are the most actionable improvements

As a result, you use these categories to focus and prioritize your security work, so that if you know the most prevalent security issues occur in the input validation, authentication and authorization categories, you can start there. For more information visit [this patent link](#)

Next steps

Visit [Threat Modeling Tool Threats](#) to learn more about the threat categories the tool uses to generate possible design threats.

Security Frame: Auditing and Logging | Mitigations

8/22/2017 • 8 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
Dynamics CRM	<ul style="list-style-type: none">• Identify sensitive entities in your solution and implement change auditing
Web Application	<ul style="list-style-type: none">• Ensure that auditing and logging is enforced on the application• Ensure that log rotation and separation are in place• Ensure that the application does not log sensitive user data• Ensure that Audit and Log Files have Restricted Access• Ensure that User Management Events are Logged• Ensure that the system has inbuilt defenses against misuse• Enable diagnostics logging for web apps in Azure App Service
Database	<ul style="list-style-type: none">• Ensure that login auditing is enabled on SQL Server• Enable Threat detection on Azure SQL
Azure Storage	<ul style="list-style-type: none">• Use Azure Storage Analytics to audit access of Azure Storage
WCF	<ul style="list-style-type: none">• Implement sufficient Logging• Implement sufficient Audit Failure Handling
Web API	<ul style="list-style-type: none">• Ensure that auditing and logging is enforced on Web API
IoT Field Gateway	<ul style="list-style-type: none">• Ensure that appropriate auditing and logging is enforced on Field Gateway
IoT Cloud Gateway	<ul style="list-style-type: none">• Ensure that appropriate auditing and logging is enforced on Cloud Gateway

Identify sensitive entities in your solution and implement change auditing

TITLE	DETAILS
Component	Dynamics CRM

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Identify entities in your solution containing sensitive data and implement change auditing on those entities and fields

Ensure that auditing and logging is enforced on the application

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Enable auditing and logging on all components. Audit logs should capture user context. Identify all important events and log those events. Implement centralized logging

Ensure that log rotation and separation are in place

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	<p>Log rotation is an automated process used in system administration in which dated log files are archived. Servers which run large applications often log every request: in the face of bulky logs, log rotation is a way to limit the total size of the logs while still allowing analysis of recent events.</p> <p>Log separation basically means that you have to store your log files on a different partition as where your OS/application is running on in order to avert a Denial of service attack or the downgrading of your application its performance</p>

Ensure that the application does not log sensitive user data

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	<p>Check that you do not log any sensitive data that a user submits to your site. Check for intentional logging as well as side effects caused by design issues. Examples of sensitive data include:</p> <ul style="list-style-type: none"> • User Credentials • Social Security number or other identifying information • Credit card numbers or other financial information • Health information • Private keys or other data that could be used to decrypt encrypted information • System or application information that can be used to more effectively attack the application

Ensure that Audit and Log Files have Restricted Access

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	<p>Check to ensure access rights to log files are appropriately set. Application accounts should have write-only access and operators and support personnel should have read-only access as needed.</p> <p>Administrators accounts are the only accounts which should have full access. Check Windows ACL on log files to ensure they are properly restricted:</p> <ul style="list-style-type: none"> • Application accounts should have write-only access • Operators and support personnel should have read-only access as needed • Administrators are the only accounts that should have full access

Ensure that User Management Events are Logged

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that the application monitors user management events such as successful and failed user logins, password resets, password changes, account lockout, user registration. Doing this helps to detect and react to potentially suspicious behavior. It also enables to gather operations data; for example, to track who is accessing the application

Ensure that the system has inbuilt defenses against misuse

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	<p>Controls should be in place which throw security exception in case of application misuse. E.g., If input validation is in place and an attacker attempts to inject malicious code that does not match the regex, a security exception can be thrown which can be an indicative of system misuse</p> <p>For example, it is recommended to have security exceptions logged and actions taken for the following issues:</p> <ul style="list-style-type: none"> • Input validation • CSRF violations • Brute force (upper limit for number of requests per user per resource) • File upload violations

Enable diagnostics logging for web apps in Azure App Service

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	EnvironmentType - Azure
References	N/A
Steps	<p>Azure provides built-in diagnostics to assist with debugging an App Service web app. It also applies to API apps and mobile apps. App Service web apps provide diagnostic functionality for logging information from both the web server and the web application.</p> <p>These are logically separated into web server diagnostics and application diagnostics</p>

Ensure that login auditing is enabled on SQL Server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	Configure Login Auditing
Steps	Database Server login auditing must be enabled to detect/confirm password guessing attacks. It is important to capture failed login attempts. Capturing both successful and failed login attempts provides additional benefit during forensic investigations

Enable Threat detection on Azure SQL

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure
Attributes	SQL Version - V12
References	Get Started with SQL Database Threat Detection
Steps	<p>Threat Detection detects anomalous database activities indicating potential security threats to the database. It provides a new layer of security, which enables customers to detect and respond to potential threats as they occur by providing security alerts on anomalous activities.</p> <p>Users can explore the suspicious events using Azure SQL Database Auditing to determine if they result from an attempt to access, breach or exploit data in the database.</p> <p>Threat Detection makes it simple to address potential threats to the database without the need to be a security expert or manage advanced security monitoring systems</p>

Use Azure Storage Analytics to audit access of Azure Storage

TITLE	DETAILS
Component	Azure Storage
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Using Storage Analytics to monitor authorization type

TITLE	DETAILS
Steps	<p>For each storage account, one can enable Azure Storage Analytics to perform logging and store metrics data. The storage analytics logs provide important information such as authentication method used by someone when they access storage.</p> <p>This can be really helpful if you are tightly guarding access to storage. For example, in Blob Storage you can set all of the containers to private and implement the use of an SAS service throughout your applications. Then you can check the logs regularly to see if your blobs are accessed using the storage account keys, which may indicate a breach of security, or if the blobs are public but they shouldn't be.</p>

Implement sufficient Logging

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework
Attributes	N/A
References	MSDN , Fortify Kingdom
Steps	<p>The lack of a proper audit trail after a security incident can hamper forensic efforts. Windows Communication Foundation (WCF) offers the ability to log successful and/or failed authentication attempts.</p>
	<p>Logging failed authentication attempts can warn administrators of potential brute-force attacks. Similarly, logging successful authentication events can provide a useful audit trail when a legitimate account is compromised. Enable WCF's service security audit feature</p>

Example

The following is an example configuration with auditing enabled

```
<system.serviceModel>
  <behaviors>
    <serviceBehaviors>
      <behavior name=""NewBehavior"">
        <serviceSecurityAudit auditLogLocation=""Default"""
          suppressAuditFailure=""false"""
          serviceAuthorizationAuditLevel=""SuccessAndFailure"""
          messageAuthenticationAuditLevel=""SuccessAndFailure"" />
        ...
      </behavior>
    </serviceBehaviors>
  </behaviors>
</system.serviceModel>
```

Implement sufficient Audit Failure Handling

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework
Attributes	N/A
References	MSDN , Fortify Kingdom
Steps	Developed solution is configured not to generate an exception when it fails to write to an audit log. If WCF is configured not to throw an exception when it is unable to write to an audit log, the program will not be notified of the failure and auditing of critical security events may not occur.

Example

The `<behavior/>` element of the WCF configuration file below instructs WCF to not notify the application when WCF fails to write to an audit log.

```
<behaviors>
    <serviceBehaviors>
        <behavior name="NewBehavior">
            <serviceSecurityAudit auditLogLocation="Application"
                suppressAuditFailure="true"
                serviceAuthorizationAuditLevel="Success"
                messageAuthenticationAuditLevel="Success" />
        </behavior>
    </serviceBehaviors>
</behaviors>
```

Configure WCF to notify the program whenever it is unable to write to an audit log. The program should have an alternative notification scheme in place to alert the organization that audit trails are not being maintained.

Ensure that auditing and logging is enforced on Web API

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Steps	Enable auditing and logging on Web APIs. Audit logs should capture user context. Identify all important events and log those events. Implement centralized logging

Ensure that appropriate auditing and logging is enforced on Field Gateway

TITLE	DETAILS
Component	IoT Field Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	<p>When multiple devices connect to a Field Gateway, ensure that connection attempts and authentication status (success or failure) for individual devices are logged and maintained on the Field Gateway.</p> <p>Also, in cases where Field Gateway is maintaining the IoT Hub credentials for individual devices, ensure that auditing is performed when these credentials are retrieved. Develop a process to periodically upload the logs to Azure IoT Hub/storage for long term retention.</p>

Ensure that appropriate auditing and logging is enforced on Cloud Gateway

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Introduction to IoT Hub operations monitoring

TITLE	DETAILS
Steps	<p>Design for collecting and storing audit data gathered through IoT Hub Operations Monitoring. Enable the following monitoring categories:</p> <ul style="list-style-type: none">• Device identity operations• Device-to-cloud communications• Cloud-to-device communications• Connections• File uploads

Security Frame: Authentication | Mitigations

8/22/2017 • 25 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
Web Application	<ul style="list-style-type: none">• Consider using a standard authentication mechanism to authenticate to Web Application• Applications must handle failed authentication scenarios securely• Enable step up or adaptive authentication• Ensure that administrative interfaces are appropriately locked down• Implement forgot password functionalities securely• Ensure that password and account policy are implemented• Implement controls to prevent username enumeration
Database	<ul style="list-style-type: none">• When possible, use Windows Authentication for connecting to SQL Server• When possible use Azure Active Directory Authentication for Connecting to SQL Database• When SQL authentication mode is used, ensure that account and password policy are enforced on SQL server• Do not use SQL Authentication in contained databases
Azure Event Hub	<ul style="list-style-type: none">• Use per device authentication credentials using SAs tokens
Azure Trust Boundary	<ul style="list-style-type: none">• Enable Azure Multi-Factor Authentication for Azure Administrators
Service Fabric Trust Boundary	<ul style="list-style-type: none">• Restrict anonymous access to Service Fabric Cluster• Ensure that Service Fabric client-to-node certificate is different from node-to-node certificate• Use AAD to authenticate clients to service fabric clusters• Ensure that service fabric certificates are obtained from an approved Certificate Authority (CA)
Identity Server	<ul style="list-style-type: none">• Use standard authentication scenarios supported by Identity Server• Override the default Identity Server token cache with a scalable alternative
Machine Trust Boundary	<ul style="list-style-type: none">• Ensure that deployed application's binaries are digitally signed

PRODUCT/SERVICE	ARTICLE
WCF	<ul style="list-style-type: none"> Enable authentication when connecting to MSMQ queues in WCF WCF-Do not set Message clientCredentialType to none WCF-Do not set Transport clientCredentialType to none
Web API	<ul style="list-style-type: none"> Ensure that standard authentication techniques are used to secure Web APIs
Azure AD	<ul style="list-style-type: none"> Use standard authentication scenarios supported by Azure Active Directory Override the default ADAL token cache with a scalable alternative Ensure that TokenReplayCache is used to prevent the replay of ADAL authentication tokens Use ADAL libraries to manage token requests from OAuth2 clients to AAD (or on-premises AD)
IoT Field Gateway	<ul style="list-style-type: none"> Authenticate devices connecting to the Field Gateway
IoT Cloud Gateway	<ul style="list-style-type: none"> Ensure that devices connecting to Cloud gateway are authenticated Use per-device authentication credentials
Azure Storage	<ul style="list-style-type: none"> Ensure that only the required containers and blobs are given anonymous read access Grant limited access to objects in Azure storage using SAS or SAP

Consider using a standard authentication mechanism to authenticate to Web Application

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Details	<p>Authentication is the process where an entity proves its identity, typically through credentials, such as a user name and password. There are multiple authentication protocols available which may be considered. Some of them are listed below:</p> <ul style="list-style-type: none"> • Client certificates • Windows based • Forms based • Federation - ADFS • Federation - Azure AD • Federation - Identity Server <p>Consider using a standard authentication mechanism to identify the source process</p>

Applications must handle failed authentication scenarios securely

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Details	<p>Applications that explicitly authenticate users must handle failed authentication scenarios securely. The authentication mechanism must:</p> <ul style="list-style-type: none"> • Deny access to privileged resources when authentication fails • Display a generic error message after failed authentication and access denied occurs <p>Test for:</p> <ul style="list-style-type: none"> • Protection of privileged resources after failed logins • A generic error message is displayed on failed authentication and access denied event(s) • Accounts are disabled after an excessive number of failed attempts

Enable step up or adaptive authentication

TITLE	DETAILS
Component	Web Application
SDL Phase	Build

TITLE	DETAILS
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Details	<p>Verify the application has additional authorization (such as step up or adaptive authentication, via multi-factor authentication such as sending OTP in SMS, email etc. or prompting for re-authentication) so the user is challenged before being granted access to sensitive information. This rule also applies for making critical changes to an account or action</p> <p>This also means that the adaptation of authentication has to be implemented in such a manner that the application correctly enforces context-sensitive authorization so as to not allow unauthorized manipulation by means of for example, parameter tampering</p>

Ensure that administrative interfaces are appropriately locked down

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Details	The first solution is to grant access only from a certain source IP range to the administrative interface. If that solution would not be possible than it is always recommended to enforce a step-up or adaptive authentication for logging in into the administrative interface

Implement forgot password functionalities securely

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Details	<p>The first thing is to verify that forgot password and other recovery paths send a link including a time-limited activation token rather than the password itself. Additional authentication based on soft-tokens (e.g. SMS token, native mobile applications, etc.) can be required as well before the link is sent over. Second, you should not lock out the users account whilst the process of getting a new password is in progress.</p> <p>This could lead to a Denial of service attack whenever an attacker decides to intentionally lock out the users with an automated attack. Third, whenever the new password request was set in progress, the message you display should be generalized in order to prevent username enumeration. Fourth, always disallow the use of old passwords and implement a strong password policy.</p>

Ensure that password and account policy are implemented

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
Details	<p>Password and account policy in compliance with organizational policy and best practices should be implemented.</p> <p>To defend against brute-force and dictionary based guessing: Strong password policy must be implemented to ensure that users create complex password (e.g., 12 characters minimum length, alphanumeric and special characters).</p> <p>Account lockout policies may be implemented in the following manner:</p> <ul style="list-style-type: none"> • Soft lock-out: This can be a good option for protecting your users against brute force attacks. For example, whenever the user enters a wrong password three times the application could lock down the account for a minute in order to slow down the process of brute forcing his password making it less profitable for the attacker to proceed. If you were to implement hard lock-out countermeasures for this example you would achieve a "Dos" by permanently locking out accounts. Alternatively, application may generate an OTP (One Time Password) and send it out-of-band (through email, sms etc.) to the user. Another approach may be to implement CAPTCHA after a threshold number of failed attempts is reached. • Hard lock-out: This type of lockout should be applied whenever you detect a user attacking your application and counter him by means of permanently locking out his account until a response team had time to do their forensics. After this process you can decide to give the user back his account or take further legal actions against him. This type of approach prevents the attacker from further penetrating your application and infrastructure. <p>To defend against attacks on default and predictable accounts, verify that all keys and passwords are replaceable, and are generated or replaced after installation time.</p> <p>If the application has to auto-generate passwords, ensure that the generated passwords are random and have high entropy.</p>

Implement controls to prevent username enumeration

TITLE	DETAILS
Component	Web Application
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	N/A
Steps	All error messages should be generalized in order to prevent username enumeration. Also sometimes you cannot avoid information leaking in functionalities such as a registration page. Here you need to use rate-limiting methods like CAPTCHA to prevent an automated attack by an attacker.

When possible, use Windows Authentication for connecting to SQL Server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	OnPrem
Attributes	SQL Version - All
References	SQL Server - Choose an Authentication Mode
Steps	Windows Authentication uses Kerberos security protocol, provides password policy enforcement with regard to complexity validation for strong passwords, provides support for account lockout, and supports password expiration.

When possible use Azure Active Directory Authentication for Connecting to SQL Database

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	SQL Azure
Attributes	SQL Version - V12
References	Connecting to SQL Database By Using Azure Active Directory Authentication
Steps	Minimum version: Azure SQL Database V12 required to allow Azure SQL Database to use AAD Authentication against the Microsoft Directory

When SQL authentication mode is used, ensure that account and

password policy are enforced on SQL server

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	SQL Server password policy
Steps	When using SQL Server Authentication, logins are created in SQL Server that are not based on Windows user accounts. Both the user name and the password are created by using SQL Server and stored in SQL Server. SQL Server can use Windows password policy mechanisms. It can apply the same complexity and expiration policies used in Windows to passwords used inside SQL Server.

Do not use SQL Authentication in contained databases

TITLE	DETAILS
Component	Database
SDL Phase	Build
Applicable Technologies	OnPrem, SQL Azure
Attributes	SQL Version - MSSQL2012, SQL Version - V12
References	Security Best Practices with Contained Databases
Steps	The absence of an enforced password policy may increase the likelihood of a weak credential being established in a contained database. Leverage Windows Authentication.

Use per device authentication credentials using SaS tokens

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A

TITLE	DETAILS
References	Event Hubs authentication and security model overview
Steps	<p>The Event Hubs security model is based on a combination of Shared Access Signature (SAS) tokens and event publishers. The publisher name represents the DeviceID that receives the token. This would help associate the tokens generated with the respective devices.</p> <p>All messages are tagged with originator on service side allowing detection of in-payload origin spoofing attempts. When authenticating devices, generate a per device SaS token scoped to a unique publisher.</p>

Enable Azure Multi-Factor Authentication for Azure Administrators

TITLE	DETAILS
Component	Azure Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	What is Azure Multi-Factor Authentication?
Steps	<p>Multi-factor authentication (MFA) is a method of authentication that requires more than one verification method and adds a critical second layer of security to user sign-ins and transactions. It works by requiring any two or more of the following verification methods:</p> <ul style="list-style-type: none"> • Something you know (typically a password) • Something you have (a trusted device that is not easily duplicated, like a phone) • Something you are (biometrics)

Restrict anonymous access to Service Fabric Cluster

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	Service Fabric cluster security scenarios

TITLE	DETAILS
Steps	<p>Clusters should always be secured to prevent unauthorized users from connecting to your cluster, especially when it has production workloads running on it.</p> <p>While creating a service fabric cluster, ensure that the security mode is set to "secure" and configure the required X.509 server certificate. Creating an "insecure" cluster will allow any anonymous user to connect to it if it exposes management endpoints to the public Internet.</p>

Ensure that Service Fabric client-to-node certificate is different from node-to-node certificate

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure, Environment - Stand alone
References	Service Fabric Client-to-node certificate security , Connect to a secure cluster using client certificate
Steps	<p>Client-to-node certificate security is configured while creating the cluster either through the Azure portal, Resource Manager templates or a standalone JSON template by specifying an admin client certificate and/or a user client certificate.</p> <p>The admin client and user client certificates you specify should be different than the primary and secondary certificates you specify for Node-to-node security.</p>

Use AAD to authenticate clients to service fabric clusters

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	Cluster security scenarios - Security Recommendations

TITLE	DETAILS
Steps	Clusters running on Azure can also secure access to the management endpoints using Azure Active Directory (AAD), apart from client certificates. For Azure clusters, it is recommended that you use AAD security to authenticate clients and certificates for node-to-node security.

Ensure that service fabric certificates are obtained from an approved Certificate Authority (CA)

TITLE	DETAILS
Component	Service Fabric Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	Environment - Azure
References	X.509 certificates and Service Fabric
Steps	<p>Service Fabric uses X.509 server certificates for authenticating nodes and clients.</p> <p>Some important things to consider while using certificates in service fabrics:</p> <ul style="list-style-type: none"> • Certificates used in clusters running production workloads should be created by using a correctly configured Windows Server certificate service or obtained from an approved Certificate Authority (CA). The CA can be an approved external CA or a properly managed internal Public Key Infrastructure (PKI) • Never use any temporary or test certificates in production that are created with tools such as MakeCert.exe • You can use a self-signed certificate, but should only do so for test clusters and not in production

Use standard authentication scenarios supported by Identity Server

TITLE	DETAILS
Component	Identity Server
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	IdentityServer3 - The Big Picture

TITLE	DETAILS
Steps	<p>Below are the typical interactions supported by Identity Server:</p> <ul style="list-style-type: none"> • Browsers communicate with web applications • Web applications communicate with web APIs (sometimes on their own, sometimes on behalf of a user) • Browser-based applications communicate with web APIs • Native applications communicate with web APIs • Server-based applications communicate with web APIs • Web APIs communicate with web APIs (sometimes on their own, sometimes on behalf of a user)

Override the default Identity Server token cache with a scalable alternative

TITLE	DETAILS
Component	Identity Server
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	Identity Server Deployment - Caching
Steps	<p>IdentityServer has a simple built-in in-memory cache. While this is good for small scale native apps, it does not scale for mid tier and backend applications for the following reasons:</p> <ul style="list-style-type: none"> • These applications are accessed by many users at once. Saving all access tokens in the same store creates isolation issues and presents challenges when operating at scale: many users, each with as many tokens as the resources the app accesses on their behalf, can mean huge numbers and very expensive lookup operations • These applications are typically deployed on distributed topologies, where multiple nodes must have access to the same cache • Cached tokens must survive process recycles and deactivations • For all the above reasons, while implementing web apps, it is recommended to override the default Identity Server's token cache with a scalable alternative such as Azure Redis cache

Ensure that deployed application's binaries are digitally signed

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that deployed application's binaries are digitally signed so that the integrity of the binaries can be verified

Enable authentication when connecting to MSMQ queues in WCF

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, .NET Framework 3
Attributes	N/A
References	MSDN
Steps	Program fails to enable authentication when connecting to MSMQ queues, an attacker can anonymously submit messages to the queue for processing. If authentication is not used to connect to an MSMQ queue used to deliver a message to another program, an attacker could submit an anonymous message that is malicious.

Example

The `<netMsmqBinding/>` element of the WCF configuration file below instructs WCF to disable authentication when connecting to an MSMQ queue for message delivery.

```

<bindings>
    <netMsmqBinding>
        <binding>
            <security>
                <transport msmqAuthenticationMode=""None"" />
            </security>
        </binding>
    </netMsmqBinding>
</bindings>

```

Configure MSMQ to require Windows Domain or Certificate authentication at all times for any incoming or outgoing messages.

Example

The `<netMsmqBinding/>` element of the WCF configuration file below instructs WCF to enable certificate authentication when connecting to an MSMQ queue. The client is authenticated using X.509 certificates. The client certificate must be present in the certificate store of the server.

```

<bindings>
  <netMsmqBinding>
    <binding>
      <security>
        <transport msmqAuthenticationMode=""Certificate"" />
      </security>
    </binding>
  </netMsmqBinding>
</bindings>

```

WCF-Do not set Message clientCredentialType to none

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	.NET Framework 3
Attributes	Client Credential Type - None
References	MSDN , Fortify
Steps	The absence of authentication means everyone is able to access this service. A service that does not authenticate its clients allows access to all users. Configure the application to authenticate against client credentials. This can be done by setting the message clientCredentialType to Windows or Certificate.

Example

```
<message clientCredentialType=""Certificate""/>
```

WCF-Do not set Transport clientCredentialType to none

TITLE	DETAILS
Component	WCF
SDL Phase	Build
Applicable Technologies	Generic, .NET Framework 3
Attributes	Client Credential Type - None
References	MSDN , Fortify

TITLE	DETAILS
Steps	The absence of authentication means everyone is able to access this service. A service that does not authenticate its clients allows all users to access its functionality. Configure the application to authenticate against client credentials. This can be done by setting the transport clientCredentialType to Windows or Certificate.

Example

```
<transport clientCredentialType=""Certificate""/>
```

Ensure that standard authentication techniques are used to secure Web APIs

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Authentication and Authorization in ASP.NET Web API , External Authentication Services with ASP.NET Web API (C#)
Steps	<p>Authentication is the process where an entity proves its identity, typically through credentials, such as a user name and password. There are multiple authentication protocols available which may be considered. Some of them are listed below:</p> <ul style="list-style-type: none"> • Client certificates • Windows based • Forms based • Federation - ADFS • Federation - Azure AD • Federation - Identity Server <p>Links in the references section provide low-level details on how each of the authentication schemes can be implemented to secure a Web API.</p>

Use standard authentication scenarios supported by Azure Active Directory

TITLE	DETAILS
Component	Azure AD

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Authentication Scenarios for Azure AD , Azure Active Directory Code Samples , Azure Active Directory developer's guide
Steps	<p>Azure Active Directory (Azure AD) simplifies authentication for developers by providing identity as a service, with support for industry-standard protocols such as OAuth 2.0 and OpenID Connect. Below are the five primary application scenarios supported by Azure AD:</p> <ul style="list-style-type: none"> • Web Browser to Web Application: A user needs to sign in to a web application that is secured by Azure AD • Single Page Application (SPA): A user needs to sign in to a single page application that is secured by Azure AD • Native Application to Web API: A native application that runs on a phone, tablet, or PC needs to authenticate a user to get resources from a web API that is secured by Azure AD • Web Application to Web API: A web application needs to get resources from a web API secured by Azure AD • Daemon or Server Application to Web API: A daemon application or a server application with no web user interface needs to get resources from a web API secured by Azure AD <p>Please refer to the links in the references section for low-level implementation details</p>

Override the default ADAL token cache with a scalable alternative

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Modern Authentication with Azure Active Directory for Web Applications, Using Redis as ADAL token cache

TITLE	DETAILS
Steps	<p>The default cache that ADAL (Active Directory Authentication Library) uses is an in-memory cache that relies on a static store, available process-wide. While this works for native applications, it does not scale for mid tier and backend applications for the following reasons:</p> <ul style="list-style-type: none"> • These applications are accessed by many users at once. Saving all access tokens in the same store creates isolation issues and presents challenges when operating at scale: many users, each with as many tokens as the resources the app accesses on their behalf, can mean huge numbers and very expensive lookup operations • These applications are typically deployed on distributed topologies, where multiple nodes must have access to the same cache • Cached tokens must survive process recycles and deactivations <p>For all the above reasons, while implementing web apps, it is recommended to override the default ADAL token cache with a scalable alternative such as Azure Redis cache.</p>

Ensure that TokenReplayCache is used to prevent the replay of ADAL authentication tokens

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	Modern Authentication with Azure Active Directory for Web Applications

TITLE	DETAILS
Steps	<p>The TokenReplayCache property allows developers to define a token replay cache, a store that can be used for saving tokens for the purpose of verifying that no token can be used more than once.</p> <p>This is a measure against a common attack, the aptly called token replay attack: an attacker intercepting the token sent at sign-in might try to send it to the app again ("replay" it) for establishing a new session. E.g., In OIDC code-grant flow, after successful user authentication, a request to "/signin-oidc" endpoint of the relying party is made with "id_token", "code" and "state" parameters.</p> <p>The relying party validates this request and establishes a new session. If an adversary captures this request and replays it, he/she can establish a successful session and spoof the user. The presence of the nonce in OpenID Connect can limit but not fully eliminate the circumstances in which the attack can be successfully enacted. To protect their applications, developers can provide an implementation of ITokenReplayCache and assign an instance to TokenReplayCache.</p>

Example

```
// ITokenReplayCache defined in ADAL
public interface ITokenReplayCache
{
    bool TryAdd(string securityToken, DateTime expiresOn);
    bool TryFind(string securityToken);
}
```

Example

Here is a sample implementation of the ITokenReplayCache interface. (Please customize and implement your project-specific caching framework)

```
public class TokenReplayCache : ITokenReplayCache
{
    private readonly ICacheProvider cache; // Your project-specific cache provider
    public TokenReplayCache(ICacheProvider cache)
    {
        this.cache = cache;
    }
    public bool TryAdd(string securityToken, DateTime expiresOn)
    {
        if (this.cache.Get<string>(securityToken) == null)
        {
            this.cache.Set(securityToken, securityToken);
            return true;
        }
        return false;
    }
    public bool TryFind(string securityToken)
    {
        return this.cache.Get<string>(securityToken) != null;
    }
}
```

The implemented cache has to be referenced in OIDC options via the "TokenValidationParameters" property as

follows.

```
OpenIdConnectOptions openIdConnectOptions = new OpenIdConnectOptions
{
    AutomaticAuthenticate = true,
    ... // other configuration properties follow..
    TokenValidationParameters = new TokenValidationParameters
    {
        TokenReplayCache = new TokenReplayCache(/*Inject your cache provider*/);
    }
}
```

Please note that to test the effectiveness of this configuration, login into your local OIDC-protected application and capture the request to `"/signin-oidc"` endpoint in fiddler. When the protection is not in place, replaying this request in fiddler will set a new session cookie. When the request is replayed after the `TokenReplayCache` protection is added, the application will throw an exception as follows:

```
SecurityTokenReplayDetectedException: IDX10228: The securityToken has previously been validated, securityToken: 'eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Ik1uQ19WVmNBVGZNNXBPWWlKSE1iYTlnb0VLWSIsImtpZCI6Ik1uQ1.....'
```

Use ADAL libraries to manage token requests from OAuth2 clients to AAD (or on-premises AD)

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	ADAL
Steps	<p>The Azure AD authentication Library (ADAL) enables client application developers to easily authenticate users to cloud or on-premises Active Directory (AD), and then obtain access tokens for securing API calls.</p> <p>ADAL has many features that make authentication easier for developers, such as asynchronous support, a configurable token cache that stores access tokens and refresh tokens, automatic token refresh when an access token expires and a refresh token is available, and more.</p> <p>By handling most of the complexity, ADAL can help a developer focus on business logic in their application and easily secure resources without being an expert on security. Separate libraries are available for .NET, JavaScript (client and Node.js), iOS, Android and Java.</p>

Authenticate devices connecting to the Field Gateway

TITLE	DETAILS
Component	IoT Field Gateway

TITLE	DETAILS
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that each device is authenticated by the Field Gateway before accepting data from them and before facilitating upstream communications with the Cloud Gateway. Also, ensure that devices connect with a per device credential so that individual devices can be uniquely identified.

Ensure that devices connecting to Cloud gateway are authenticated

TITLE	DETAILS
Component	IoT Cloud Gateway
SDL Phase	Build
Applicable Technologies	Generic, C#, Node.JS,
Attributes	N/A, Gateway choice - Azure IoT Hub
References	N/A, Azure IoT hub with .NET, Getting Started with IoT hub and Node JS, Securing IoT with SAS and certificates, Git repository
Steps	<ul style="list-style-type: none"> • Generic: Authenticate the device using Transport Layer Security (TLS) or IPSec. Infrastructure should support using pre-shared key (PSK) on those devices that cannot handle full asymmetric cryptography. Leverage Azure AD, Oauth. • C#: When creating a DeviceClient instance, by default, the Create method creates a DeviceClient instance that uses the AMQP protocol to communicate with IoT Hub. To use the HTTPS protocol, use the override of the Create method that enables you to specify the protocol. If you use the HTTPS protocol, you should also add the <code>Microsoft.AspNet.WebApi.Client</code> NuGet package to your project to include the <code>System.Net.Http.Formatting</code> namespace.

Example

```

static DeviceClient deviceClient;

static string deviceKey = "{device key}";
static string iotHubUri = "{iot hub hostname}";

var messageString = "{message in string format}";
var message = new Message(Encoding.ASCII.GetBytes(messageString));

deviceClient = DeviceClient.Create(iotHubUri, new DeviceAuthenticationWithRegistrySymmetricKey("myFirstDevice",
deviceKey));

await deviceClient.SendEventAsync(message);

```

Example

Node.JS: Authentication

Symmetric key

- Create a IoT hub on azure
- Create an entry in the device identity registry

```
javascript var device = new iothub.Device(null); device.deviceId = <DeviceId> registry.create(device,
function(err, deviceInfo, res) {})
```

- Create a simulated device

```
javascript var clientFromConnectionString = require('azure-iot-device-amqp').clientFromConnectionString; var
Message = require('azure-iot-device').Message; var connectionString = 'HostName=<HostName>DeviceId=
<DeviceId>SharedAccessKey=<SharedAccessKey>'; var client = clientFromConnectionString(connectionString);
```

SAS Token

- Gets internally generated when using symmetric key but we can generate and use it explicitly as well
- Define a protocol :

```
var Http = require('azure-iot-device-http').Http;
```
- Create a sas token :

```
``javascript resourceUri =
encodeURIComponent(resourceUri.toLowerCase()).toLowerCase(); var deviceIdName = ""; var expires =
(Date.now() / 1000) + expiresInMins * 60; var toSign = resourceUri + '\n' + expires; // using crypto var
decodedPassword = new Buffer(signingKey, 'base64').toString('binary'); const hmac =
crypto.createHmac('sha256', decodedPassword); hmac.update(toSign); var base64signature =
hmac.digest('base64'); var base64UriEncoded = encodeURIComponent(base64signature); // construct
authorization string var token = "SharedAccessSignature sr=" + resourceUri +
"%2fdevices%2f" + deviceIdName + "&sig="
```
- ```
base64UriEncoded + "&se=" + expires; if (policyName) token += "&skn=" + policyName; return token; ``
```
- Connect using sas token: 

```
javascript Client.fromSharedAccessSignature(sas, Http); ##### Certificates
```
- Generate a self signed X509 certificate using any tool such as OpenSSL to generate a .cert and .key files to store the certificate and the key respectively
- Provision a device that accepts secured connection using certificates.

```
javascript var connectionString = '<connectionString>'; var registry =
iothub.Registry.fromConnectionString(connectionString); var deviceJSON = {deviceId:<deviceId>,
authentication: { x509Thumbprint: { primaryThumbprint: "<PrimaryThumbprint>", secondaryThumbprint: "
<SecondaryThumbprint>" } }} var device = deviceJSON; registry.create(device, function (err) {});
```

- Connect a device using a certificate

```
javascript var Protocol = require('azure-iot-device-http').Http; var Client = require('azure-iot-
device').Client; var connectionString = 'HostName=<HostName>DeviceId=<DeviceId>x509=true'; var client =
Client.fromConnectionString(connectionString, Protocol); var options = { key: fs.readFileSync('./key.pem',
'utf8'), cert: fs.readFileSync('./server.crt', 'utf8') }; // Calling setOptions with the x509 certificate and
key (and optionally, passphrase) will configure the client //transport to use x509 when connecting to IoT Hub
client.setOptions(options); //call fn to execute after the connection is set up client.open(fn);
```

## Use per-device authentication credentials

| TITLE                          | DETAILS                                                                                                                                                                                                                                           |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | IoT Cloud Gateway                                                                                                                                                                                                                                 |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                             |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                           |
| <b>Attributes</b>              | Gateway choice - Azure IoT Hub                                                                                                                                                                                                                    |
| <b>References</b>              | <a href="#">Azure IoT Hub Security Tokens</a>                                                                                                                                                                                                     |
| <b>Steps</b>                   | Use per device authentication credentials using SAs tokens based on Device key or Client Certificate, instead of IoT Hub-level shared access policies. This prevents the reuse of authentication tokens of one device or field gateway by another |

Ensure that only the required containers and blobs are given anonymous read access

| TITLE                          | DETAILS                                                                                                                             |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Storage                                                                                                                       |
| <b>SDL Phase</b>               | Build                                                                                                                               |
| <b>Applicable Technologies</b> | Generic                                                                                                                             |
| <b>Attributes</b>              | StorageType - Blob                                                                                                                  |
| <b>References</b>              | <a href="#">Manage anonymous read access to containers and blobs, Shared Access Signatures, Part 1: Understanding the SAS model</a> |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | <p>By default, a container and any blobs within it may be accessed only by the owner of the storage account. To give anonymous users read permissions to a container and its blobs, one can set the container permissions to allow public access. Anonymous users can read blobs within a publicly accessible container without authenticating the request.</p> <p>Containers provide the following options for managing container access:</p> <ul style="list-style-type: none"> <li>• Full public read access: Container and blob data can be read via anonymous request. Clients can enumerate blobs within the container via anonymous request, but cannot enumerate containers within the storage account.</li> <li>• Public read access for blobs only: Blob data within this container can be read via anonymous request, but container data is not available. Clients cannot enumerate blobs within the container via anonymous request</li> <li>• No public read access: Container and blob data can be read by the account owner only</li> </ul> <p>Anonymous access is best for scenarios where certain blobs should always be available for anonymous read access. For finer-grained control, one can create a shared access signature, which enables to delegate restricted access using different permissions and over a specified time interval. Ensure that containers and blobs, which may potentially contain sensitive data, are not given anonymous access accidentally</p> |

## Grant limited access to objects in Azure storage using SAS or SAP

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Storage                                                                                                                                                                                                                                                                                          |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                                                  |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                                                                |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                                                                                    |
| <b>References</b>              | <a href="#">Shared Access Signatures, Part 1: Understanding the SAS model</a> , <a href="#">Shared Access Signatures, Part 2: Create and use a SAS with Blob storage</a> , <a href="#">How to delegate access to objects in your account using Shared Access Signatures and Stored Access Policies</a> |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | <p>Using a shared access signature (SAS) is a powerful way to grant limited access to objects in a storage account to other clients, without having to expose account access key. The SAS is a URI that encompasses in its query parameters all of the information necessary for authenticated access to a storage resource. To access storage resources with the SAS, the client only needs to pass in the SAS to the appropriate constructor or method.</p> <p>You can use a SAS when you want to provide access to resources in your storage account to a client that can't be trusted with the account key. Your storage account keys include both a primary and secondary key, both of which grant administrative access to your account and all of the resources in it. Exposing either of your account keys opens your account to the possibility of malicious or negligent use. Shared access signatures provide a safe alternative that allows other clients to read, write, and delete data in your storage account according to the permissions you've granted, and without need for the account key.</p> <p>If you have a logical set of parameters that are similar each time, using a Stored Access Policy (SAP) is a better idea. Because using a SAS derived from a Stored Access Policy gives you the ability to revoke that SAS immediately, it is the recommended best practice to always use Stored Access Policies when possible.</p> |

# Security Frame: Authorization | Mitigations

8/22/2017 • 17 min to read • [Edit Online](#)

| PRODUCT/SERVICE                      | ARTICLE                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|--------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Machine Trust Boundary</b>        | <ul style="list-style-type: none"><li>• Ensure that proper ACLs are configured to restrict unauthorized access to data on the device</li><li>• Ensure that sensitive user-specific application content is stored in user-profile directory</li><li>• Ensure that the deployed applications are run with least privileges</li></ul>                                                                                                                                                                                      |
| <b>Web Application</b>               | <ul style="list-style-type: none"><li>• Enforce sequential step order when processing business logic flows</li><li>• Implement rate limiting mechanism to prevent enumeration</li><li>• Ensure that proper authorization is in place and principle of least privileges is followed</li><li>• Business logic and resource access authorization decisions should not be based on incoming request parameters</li><li>• Ensure that content and resources are not enumerable or accessible via forceful browsing</li></ul> |
| <b>Database</b>                      | <ul style="list-style-type: none"><li>• Ensure that least-privileged accounts are used to connect to Database server</li><li>• Implement Row Level Security RLS to prevent tenants from accessing each other's data</li><li>• Sysadmin role should only have valid necessary users</li></ul>                                                                                                                                                                                                                            |
| <b>IoT Cloud Gateway</b>             | <ul style="list-style-type: none"><li>• Connect to Cloud Gateway using least-privileged tokens</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Azure Event Hub</b>               | <ul style="list-style-type: none"><li>• Use a send-only permissions SAS Key for generating device tokens</li><li>• Do not use access tokens that provide direct access to the Event Hub</li><li>• Connect to Event Hub using SAS keys that have the minimum permissions required</li></ul>                                                                                                                                                                                                                              |
| <b>Azure Document DB</b>             | <ul style="list-style-type: none"><li>• Use resource tokens to connect to DocumentDB whenever possible</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Azure Trust Boundary</b>          | <ul style="list-style-type: none"><li>• Enable fine-grained access management to Azure Subscription using RBAC</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Service Fabric Trust Boundary</b> | <ul style="list-style-type: none"><li>• Restrict client's access to cluster operations using RBAC</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                             |

| PRODUCT/SERVICE     | ARTICLE                                                                                                                                                                                                                              |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Dynamics CRM        | <ul style="list-style-type: none"> <li>Perform security modeling and use Field Level Security where required</li> </ul>                                                                                                              |
| Dynamics CRM Portal | <ul style="list-style-type: none"> <li>Perform security modeling of portal accounts keeping in mind that the security model for the portal differs from the rest of CRM</li> </ul>                                                   |
| Azure Storage       | <ul style="list-style-type: none"> <li>Grant fine-grained permission on a range of entities in Azure Table Storage</li> <li>Enable Role-Based Access Control (RBAC) to Azure storage account using Azure Resource Manager</li> </ul> |
| Mobile Client       | <ul style="list-style-type: none"> <li>Implement implicit jailbreak or rooting detection</li> </ul>                                                                                                                                  |
| WCF                 | <ul style="list-style-type: none"> <li>Weak Class Reference in WCF</li> <li>WCF-Implement Authorization control</li> </ul>                                                                                                           |
| Web API             | <ul style="list-style-type: none"> <li>Implement proper authorization mechanism in ASP.NET Web API</li> </ul>                                                                                                                        |
| IoT Device          | <ul style="list-style-type: none"> <li>Perform authorization checks in the device if it supports various actions that require different permission levels</li> </ul>                                                                 |
| IoT Field Gateway   | <ul style="list-style-type: none"> <li>Perform authorization checks in the Field Gateway if it supports various actions that require different permission levels</li> </ul>                                                          |

Ensure that proper ACLs are configured to restrict unauthorized access to data on the device

| TITLE                   | DETAILS                                                                                      |
|-------------------------|----------------------------------------------------------------------------------------------|
| Component               | Machine Trust Boundary                                                                       |
| SDL Phase               | Deployment                                                                                   |
| Applicable Technologies | Generic                                                                                      |
| Attributes              | N/A                                                                                          |
| References              | N/A                                                                                          |
| Steps                   | Ensure that proper ACLs are configured to restrict unauthorized access to data on the device |

## Ensure that sensitive user-specific application content is stored in user-profile directory

| TITLE                          | DETAILS                                                                                                                                                                         |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Machine Trust Boundary                                                                                                                                                          |
| <b>SDL Phase</b>               | Deployment                                                                                                                                                                      |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                         |
| <b>Attributes</b>              | N/A                                                                                                                                                                             |
| <b>References</b>              | N/A                                                                                                                                                                             |
| <b>Steps</b>                   | Ensure that sensitive user-specific application content is stored in user-profile directory. This is to prevent multiple users of the machine from accessing each other's data. |

## Ensure that the deployed applications are run with least privileges

| TITLE                          | DETAILS                                                            |
|--------------------------------|--------------------------------------------------------------------|
| <b>Component</b>               | Machine Trust Boundary                                             |
| <b>SDL Phase</b>               | Deployment                                                         |
| <b>Applicable Technologies</b> | Generic                                                            |
| <b>Attributes</b>              | N/A                                                                |
| <b>References</b>              | N/A                                                                |
| <b>Steps</b>                   | Ensure that the deployed application is run with least privileges. |

## Enforce sequential step order when processing business logic flows

| TITLE                          | DETAILS         |
|--------------------------------|-----------------|
| <b>Component</b>               | Web Application |
| <b>SDL Phase</b>               | Build           |
| <b>Applicable Technologies</b> | Generic         |
| <b>Attributes</b>              | N/A             |
| <b>References</b>              | N/A             |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | In order to verify that this stage was run through by a genuine user you want to enforce the application to only process business logic flows in sequential step order, with all steps being processed in realistic human time, and not process out of order, skipped steps, processed steps from another user, or too quickly submitted transactions. |

## Implement rate limiting mechanism to prevent enumeration

| TITLE                          | DETAILS                                                                                                                                                     |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Web Application                                                                                                                                             |
| <b>SDL Phase</b>               | Build                                                                                                                                                       |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                     |
| <b>Attributes</b>              | N/A                                                                                                                                                         |
| <b>References</b>              | N/A                                                                                                                                                         |
| <b>Steps</b>                   | Ensure that sensitive identifiers are random. Implement CAPTCHA control on anonymous pages. Ensure that error and exception should not reveal specific data |

## Ensure that proper authorization is in place and principle of least privileges is followed

| TITLE                          | DETAILS         |
|--------------------------------|-----------------|
| <b>Component</b>               | Web Application |
| <b>SDL Phase</b>               | Build           |
| <b>Applicable Technologies</b> | Generic         |
| <b>Attributes</b>              | N/A             |
| <b>References</b>              | N/A             |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | <p>The principle means giving a user account only those privileges which are essential to that user's work. For example, a backup user does not need to install software; hence, the backup user has rights only to run backup and backup-related applications. Any other privileges, such as installing new software, are blocked. The principle applies also to a personal computer user who usually does work in a normal user account, and opens a privileged, password protected account (that is, a superuser) only when the situation absolutely demands it.</p> <p>This principle can also be applied to your web-applications. Instead of solely depending on role-based authentication methods using sessions, we rather want to assign privileges to users by means of a Database-Based Authentication system. We still use sessions in order to identify if the user was logged in correctly, only now instead of assigning that user with a specific role we assign him with privileges to verify which actions he is privileged to perform on the system. Also a big pro of this method is, whenever a user has to be assigned fewer privileges your changes will be applied on the fly since the assigning does not depend on the session which otherwise had to expire first.</p> |

## Business logic and resource access authorization decisions should not be based on incoming request parameters

| TITLE                          | DETAILS                                                                                                                                                                                                                                                              |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Web Application                                                                                                                                                                                                                                                      |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                              |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                                                  |
| <b>References</b>              | N/A                                                                                                                                                                                                                                                                  |
| <b>Steps</b>                   | Whenever you are checking whether a user is restricted to review certain data, the access restrictions should be processed server-side. The userID should be stored inside of a session variable on login and should be used to retrieve user data from the database |

### Example

```
SELECT data
FROM personaldata
WHERE userID=:id < - session var
```

Now an possible attacker can not tamper and change the application operation since the identifier for retrieving the data is handled server-side.

## Ensure that content and resources are not enumerable or accessible via forceful browsing

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Web Application                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>References</b>              | N/A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Steps</b>                   | <p>Sensitive static and configuration files should not be kept in the web-root. For content not required to be public, either proper access controls should be applied or removal of the content itself.</p> <p>Also, forceful browsing is usually combined with Brute Force techniques to gather information by attempting to access as many URLs as possible to enumerate directories and files on a server. Attackers may check for all variations of commonly existing files. For example, a password file search would encompass files including psswd.txt, password.htm, password.dat, and other variations.</p> <p>To mitigate this, capabilities for detection of brute force attempts should be included.</p> |

## Ensure that least-privileged accounts are used to connect to Database server

| TITLE                          | DETAILS                                                                                                                                                                                                                                 |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Database                                                                                                                                                                                                                                |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                   |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                 |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                     |
| <b>References</b>              | <a href="#">SQL Database permissions hierarchy</a> , <a href="#">SQL database securities</a>                                                                                                                                            |
| <b>Steps</b>                   | Least-privileged accounts should be used to connect to the database. Application login should be restricted in the database and should only execute selected stored procedures. Application's login should have no direct table access. |

## Implement Row Level Security RLS to prevent tenants from accessing each other's data

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Database                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Applicable Technologies</b> | Sql Azure, OnPrem                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Attributes</b>              | SQL Version - V12, SQL Version - MsSQL2016                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>References</b>              | <a href="#">SQL Server Row-Level Security (RLS)</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Steps</b>                   | <p>Row-Level Security enables customers to control access to rows in a database table based on the characteristics of the user executing a query (e.g., group membership or execution context).</p> <p>Row-Level Security (RLS) simplifies the design and coding of security in your application. RLS enables you to implement restrictions on data row access. For example ensuring that workers can access only those data rows that are pertinent to their department, or restricting a customer's data access to only the data relevant to their company.</p> <p>The access restriction logic is located in the database tier rather than away from the data in another application tier. The database system applies the access restrictions every time that data access is attempted from any tier. This makes the security system more reliable and robust by reducing the surface area of the security system.</p> |

Please note that RLS as an out-of-the-box database feature is applicable only to SQL Server starting 2016 and Azure SQL database. If the out-of-the-box RLS feature is not implemented, it should be ensured that data access is restricted Using Views and Procedures

## Sysadmin role should only have valid necessary users

| TITLE                          | DETAILS                                                                                                                                                                                           |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Database                                                                                                                                                                                          |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                             |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                           |
| <b>Attributes</b>              | N/A                                                                                                                                                                                               |
| <b>References</b>              | <a href="#">SQL Database permissions hierarchy, SQL database securables</a>                                                                                                                       |
| <b>Steps</b>                   | Members of the SysAdmin fixed server role should be very limited and never contain accounts used by applications. Please review the list of users in the role and remove any unnecessary accounts |

## Connect to Cloud Gateway using least-privileged tokens

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                   |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | IoT Cloud Gateway                                                                                                                                                                                                                                                                         |
| <b>SDL Phase</b>               | Deployment                                                                                                                                                                                                                                                                                |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                                                   |
| <b>Attributes</b>              | Gateway choice - Azure IoT Hub                                                                                                                                                                                                                                                            |
| <b>References</b>              | <a href="#">IoT Hub Access Control</a>                                                                                                                                                                                                                                                    |
| <b>Steps</b>                   | Provide least privilege permissions to various components that connect to Cloud Gateway (IoT Hub). Typical example is – Device management/provisioning component uses registryread/write, Event Processor (ASA) uses Service Connect. Individual devices connect using Device credentials |

## Use a send-only permissions SAS Key for generating device tokens

| TITLE                          | DETAILS                                                                                                                                             |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Event Hub                                                                                                                                     |
| <b>SDL Phase</b>               | Build                                                                                                                                               |
| <b>Applicable Technologies</b> | Generic                                                                                                                                             |
| <b>Attributes</b>              | N/A                                                                                                                                                 |
| <b>References</b>              | <a href="#">Event Hubs authentication and security model overview</a>                                                                               |
| <b>Steps</b>                   | A SAS key is used to generate individual device tokens. Use a send-only permissions SAS key while generating the device token for a given publisher |

## Do not use access tokens that provide direct access to the Event Hub

| TITLE                          | DETAILS                                                               |
|--------------------------------|-----------------------------------------------------------------------|
| <b>Component</b>               | Azure Event Hub                                                       |
| <b>SDL Phase</b>               | Build                                                                 |
| <b>Applicable Technologies</b> | Generic                                                               |
| <b>Attributes</b>              | N/A                                                                   |
| <b>References</b>              | <a href="#">Event Hubs authentication and security model overview</a> |

| TITLE        | DETAILS                                                                                                                                                                                                                                                      |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | A token that grants direct access to the event hub should not be given to the device. Using a least privileged token for the device that gives access only to a publisher would help identify and blacklist it if found to be a rogue or compromised device. |

Connect to Event Hub using SAS keys that have the minimum permissions required

| TITLE                          | DETAILS                                                                                                                                                                                                                                   |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Event Hub                                                                                                                                                                                                                           |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                     |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                   |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                       |
| <b>References</b>              | <a href="#">Event Hubs authentication and security model overview</a>                                                                                                                                                                     |
| <b>Steps</b>                   | Provide least privilege permissions to various back-end applications that connect to the Event Hub. Generate separate SAS keys for each back-end application and only provide the required permissions - Send, Receive or Manage to them. |

Use resource tokens to connect to Cosmos DB whenever possible

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Document DB                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>References</b>              | N/A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Steps</b>                   | A resource token is associated with a DocumentDB permission resource and captures the relationship between the user of a database and the permission that user has for a specific DocumentDB application resource (e.g. collection, document). Always use a resource token to access the DocumentDB if the client cannot be trusted with handling master or read-only keys - like an end user application like a mobile or desktop client. Use Master key or read-only keys from backend applications which can store these keys securely. |

Enable fine-grained access management to Azure Subscription using

## RBAC

| TITLE                          | DETAILS                                                                                                                                                                             |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Trust Boundary                                                                                                                                                                |
| <b>SDL Phase</b>               | Build                                                                                                                                                                               |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                             |
| <b>Attributes</b>              | N/A                                                                                                                                                                                 |
| <b>References</b>              | <a href="#">Use role assignments to manage access to your Azure subscription resources</a>                                                                                          |
| <b>Steps</b>                   | Azure Role-Based Access Control (RBAC) enables fine-grained access management for Azure. Using RBAC, you can grant only the amount of access that users need to perform their jobs. |

## Restrict client's access to cluster operations using RBAC

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Service Fabric Trust Boundary                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>SDL Phase</b>               | Deployment                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Attributes</b>              | Environment - Azure                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>References</b>              | <a href="#">Role-based access control for Service Fabric clients</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>Steps</b>                   | <p>Azure Service Fabric supports two different access control types for clients that are connected to a Service Fabric cluster: administrator and user. Access control allows the cluster administrator to limit access to certain cluster operations for different groups of users, making the cluster more secure.</p> <p>Administrators have full access to management capabilities (including read/write capabilities). Users, by default, have only read access to management capabilities (for example, query capabilities), and the ability to resolve applications and services.</p> <p>You specify the two client roles (administrator and client) at the time of cluster creation by providing separate certificates for each.</p> |

## Perform security modeling and use Field Level Security where required

| TITLE                          | DETAILS                                                               |
|--------------------------------|-----------------------------------------------------------------------|
| <b>Component</b>               | Dynamics CRM                                                          |
| <b>SDL Phase</b>               | Build                                                                 |
| <b>Applicable Technologies</b> | Generic                                                               |
| <b>Attributes</b>              | N/A                                                                   |
| <b>References</b>              | N/A                                                                   |
| <b>Steps</b>                   | Perform security modeling and use Field Level Security where required |

Perform security modeling of portal accounts keeping in mind that the security model for the portal differs from the rest of CRM

| TITLE                          | DETAILS                                                                                                                          |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Dynamics CRM Portal                                                                                                              |
| <b>SDL Phase</b>               | Build                                                                                                                            |
| <b>Applicable Technologies</b> | Generic                                                                                                                          |
| <b>Attributes</b>              | N/A                                                                                                                              |
| <b>References</b>              | N/A                                                                                                                              |
| <b>Steps</b>                   | Perform security modeling of portal accounts keeping in mind that the security model for the portal differs from the rest of CRM |

Grant fine-grained permission on a range of entities in Azure Table Storage

| TITLE                          | DETAILS                                                                                   |
|--------------------------------|-------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Storage                                                                             |
| <b>SDL Phase</b>               | Build                                                                                     |
| <b>Applicable Technologies</b> | Generic                                                                                   |
| <b>Attributes</b>              | StorageType - Table                                                                       |
| <b>References</b>              | <a href="#">How to delegate access to objects in your Azure storage account using SAS</a> |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                  |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | In certain business scenarios, Azure Table Storage may be required to store sensitive data that caters to different parties. E.g., sensitive data pertaining to different countries. In such cases, SAS signatures can be constructed by specifying the partition and row key ranges, such that a user can access data specific to a particular country. |

## Enable Role-Based Access Control (RBAC) to Azure storage account using Azure Resource Manager

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | Azure Storage                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Applicable Technologies</b> | Generic                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>References</b>              | <a href="#">How to secure your storage account with Role-Based Access Control (RBAC)</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Steps</b>                   | <p>When you create a new storage account, you select a deployment model of Classic or Azure Resource Manager. The Classic model of creating resources in Azure only allows all-or-nothing access to the subscription, and in turn, the storage account.</p> <p>With the Azure Resource Manager model, you put the storage account in a resource group and control access to the management plane of that specific storage account using Azure Active Directory. For example, you can give specific users the ability to access the storage account keys, while other users can view information about the storage account, but cannot access the storage account keys.</p> |

## Implement implicit jailbreak or rooting detection

| TITLE                          | DETAILS       |
|--------------------------------|---------------|
| <b>Component</b>               | Mobile Client |
| <b>SDL Phase</b>               | Build         |
| <b>Applicable Technologies</b> | Generic       |
| <b>Attributes</b>              | N/A           |
| <b>References</b>              | N/A           |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | <p>Application should safeguard its own configuration and user data in case if phone is rooted or jail broken. Rooting/jail breaking implies unauthorized access, which normal users won't do on their own phones. Therefore application should have implicit detection logic on application startup, to detect if the phone has been rooted.</p> <p>The detection logic can be simply accessing files which normally only root user can access, for example:</p> <ul style="list-style-type: none"> <li>• /system/app/Superuser.apk</li> <li>• /sbin/su</li> <li>• /system/bin/su</li> <li>• /system/xbin/su</li> <li>• /data/local/xbin/su</li> <li>• /data/local/bin/su</li> <li>• /system/sd/xbin/su</li> <li>• /system/bin/failsafe/su</li> <li>• /data/local/su</li> </ul> <p>If the application can access any of these files, it denotes that the application is running as root user.</p> |

## Weak Class Reference in WCF

| TITLE                          | DETAILS                                                |
|--------------------------------|--------------------------------------------------------|
| <b>Component</b>               | WCF                                                    |
| <b>SDL Phase</b>               | Build                                                  |
| <b>Applicable Technologies</b> | Generic, .NET Framework 3                              |
| <b>Attributes</b>              | N/A                                                    |
| <b>References</b>              | <a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a> |

| TITLE        | DETAILS                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Steps</b> | <p>The system uses a weak class reference, which might allow an attacker to execute unauthorized code. The program references a user-defined class that is not uniquely identified. When .NET loads this weakly identified class, the CLR type loader searches for the class in the following locations in the specified order:</p> <ol style="list-style-type: none"> <li>1. If the assembly of the type is known, the loader searches the configuration file's redirect locations, GAC, the current assembly using configuration information, and the application base directory</li> <li>2. If the assembly is unknown, the loader searches the current assembly, mscorelib, and the location returned by the TypeResolve event handler</li> <li>3. This CLR search order can be modified with hooks such as the Type Forwarding mechanism and the AppDomain.TypeResolve event</li> </ol> <p>If an attacker exploits the CLR search order by creating an alternative class with the same name and placing it in an alternative location that the CLR will load first, the CLR will unintentionally execute the attacker-supplied code</p> |

## Example

The `<behaviorExtensions/>` element of the WCF configuration file below instructs WCF to add a custom behavior class to a particular WCF extension.

```

<system.serviceModel>
 <extensions>
 <behaviorExtensions>
 <add name=""myBehavior"" type=""MyBehavior"" />
 </behaviorExtensions>
 </extensions>
</system.serviceModel>

```

Using fully qualified (strong) names uniquely identifies a type and further increases security of your system. Use fully qualified assembly names when registering types in the machine.config and app.config files.

## Example

The `<behaviorExtensions/>` element of the WCF configuration file below instructs WCF to add strongly-referenced custom behavior class to a particular WCF extension.

```

<system.serviceModel>
 <extensions>
 <behaviorExtensions>
 <add name=""myBehavior"" type=""Microsoft.ServiceModel.Samples.MyBehaviorSection, MyBehavior,
Version=1.0.0.0, Culture=neutral, PublicKeyToken=null"" />
 </behaviorExtensions>
 </extensions>
</system.serviceModel>

```

## WCF-Implement Authorization control

| TITLE                          | DETAILS                                                                                                                                                                                                                                                                                                                                                  |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Component</b>               | WCF                                                                                                                                                                                                                                                                                                                                                      |
| <b>SDL Phase</b>               | Build                                                                                                                                                                                                                                                                                                                                                    |
| <b>Applicable Technologies</b> | Generic, .NET Framework 3                                                                                                                                                                                                                                                                                                                                |
| <b>Attributes</b>              | N/A                                                                                                                                                                                                                                                                                                                                                      |
| <b>References</b>              | <a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a>                                                                                                                                                                                                                                                                                                   |
| <b>Steps</b>                   | This service does not use an authorization control. When a client calls a particular WCF service, WCF provides various authorization schemes that verify that the caller has permission to execute the service method on the server. If authorization controls are not enabled for WCF services, an authenticated user can achieve privilege escalation. |

## Example

The following configuration instructs WCF to not check the authorization level of the client when executing the service:

```
<behaviors>
 <serviceBehaviors>
 <behavior>
 ...
 <serviceAuthorization principalPermissionMode=""None"" />
 </behavior>
 </serviceBehaviors>
</behaviors>
```

Use a service authorization scheme to verify that the caller of the service method is authorized to do so. WCF provides two modes and allows the definition of a custom authorization scheme. The UseWindowsGroups mode uses Windows roles and users and the UseAspNetRoles mode uses an ASP.NET role provider, such as SQL Server, to authenticate.

## Example

The following configuration instructs WCF to make sure that the client is part of the Administrators group before executing the Add service:

```
<behaviors>
 <serviceBehaviors>
 <behavior>
 ...
 <serviceAuthorization principalPermissionMode=""UseWindowsGroups"" />
 </behavior>
 </serviceBehaviors>
</behaviors>
```

The service is then declared as the following:

```
[PrincipalPermission(SecurityAction.Demand,
Role = "\"Builtin\\Administrators\"")]
public double Add(double n1, double n2)
{
 double result = n1 + n2;
 return result;
}
```

## Implement proper authorization mechanism in ASP.NET Web API

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, MVC5
<b>Attributes</b>	N/A, Identity Provider - ADFS, Identity Provider - Azure AD
<b>References</b>	<a href="#">Authentication and Authorization in ASP.NET Web API</a>
<b>Steps</b>	<p>Role information for the application users can be derived from Azure AD or ADFS claims if the application relies on them as Identity provider or the application itself might provided it. In any of these cases, the custom authorization implementation should validate the user role information.</p> <p>Role information for the application users can be derived from Azure AD or ADFS claims if the application relies on them as Identity provider or the application itself might provided it. In any of these cases, the custom authorization implementation should validate the user role information.</p>

### Example

```
[AttributeUsage(AttributeTargets.Class | AttributeTargets.Method, Inherited = true, AllowMultiple = true)]
public class ApiAuthorizeAttribute : System.Web.Http.AuthorizeAttribute
{
 public async override Task OnAuthorizationAsync(HttpContext actionContext, CancellationToken cancellationToken)
 {
 if (actionContext == null)
 {
 throw new Exception();
 }

 if (!string.IsNullOrEmpty(base.Roles))
 {
 bool isAuthorized = ValidateRoles(actionContext);
 if (!isAuthorized)
 {
 HandleUnauthorizedRequest(actionContext);
 }
 }
 }

 base.OnAuthorization(actionContext);
}

public bool ValidateRoles(HttpContext)
{
 //Authorization logic here; returns true or false
}
}
```

All the controllers and action methods which needs to protected should be decorated with above attribute.

```
[ApiAuthorize]
public class CustomController : ApiController
{
 //Application code goes here
}
```

**Perform authorization checks in the device if it supports various actions that require different permission levels**

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A

TITLE	DETAILS
<b>Steps</b>	<p>The Device should authorize the caller to check if the caller has the required permissions to perform the action requested. For e.g. Lets say the device is a Smart Door Lock that can be monitored from the cloud, plus it provides functionalities like Remotely locking the door.</p> <p>The Smart Door Lock provides unlocking functionality only when someone physically comes near the door with a Card. In this case, the implementation of the remote command and control should be done in such a way that it does not provide any functionality to unlock the door as the cloud gateway is not authorized to send a command to unlock the door.</p>

Perform authorization checks in the Field Gateway if it supports various actions that require different permission levels

TITLE	DETAILS
<b>Component</b>	IoT Field Gateway
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>The Field Gateway should authorize the caller to check if the caller has the required permissions to perform the action requested. For e.g. there should be different permissions for an admin user interface/API used to configure a field gateway v/s devices that connect to it.</p>

# Security Frame: Communication Security | Mitigations

8/22/2017 • 14 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
Azure Event Hub	<ul style="list-style-type: none"><li>Secure communication to Event Hub using SSL/TLS</li></ul>
Dynamics CRM	<ul style="list-style-type: none"><li>Check service account privileges and check that the custom Services or ASP.NET Pages respect CRM's security</li></ul>
Azure Data Factory	<ul style="list-style-type: none"><li>Use Data management gateway while connecting On Prem SQL Server to Azure Data Factory</li></ul>
Identity Server	<ul style="list-style-type: none"><li>Ensure that all traffic to Identity Server is over HTTPS connection</li></ul>
Web Application	<ul style="list-style-type: none"><li>Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections</li><li>Configure SSL certificate for custom domain in Azure App Service</li><li>Force all traffic to Azure App Service over HTTPS connection</li><li>Enable HTTP Strict Transport Security (HSTS)</li></ul>
Database	<ul style="list-style-type: none"><li>Ensure SQL server connection encryption and certificate validation</li><li>Force Encrypted communication to SQL server</li></ul>
Azure Storage	<ul style="list-style-type: none"><li>Ensure that communication to Azure Storage is over HTTPS</li><li>Validate MD5 hash after downloading blob if HTTPS cannot be enabled</li><li>Use SMB 3.0 compatible client to ensure in-transit data encryption to Azure File Shares</li></ul>
Mobile Client	<ul style="list-style-type: none"><li>Implement Certificate Pinning</li></ul>
WCF	<ul style="list-style-type: none"><li>Enable HTTPS - Secure Transport channel</li><li>WCF: Set Message security Protection level to EncryptAndSign</li><li>WCF: Use a least-privileged account to run your WCF service</li></ul>
Web API	<ul style="list-style-type: none"><li>Force all traffic to Web APIs over HTTPS connection</li></ul>

PRODUCT/SERVICE	ARTICLE
Azure Redis Cache	<ul style="list-style-type: none"> <li>• Ensure that communication to Azure Redis Cache is over SSL</li> </ul>
IoT Field Gateway	<ul style="list-style-type: none"> <li>• Secure Device to Field Gateway communication</li> </ul>
IoT Cloud Gateway	<ul style="list-style-type: none"> <li>• Secure Device to Cloud Gateway communication using SSL/TLS</li> </ul>

## Secure communication to Event Hub using SSL/TLS

TITLE	DETAILS
Component	Azure Event Hub
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	<a href="#">Event Hubs authentication and security model overview</a>
Steps	Secure AMQP or HTTP connections to Event Hub using SSL/TLS

Check service account privileges and check that the custom Services or ASP.NET Pages respect CRM's security

TITLE	DETAILS
Component	Dynamics CRM
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Check service account privileges and check that the custom Services or ASP.NET Pages respect CRM's security

Use Data management gateway while connecting On Prem SQL Server to Azure Data Factory

TITLE	DETAILS
<b>Component</b>	Azure Data Factory
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	Linked Service Types - Azure and On Prem
<b>References</b>	<a href="#">Moving data between On Prem and Azure Data Factory</a> , <a href="#">Data management gateway</a>
<b>Steps</b>	<p>The Data Management Gateway (DMG) tool is required to connect to data sources which are protected behind corpnet or a firewall.</p> <ol style="list-style-type: none"> <li>1. Locking down the machine isolates the DMG tool and prevents malfunctioning programs from damaging or snooping on the data source machine. (E.g. latest updates must be installed, enable minimum required ports, controlled accounts provisioning, auditing enabled, disk encryption enabled etc.)</li> <li>2. Data Gateway key must be rotated at frequent intervals or whenever the DMG service account password renewes</li> <li>3. Data transits through Link Service must be encrypted</li> </ol>

## Ensure that all traffic to Identity Server is over HTTPS connection

TITLE	DETAILS
<b>Component</b>	Identity Server
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">IdentityServer3 - Keys, Signatures and Cryptography</a> , <a href="#">IdentityServer3 - Deployment</a>
<b>Steps</b>	<p>By default, IdentityServer requires all incoming connections to come over HTTPS. It is absolutely mandatory that communication with IdentityServer is done over secured transports only. There are certain deployment scenarios like SSL offloading where this requirement can be relaxed. See the Identity Server deployment page in the references for more information.</p>

## Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>Applications that use SSL, TLS, or DTLS must fully verify the X.509 certificates of the entities they connect to. This includes verification of the certificates for:</p> <ul style="list-style-type: none"> <li>• Domain name</li> <li>• Validity dates (both beginning and expiration dates)</li> <li>• Revocation status</li> <li>• Usage (for example, Server Authentication for servers, Client Authentication for clients)</li> <li>• Trust chain. Certificates must chain to a root certification authority (CA) that is trusted by the platform or explicitly configured by the administrator</li> <li>• Key length of certificate's public key must be &gt;2048 bits</li> <li>• Hashing algorithm must be SHA256 and above</li> </ul>

## Configure SSL certificate for custom domain in Azure App Service

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	EnvironmentType - Azure
<b>References</b>	<a href="#">Enable HTTPS for an app in Azure App Service</a>
<b>Steps</b>	<p>By default, Azure already enables HTTPS for every app with a wildcard certificate for the *.azurewebsites.net domain. However, like all wildcard domains, it is not as secure as using a custom domain with own certificate <a href="#">Refer</a>. It is recommended to enable SSL for the custom domain which the deployed app will be accessed through</p>

## Force all traffic to Azure App Service over HTTPS connection

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	EnvironmentType - Azure
<b>References</b>	[Enforce HTTPS on Azure App Service] <a href="https://azure.microsoft.com/documentation/articles/web-sites-configure-ssl-certificate/#4-enforce-https-on-your-app">https://azure.microsoft.com/documentation/articles/web-sites-configure-ssl-certificate/#4-enforce-https-on-your-app</a>
<b>Steps</b>	<p>Though Azure already enables HTTPS for Azure app services with a wildcard certificate for the domain *.azurewebsites.net, it does not enforce HTTPS. Visitors may still access the app using HTTP, which may compromise the app's security and hence HTTPS has to be enforced explicitly. ASP.NET MVC applications should use the <a href="#">RequireHttps filter</a> that forces an unsecured HTTP request to be re-sent over HTTPS.</p> <p>Alternatively, the URL Rewrite module, which is included with Azure App Service can be used to enforce HTTPS. URL Rewrite module enables developers to define rules that are applied to incoming requests before the requests are handed to your application. URL Rewrite rules are defined in a web.config file stored in the root of the application</p>

## Example

The following example contains a basic URL Rewrite rule that forces all incoming traffic to use HTTPS

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
 <system.webServer>
 <rewrite>
 <rules>
 <rule name="Force HTTPS" enabled="true">
 <match url="(.*)" ignoreCase="false" />
 <conditions>
 <add input="{HTTPS}" pattern="off" />
 </conditions>
 <action type="Redirect" url="https:///{HTTP_HOST}/{R:1}" appendQueryString="true"
redirectType="Permanent" />
 </rule>
 </rules>
 </rewrite>
 </system.webServer>
</configuration>
```

This rule works by returning an HTTP status code of 301 (permanent redirect) when the user requests a page using HTTP. The 301 redirects the request to the same URL as the visitor requested, but replaces the HTTP portion of the request with HTTPS. For example, [HTTP://contoso.com](http://contoso.com) would be redirected to [HTTPS://contoso.com](https://contoso.com).

## Enable HTTP Strict Transport Security (HSTS)

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">OWASP HTTP Strict Transport Security Cheat Sheet</a>
<b>Steps</b>	<p>HTTP Strict Transport Security (HSTS) is an opt-in security enhancement that is specified by a web application through the use of a special response header. Once a supported browser receives this header that browser will prevent any communications from being sent over HTTP to the specified domain and will instead send all communications over HTTPS. It also prevents HTTPS click through prompts on browsers.</p> <p>To implement HSTS, the following response header has to be configured for a website globally, either in code or in config. Strict-Transport-Security: max-age=300; includeSubDomains HSTS addresses the following threats:</p> <ul style="list-style-type: none"> <li>• User bookmarks or manually types <a href="http://example.com">http://example.com</a> and is subject to a man-in-the-middle attacker: HSTS automatically redirects HTTP requests to HTTPS for the target domain</li> <li>• Web application that is intended to be purely HTTPS inadvertently contains HTTP links or serves content over HTTP: HSTS automatically redirects HTTP requests to HTTPS for the target domain</li> <li>• A man-in-the-middle attacker attempts to intercept traffic from a victim user using an invalid certificate and hopes the user will accept the bad certificate: HSTS does not allow a user to override the invalid certificate message</li> </ul>

## Ensure SQL server connection encryption and certificate validation

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	SQL Azure
<b>Attributes</b>	SQL Version - V12
<b>References</b>	<a href="#">Best Practices on Writing Secure Connection Strings for SQL Database</a>

TITLE	DETAILS
<b>Steps</b>	<p>All communications between SQL Database and a client application are encrypted using Secure Sockets Layer (SSL) at all times. SQL Database doesn't support unencrypted connections. To validate certificates with application code or tools, explicitly request an encrypted connection and do not trust the server certificates. If your application code or tools do not request an encrypted connection, they will still receive encrypted connections</p> <p>However, they may not validate the server certificates and thus will be susceptible to "man in the middle" attacks. To validate certificates with ADO.NET application code, set <code>Encrypt=True</code> and <code>TrustServerCertificate=False</code> in the database connection string. To validate certificates via SQL Server Management Studio, open the Connect to Server dialog box. Click Encrypt connection on the Connection Properties tab</p>

## Force Encrypted communication to SQL server

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	OnPrem
<b>Attributes</b>	SQL Version - MsSQL2016, SQL Version - MsSQL2012, SQL Version - MsSQL2014
<b>References</b>	<a href="#">Enable Encrypted Connections to the Database Engine</a>
<b>Steps</b>	Enabling SSL encryption increases the security of data transmitted across networks between instances of SQL Server and applications.

## Ensure that communication to Azure Storage is over HTTPS

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Azure Storage Transport-Level Encryption – Using HTTPS</a>

TITLE	DETAILS
<b>Steps</b>	To ensure the security of Azure Storage data in-transit, always use the HTTPS protocol when calling the REST APIs or accessing objects in storage. Also, Shared Access Signatures, which can be used to delegate access to Azure Storage objects, include an option to specify that only the HTTPS protocol can be used when using Shared Access Signatures, ensuring that anybody sending out links with SAS tokens will use the proper protocol.

## Validate MD5 hash after downloading blob if HTTPS cannot be enabled

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	StorageType - Blob
<b>References</b>	<a href="#">Windows Azure Blob MD5 Overview</a>
<b>Steps</b>	<p>Windows Azure Blob service provides mechanisms to ensure data integrity both at the application and transport layers. If for any reason you need to use HTTP instead of HTTPS and you are working with block blobs, you can use MD5 checking to help verify the integrity of the blobs being transferred.</p> <p>This will help with protection from network/transport layer errors, but not necessarily with intermediary attacks. If you can use HTTPS, which provides transport level security, then using MD5 checking is redundant and unnecessary.</p>

## Use SMB 3.0 compatible client to ensure in-transit data encryption to Azure File shares

TITLE	DETAILS
<b>Component</b>	Mobile Client
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	StorageType - File
<b>References</b>	<a href="#">Azure File Storage</a> , <a href="#">Azure File Storage SMB Support for Windows Clients</a>

TITLE	DETAILS
<b>Steps</b>	Azure File Storage supports HTTPS when using the REST API, but is more commonly used as an SMB file share attached to a VM. SMB 2.1 does not support encryption, so connections are only allowed within the same region in Azure. However, SMB 3.0 supports encryption, and can be used with Windows Server 2012 R2, Windows 8, Windows 8.1, and Windows 10, allowing cross-region access and even access on the desktop.

## Implement Certificate Pinning

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, Windows Phone
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Certificate and Public Key Pinning</a>
<b>Steps</b>	<p>Certificate pinning defends against Man-In-The-Middle (MITM) attacks. Pinning is the process of associating a host with their expected X509 certificate or public key. Once a certificate or public key is known or seen for a host, the certificate or public key is associated or 'pinned' to the host.</p> <p>Thus, when an adversary attempts to do SSL MITM attack, during SSL handshake the key from attacker's server will be different from the pinned certificate's key, and the request will be discarded, thus preventing MITM. Certificate pinning can be achieved by implementing ServicePointManager's <code>ServerCertificateValidationCallback</code> delegate.</p>

### Example

```

using System;
using System.Net;
using System.Net.Security;
using System.Security.Cryptography;

namespace CertificatePinningExample
{
 class CertificatePinningExample
 {
 /* Note: In this example, we're hardcoding a the certificate's public key and algorithm for
 demonstration purposes. In a real-world application, this should be stored in a secure
 configuration area that can be updated as needed. */

 private static readonly string PINNED_ALGORITHM = "RSA";

 private static readonly string PINNED_PUBLIC_KEY = "3082010A0282010100B0E75B7CBE56D31658EF79B3A1" +
 "294D506A88DFCDD603F6EF15E7F5BCBDF32291EC50B2B82BA158E905FE6A83EE044A48258B07FAC3D6356AF09B2" +
 "3EDAB15D00507B70DB08DB9A20C7D1201417B3071A346D663A241061C151B6EC5B5B4ECCDCDBEA24F051962809" +
 "FEC499BF2D093C06E3BDA7D0BB83CDC1C2C6660B8ECB2EA30A685ADE2DC83C88314010FC7F4F0F895EDDBE5C02" +
 "ABF78E50B708E0A0EB984A9AA536BCE61A0C31DB95425C6FEE5A564B158EE7C4F0693C439AE010EF83CA8155750" +
 "09B17537C29F86071E5DD8CA50EBD8A409494F479B07574D83EDCE6F68A8F7D40447471D05BC3F5EAD7862FA748" +
 "EA3C92A60A128344B1CEF7A0B0D94E50203010001";

 public static void Main(string[] args)
 {
 HttpWebRequest request = (HttpWebRequest)WebRequest.Create("https://azure.microsoft.com");
 request.ServerCertificateValidationCallback = (sender, certificate, chain, sslPolicyErrors) =>
 {
 if (certificate == null || sslPolicyErrors != SslPolicyErrors.None)
 {
 // Error getting certificate or the certificate failed basic validation
 return false;
 }

 var targetKeyAlgorithm = new Oid(certificate.GetKeyAlgorithm()).FriendlyName;
 var targetPublicKey = certificate.GetPublicKeyString();

 if (targetKeyAlgorithm == PINNED_ALGORITHM &&
 targetPublicKey == PINNED_PUBLIC_KEY)
 {
 // Success, the certificate matches the pinned value.
 return true;
 }
 // Reject, either the key or the algorithm does not match the expected value.
 return false;
 };

 try
 {
 var response = (HttpWebResponse)request.GetResponse();
 Console.WriteLine($"Success, HTTP status code: {response.StatusCode}");
 }
 catch(Exception ex)
 {
 Console.WriteLine($"Failure, {ex.Message}");
 }
 Console.WriteLine("Press any key to end.");
 Console.ReadKey();
 }
 }
}

```

## Enable HTTPS - Secure Transport channel

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	.NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a>
<b>Steps</b>	<p>The application configuration should ensure that HTTPS is used for all access to sensitive information.</p> <ul style="list-style-type: none"> <li>• <b>EXPLANATION:</b> If an application handles sensitive information and does not use message-level encryption, then it should only be allowed to communicate over an encrypted transport channel.</li> <li>• <b>RECOMMENDATIONS:</b> Ensure that HTTP transport is disabled and enable HTTPS transport instead. For example, replace the <code>&lt;httpTransport/&gt;</code> with <code>&lt;httpsTransport/&gt;</code> tag. Do not rely on a network configuration (firewall) to guarantee that the application can only be accessed over a secure channel. From a philosophical point of view, the application should not depend on the network for its security.</li> </ul> <p>From a practical point of view, the people responsible for securing the network do not always track the security requirements of the application as they evolve.</p>

## WCF: Set Message security Protection level to EncryptAndSign

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	.NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a>

TITLE	DETAILS
<b>Steps</b>	<ul style="list-style-type: none"> <li><b>EXPLANATION:</b> When Protection level is set to "none" it will disable message protection. Confidentiality and integrity is achieved with appropriate level of setting.</li> <li><b>RECOMMENDATIONS:</b> <ul style="list-style-type: none"> <li>when <code>Mode=None</code> - Disables message protection</li> <li>when <code>Mode=Sign</code> - Signs but does not encrypt the message; should be used when data integrity is important</li> <li>when <code>Mode=EncryptAndSign</code> - Signs and encrypts the message</li> </ul> </li> </ul> <p>Consider turning off encryption and only signing your message when you just need to validate the integrity of the information without concerns of confidentiality. This may be useful for operations or service contracts in which you need to validate the original sender but no sensitive data is transmitted. When reducing the protection level, be careful that the message does not contain any personally identifiable information (PII).</p>

### Example

Configuring the service and the operation to only sign the message is shown in the following examples. Service Contract Example of `ProtectionLevel.Sign`: The following is an example of using `ProtectionLevel.Sign` at the Service Contract level:

```
[ServiceContract(ProtectionLevel=ProtectionLevel.Sign)]
public interface IService
{
 string GetData(int value);
}
```

### Example

Operation Contract Example of `ProtectionLevel.Sign` (for Granular Control): The following is an example of using `ProtectionLevel.Sign` at the OperationContract level:

```
[OperationContract(ProtectionLevel=ProtectionLevel.Sign)]
string GetData(int value);
```

## WCF: Use a least-privileged account to run your WCF service

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	.NET Framework 3
<b>Attributes</b>	N/A

TITLE	DETAILS
References	<a href="#">MSDN</a>
Steps	<ul style="list-style-type: none"> <li><b>EXPLANATION:</b> Do not run WCF services under admin or high privilege account. in case of services compromise it will result in high impact.</li> <li><b>RECOMMENDATIONS:</b> Use a least-privileged account to host your WCF service because it will reduce your application's attack surface and reduce the potential damage if you are attacked. If the service account requires additional access rights on infrastructure resources such as MSMQ, the event log, performance counters, and the file system, appropriate permissions should be given to these resources so that the WCF service can run successfully.</li> </ul> <p>If your service needs to access specific resources on behalf of the original caller, use impersonation and delegation to flow the caller's identity for a downstream authorization check. In a development scenario, use the local network service account, which is a special built-in account that has reduced privileges. In a production scenario, create a least-privileged custom domain service account.</p>

## Force all traffic to Web APIs over HTTPS connection

TITLE	DETAILS
Component	Web API
SDL Phase	Build
Applicable Technologies	MVC5, MVC6
Attributes	N/A
References	<a href="#">Enforcing SSL in a Web API Controller</a>
Steps	If an application has both an HTTPS and an HTTP binding, clients can still use HTTP to access the site. To prevent this, use an action filter to ensure that requests to protected APIs are always over HTTPS.

### Example

The following code shows a Web API authentication filter that checks for SSL:

```

public class RequireHttpsAttribute : AuthorizationFilterAttribute
{
 public override void OnAuthorization(HttpActionContext actionContext)
 {
 if (actionContext.Request.RequestUri.Scheme != Uri.UriSchemeHttps)
 {
 actionContext.Response = new HttpResponseMessage(System.Net.HttpStatusCode.Forbidden)
 {
 ReasonPhrase = "HTTPS Required"
 };
 }
 else
 {
 base.OnAuthorization(actionContext);
 }
 }
}

```

Add this filter to any Web API actions that require SSL:

```

public class ValuesController : ApiController
{
 [RequireHttps]
 public HttpResponseMessage Get() { ... }
}

```

## Ensure that communication to Azure Redis Cache is over SSL

TITLE	DETAILS
<b>Component</b>	Azure Redis Cache
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Azure Redis SSL support</a>
<b>Steps</b>	Redis server does not support SSL out of the box, but Azure Redis Cache does. If you are connecting to Azure Redis Cache and your client supports SSL, like StackExchange.Redis, then you should use SSL. By default non-SSL port is disabled for new Azure Redis Cache instances. Ensure that the secure defaults are not changed unless there is a dependency on SSL support for redis clients.

Please note that Redis is designed to be accessed by trusted clients inside trusted environments. This means that usually it is not a good idea to expose the Redis instance directly to the internet or, in general, to an environment where untrusted clients can directly access the Redis TCP port or UNIX socket.

## Secure Device to Field Gateway communication

TITLE	DETAILS
<b>Component</b>	IoT Field Gateway
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	For IP based devices, the communication protocol could typically be encapsulated in a SSL/TLS channel to protect data in transit. For other protocols that do not support SSL/TLS investigate if there are secure versions of the protocol that provide security at transport or message layer.

## Secure Device to Cloud Gateway communication using SSL/TLS

TITLE	DETAILS
<b>Component</b>	IoT Cloud Gateway
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Choose your Communication Protocol</a>
<b>Steps</b>	Secure HTTP/AMQP or MQTT protocols using SSL/TLS.

# Security Frame: Configuration Management | Mitigations

8/22/2017 • 20 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
<b>Web Application</b>	<ul style="list-style-type: none"><li>• Implement Content Security Policy (CSP), and disable inline javascript</li><li>• Enable browser's XSS filter</li><li>• ASP.NET applications must disable tracing and debugging prior to deployment</li><li>• Access third-party javascripts from trusted sources only</li><li>• Ensure that authenticated ASP.NET pages incorporate UI Redressing or click-jacking defenses</li><li>• Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web Applications</li><li>• Enable ValidateRequest attribute on ASP.NET Pages</li><li>• Use locally-hosted latest versions of JavaScript libraries</li><li>• Disable automatic MIME sniffing</li><li>• Remove standard server headers on Windows Azure Web Sites to avoid fingerprinting</li></ul>
<b>Database</b>	<ul style="list-style-type: none"><li>• Configure a Windows Firewall for Database Engine Access</li></ul>
<b>Web API</b>	<ul style="list-style-type: none"><li>• Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web API</li><li>• Encrypt sections of Web API's configuration files that contain sensitive data</li></ul>
<b>IoT Device</b>	<ul style="list-style-type: none"><li>• Ensure that all admin interfaces are secured with strong credentials</li><li>• Ensure that unknown code cannot execute on devices</li><li>• Encrypt OS and additional partitions of IoT Device with bit-locker</li><li>• Ensure that only the minimum services/features are enabled on devices</li></ul>
<b>IoT Field Gateway</b>	<ul style="list-style-type: none"><li>• Encrypt OS and additional partitions of IoT Field Gateway with bit-locker</li><li>• Ensure that the default login credentials of the field gateway are changed during installation</li></ul>
<b>IoT Cloud Gateway</b>	<ul style="list-style-type: none"><li>• Ensure that the Cloud Gateway implements a process to keep the connected devices firmware up to date</li></ul>

PRODUCT/SERVICE	ARTICLE
<b>Machine Trust Boundary</b>	<ul style="list-style-type: none"> <li>• Ensure that devices have end-point security controls configured as per organizational policies</li> </ul>
<b>Azure Storage</b>	<ul style="list-style-type: none"> <li>• Ensure secure management of Azure storage access keys</li> <li>• Ensure that only trusted origins are allowed if CORS is enabled on Azure storage</li> </ul>
<b>WCF</b>	<ul style="list-style-type: none"> <li>• Enable WCF's service throttling feature</li> <li>• WCF-Information disclosure through metadata</li> </ul>

## Implement Content Security Policy (CSP), and disable inline javascript

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">An Introduction to Content Security Policy</a> , <a href="#">Content Security Policy Reference</a> , <a href="#">Security features</a> , <a href="#">Introduction to content security policy</a> , <a href="#">Can I use CSP?</a>

TITLE	DETAILS
<b>Steps</b>	<p>Content Security Policy (CSP) is a defense-in-depth security mechanism, a W3C standard, that enables web application owners to have control on the content embedded in their site. CSP is added as an HTTP response header on the web server and is enforced on the client side by browsers. It is a whitelist-based policy - a website can declare a set of trusted domains from which active content such as JavaScript can be loaded.</p> <p>CSP provides the following security benefits:</p> <ul style="list-style-type: none"> <li>• <b>Protection against XSS:</b> If a page is vulnerable to XSS, an attacker can exploit it in 2 ways: <ul style="list-style-type: none"> <li>◦ Inject <code>&lt;script&gt;malicious code&lt;/script&gt;</code>. This exploit will not work due to CSP's Base Restriction-1</li> <li>◦ Inject <code>&lt;script src="http://attacker.com/maliciousCode.js"/&gt;</code>. This exploit will not work since the attacker controlled domain will not be in CSP's whitelist of domains</li> </ul> </li> <li>• <b>Control over data exfiltration:</b> If any malicious content on a webpage attempts to connect to an external website and steal data, the connection will be aborted by CSP. This is because the target domain will not be in CSP's whitelist</li> <li>• <b>Defense against click-jacking:</b> click-jacking is an attack technique using which an adversary can frame a genuine website and force users to click on UI elements. Currently defense against click-jacking is achieved by configuring a response header- X-Frame-Options. Not all browsers respect this header and going forward CSP will be a standard way to defend against click-jacking</li> <li>• <b>Real-time attack reporting:</b> If there is an injection attack on a CSP-enabled website, browsers will automatically trigger a notification to an endpoint configured on the webserver. This way, CSP serves as a real-time warning system.</li> </ul>

## Example

Example policy:

```
Content-Security-Policy: default-src 'self'; script-src 'self' www.google-analytics.com
```

This policy allows scripts to load only from the web application's server and google analytics server. Scripts loaded from any other site will be rejected. When CSP is enabled on a website, the following features are automatically disabled to mitigate XSS attacks.

## Example

Inline scripts will not execute. Following are examples of inline scripts

```
<script> some Javascript code </script>
Event handling attributes of HTML tags (e.g., <button onclick="function(){}
javascript:alert(1);"
```

## Example

Strings will not be evaluated as code.

```
Example: var str="alert(1)"; eval(str);
```

## Enable browser's XSS filter

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">XSS Protection Filter</a>
<b>Steps</b>	<p>X-XSS-Protection response header configuration controls the browser's cross site script filter. This response header can have following values:</p> <ul style="list-style-type: none"><li>• <code>0</code>: This will disable the filter</li><li>• <code>1</code>: <code>Filter enabled</code>. If a cross-site scripting attack is detected, in order to stop the attack, the browser will sanitize the page</li><li>• <code>1: mode=block</code> : <code>Filter enabled</code>. Rather than sanitize the page, when a XSS attack is detected, the browser will prevent rendering of the page</li><li>• <code>1: report=http://[YOURDOMAIN]/your_report_URI</code> : <code>Filter enabled</code>. The browser will sanitize the page and report the violation.</li></ul> <p>This is a Chromium function utilizing CSP violation reports to send details to a URI of your choice. The last 2 options are considered safe values.</p>

## ASP.NET applications must disable tracing and debugging prior to deployment

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	<a href="#">ASP.NET Debugging Overview</a> , <a href="#">ASP.NET Tracing Overview</a> , <a href="#">How to: Enable Tracing for an ASP.NET Application</a> , <a href="#">How to: Enable Debugging for ASP.NET Applications</a>
<b>Steps</b>	When tracing is enabled for the page, every browser requesting it also obtains the trace information that contains data about internal server state and workflow. That information could be security sensitive. When debugging is enabled for the page, errors happening on the server result in a full stack trace data presented to the browser. That data may expose security-sensitive information about the server's workflow.

## Access third-party javascripts from trusted sources only

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	third-party JavaScripts should be referenced only from trusted sources. The reference endpoints should always be on SSL.

## Ensure that authenticated ASP.NET pages incorporate UI Redressing or click-jacking defenses

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">OWASP click-jacking Defense Cheat Sheet</a> , <a href="#">IE Internals - Combating click-jacking With X-Frame-Options</a>

TITLE	DETAILS
<b>Steps</b>	<p>click-jacking, also known as a "UI redress attack", is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top-level page.</p> <p>This layering is achieved by crafting a malicious page with an iframe, which loads the victim's page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to another page, most likely owned by another application, domain, or both. To prevent click-jacking attacks, set the proper X-Frame-Options HTTP response headers that instruct the browser to not allow framing from other domains</p>

## Example

The X-FRAME-OPTIONS header can be set via IIS web.config. Web.config code snippet for sites that should never be framed:

```
<system.webServer>
 <httpProtocol>
 <customHeader>
 <add name="X-FRAME-OPTIONS" value="DENY"/>
 </customHeaders>
 </httpProtocol>
</system.webServer>
```

## Example

Web.config code for sites that should only be framed by pages in the same domain:

```
<system.webServer>
 <httpProtocol>
 <customHeader>
 <add name="X-FRAME-OPTIONS" value="SAMEORIGIN"/>
 </customHeaders>
 </httpProtocol>
</system.webServer>
```

## Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web Applications

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Web Forms, MVC5
<b>Attributes</b>	N/A
<b>References</b>	N/A

TITLE	DETAILS
<b>Steps</b>	<p>Browser security prevents a web page from making AJAX requests to another domain. This restriction is called the same-origin policy, and prevents a malicious site from reading sensitive data from another site. However, sometimes it might be required to expose APIs securely which other sites can consume. Cross Origin Resource Sharing (CORS) is a W3C standard that allows a server to relax the same-origin policy. Using CORS, a server can explicitly allow some cross-origin requests while rejecting others.</p> <p>CORS is safer and more flexible than earlier techniques such as JSONP. At its core, enabling CORS translates to adding a few HTTP response headers (Access-Control-*) to the web application and this can be done in a couple of ways.</p>

## Example

If access to Web.config is available, then CORS can be added through the following code:

```
<system.webServer>
 <httpProtocol>
 <customHeaders>
 <clear />
 <add name="Access-Control-Allow-Origin" value="http://example.com" />
 </customHeaders>
 </httpProtocol>
```

## Example

If access to web.config is not available, then CORS can be configured by adding the following CSharp code:

```
HttpContext.Response.AppendHeader("Access-Control-Allow-Origin", "http://example.com")
```

Please note that it is critical to ensure that the list of origins in "Access-Control-Allow-Origin" attribute is set to a finite and trusted set of origins. Failing to configure this appropriately (e.g., setting the value as '\*') will allow malicious sites to trigger cross origin requests to the web application >without any restrictions, thereby making the application vulnerable to CSRF attacks.

## Enable ValidateRequest attribute on ASP.NET Pages

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Web Forms, MVC5
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Request Validation - Preventing Script Attacks</a>

TITLE	DETAILS
<b>Steps</b>	<p>Request validation, a feature of ASP.NET since version 1.1, prevents the server from accepting content containing un-encoded HTML. This feature is designed to help prevent some script-injection attacks whereby client script code or HTML can be unknowingly submitted to a server, stored, and then presented to other users. We still strongly recommend that you validate all input data and HTML encode it when appropriate.</p> <p>Request validation is performed by comparing all input data to a list of potentially dangerous values. If a match occurs, ASP.NET raises an <code>HttpRequestValidationException</code>. By default, Request Validation feature is enabled.</p>

## Example

However, this feature can be disabled at page level:

```
<%@ Page validateRequest="false" %>
```

or, at application level

```
<configuration>
 <system.web>
 <pages validateRequest="false" />
 </system.web>
</configuration>
```

Please note that Request Validation feature is not supported, and is not part of MVC6 pipeline.

## Use locally-hosted latest versions of JavaScript libraries

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A

TITLE	DETAILS
<b>Steps</b>	<p>Developers using standard JavaScript libraries like JQuery must use approved versions of common JavaScript libraries that do not contain known security flaws. A good practice is to use the most latest version of the libraries, since they contain security fixes for known vulnerabilities in their older versions.</p> <p>If the most recent release cannot be used due to compatibility reasons, the below minimum versions should be used.</p> <p>Acceptable minimum versions:</p> <ul style="list-style-type: none"> <li>• <b>JQuery</b> <ul style="list-style-type: none"> <li>◦ JQuery 1.7.1</li> <li>◦ JQueryUI 1.10.0</li> <li>◦ JQuery Validate 1.9</li> <li>◦ JQuery Mobile 1.0.1</li> <li>◦ JQuery Cycle 2.99</li> <li>◦ JQuery DataTables 1.9.0</li> </ul> </li> <li>• <b>Ajax Control Toolkit</b> <ul style="list-style-type: none"> <li>◦ Ajax Control Toolkit 40412</li> </ul> </li> <li>• <b>ASP.NET Web Forms and Ajax</b> <ul style="list-style-type: none"> <li>◦ ASP.NET Web Forms and Ajax 4</li> <li>◦ ASP.NET Ajax 3.5</li> </ul> </li> <li>• <b>ASP.NET MVC</b> <ul style="list-style-type: none"> <li>◦ ASP.NET MVC 3.0</li> </ul> </li> </ul> <p>Never load any JavaScript library from external sites such as public CDNs</p>

## Disable automatic MIME sniffing

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">IE8 Security Part V: Comprehensive Protection, MIME type</a>
<b>Steps</b>	<p>The X-Content-Type-Options header is an HTTP header that allows developers to specify that their content should not be MIME-sniffed. This header is designed to mitigate MIME-Sniffing attacks. For each page that could contain user controllable content, you must use the HTTP Header X-Content-Type-Options:nosniff. To enable the required header globally for all pages in the application, you can do one of the following</p>

### Example

Add the header in the web.config file if the application is hosted by Internet Information Services (IIS) 7 onwards.

```
<system.webServer>
<httpProtocol>
<customHeaders>
<add name="X-Content-Type-Options" value="nosniff"/>
</customHeaders>
</httpProtocol>
</system.webServer>
```

## Example

Add the header through the global Application\_BeginRequest

```
void Application_BeginRequest(object sender, EventArgs e)
{
 this.Response.Headers["X-Content-Type-Options"] = "nosniff";
}
```

## Example

Implement custom HTTP module

```
public class XContentTypeOptionsModule : IHttpModule
{
 #region IHttpModule Members
 public void Dispose()
 {
 }
 public void Init(HttpApplication context)
 {
 context.PreSendRequestHeaders += new EventHandler(context_PreSendRequestHeaders);
 }
 #endregion
 void context_PreSendRequestHeaders(object sender, EventArgs e)
 {
 HttpApplication application = sender as HttpApplication;
 if (application == null)
 return;
 if (application.Response.Headers["X-Content-Type-Options "] != null)
 return;
 application.Response.Headers.Add("X-Content-Type-Options ", "nosniff");
 }
}
```

## Example

You can enable the required header only for specific pages by adding it to individual responses:

```
this.Response.Headers["X-Content-Type-Options"] = "nosniff";
```

## Remove standard server headers on Windows Azure Web Sites to avoid fingerprinting

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build

TITLE	DETAILS
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	EnvironmentType - Azure
<b>References</b>	<a href="#">Removing standard server headers on Windows Azure Web Sites</a>
<b>Steps</b>	Headers such as Server, X-Powered-By, X-AspNet-Version reveal information about the server and the underlying technologies. It is recommended to suppress these headers thereby preventing fingerprinting the application

## Configure a Windows Firewall for Database Engine Access

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	SQL Azure, OnPrem
<b>Attributes</b>	N/A, SQL Version - V12
<b>References</b>	<a href="#">How to configure an Azure SQL database firewall, Configure a Windows Firewall for Database Engine Access</a>
<b>Steps</b>	Firewall systems help prevent unauthorized access to computer resources. To access an instance of the SQL Server Database Engine through a firewall, you must configure the firewall on the computer running SQL Server to allow access

## Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web API

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC 5
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Enabling Cross-Origin Requests in ASP.NET Web API 2, ASP.NET Web API - CORS Support in ASP.NET Web API 2</a>

TITLE	DETAILS
<b>Steps</b>	<p>Browser security prevents a web page from making AJAX requests to another domain. This restriction is called the same-origin policy, and prevents a malicious site from reading sensitive data from another site. However, sometimes it might be required to expose APIs securely which other sites can consume. Cross Origin Resource Sharing (CORS) is a W3C standard that allows a server to relax the same-origin policy.</p> <p>Using CORS, a server can explicitly allow some cross-origin requests while rejecting others. CORS is safer and more flexible than earlier techniques such as JSONP.</p>

## Example

In the App\_Start/WebApiConfig.cs, add the following code to the WebApiConfig.Register method

```
using System.Web.Http;
namespace WebService
{
 public static class WebApiConfig
 {
 public static void Register(HttpConfiguration config)
 {
 // New code
 config.EnableCors();

 config.Routes.MapHttpRoute(
 name: "DefaultApi",
 routeTemplate: "api/{controller}/{id}",
 defaults: new { id = RouteParameter.Optional }
);
 }
 }
}
```

## Example

EnableCors attribute can be applied to action methods in a controller as follows:

```

public class ResourcesController : ApiController
{
 [EnableCors("http://localhost:55912", // Origin
 null, // Request headers
 "GET", // HTTP methods
 "bar", // Response headers
 SupportsCredentials=true // Allow credentials
)]
 public HttpResponseMessage Get(int id)
 {
 var resp = Request.CreateResponse(HttpStatusCode.NoContent);
 resp.Headers.Add("bar", "a bar value");
 return resp;
 }
 [EnableCors("http://localhost:55912", // Origin
 "Accept, Origin, Content-Type", // Request headers
 "PUT", // HTTP methods
 PreflightMaxAge=600 // Preflight cache duration
)]
 public HttpResponseMessage Put(Resource data)
 {
 return Request.CreateResponse(HttpStatusCode.OK, data);
 }
 [EnableCors("http://localhost:55912", // Origin
 "Accept, Origin, Content-Type", // Request headers
 "POST", // HTTP methods
 PreflightMaxAge=600 // Preflight cache duration
)]
 public HttpResponseMessage Post(Resource data)
 {
 return Request.CreateResponse(HttpStatusCode.OK, data);
 }
}

```

Please note that it is critical to ensure that the list of origins in EnableCors attribute is set to a finite and trusted set of origins. Failing to configure this inappropriately (e.g., setting the value as '\*') will allow malicious sites to trigger cross origin requests to the API without any restrictions, thereby making the API vulnerable to CSRF attacks. EnableCors can be decorated at controller level.

## Example

To disable CORS on a particular method in a class, the DisableCors attribute can be used as shown below:

```

[EnableCors("http://example.com", "Accept, Origin, Content-Type", "POST")]
public class ResourcesController : ApiController
{
 public HttpResponseMessage Put(Resource data)
 {
 return Request.CreateResponse(HttpStatusCode.OK, data);
 }
 public HttpResponseMessage Post(Resource data)
 {
 return Request.CreateResponse(HttpStatusCode.OK, data);
 }
 // CORS not allowed because of the [DisableCors] attribute
 [DisableCors]
 public HttpResponseMessage Delete(int id)
 {
 return Request.CreateResponse(HttpStatusCode.NoContent);
 }
}

```

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC 6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Enabling Cross-Origin Requests (CORS) in ASP.NET Core 1.0</a>
<b>Steps</b>	In ASP.NET Core 1.0, CORS can be enabled either using middleware or using MVC. When using MVC to enable CORS the same CORS services are used, but the CORS middleware is not.

**Approach-1** Enabling CORS with middleware: To enable CORS for the entire application add the CORS middleware to the request pipeline using the `UseCors` extension method. A cross-origin policy can be specified when adding the CORS middleware using the `CorsPolicyBuilder` class. There are two ways to do this:

### Example

The first is to call `UseCors` with a lambda. The lambda takes a `CorsPolicyBuilder` object:

```
public void Configure(IApplicationBuilder app)
{
 app.UseCors(builder =>
 builder.WithOrigins("http://example.com")
 .WithMethods("GET", "POST", "HEAD")
 .WithHeaders("accept", "content-type", "origin", "x-custom-header"));
}
```

### Example

The second is to define one or more named CORS policies, and then select the policy by name at run time.

```
public void ConfigureServices(IServiceCollection services)
{
 services.AddCors(options =>
 {
 options.AddPolicy("AllowSpecificOrigin",
 builder => builder.WithOrigins("http://example.com"));
 });
}

public void Configure(IApplicationBuilder app)
{
 app.UseCors("AllowSpecificOrigin");
 app.Run(async (context) =>
 {
 await context.Response.WriteAsync("Hello World!");
 });
}
```

**Approach-2** Enabling CORS in MVC: Developers can alternatively use MVC to apply specific CORS per action, per controller, or globally for all controllers.

### Example

Per action: To specify a CORS policy for a specific action add the [EnableCors] attribute to the action. Specify the policy name.

```
public class HomeController : Controller
{
 [EnableCors("AllowSpecificOrigin")]
 public IActionResult Index()
 {
 return View();
 }
}
```

### Example

Per controller:

```
[EnableCors("AllowSpecificOrigin")]
public class HomeController : Controller
{
```

### Example

Globally:

```
public void ConfigureServices(IServiceCollection services)
{
 services.AddMvc();
 services.Configure<MvcOptions>(options =>
 {
 options.Filters.Add(new CorsAuthorizationFilterFactory("AllowSpecificOrigin"));
 });
}
```

Please note that it is critical to ensure that the list of origins in EnableCors attribute is set to a finite and trusted set of origins. Failing to configure this inappropriately (e.g., setting the value as '\*') will allow malicious sites to trigger cross origin requests to the API without any restrictions, thereby making the API vulnerable to CSRF attacks.

### Example

To disable CORS for a controller or action, use the [DisableCors] attribute.

```
[DisableCors]
public IActionResult About()
{
 return View();
}
```

## Encrypt sections of Web API's configuration files that contain sensitive data

TITLE	DETAILS
Component	Web API
SDL Phase	Deployment
Applicable Technologies	Generic

TITLE	DETAILS
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">How To: Encrypt Configuration Sections in ASP.NET 2.0 Using DPAPI, Specifying a Protected Configuration Provider, Using Azure Key Vault to protect application secrets</a>
<b>Steps</b>	Configuration files such as the Web.config, appsettings.json are often used to hold sensitive information, including user names, passwords, database connection strings, and encryption keys. If you do not protect this information, your application is vulnerable to attackers or malicious users obtaining sensitive information such as account user names and passwords, database names and server names. Based on the deployment type (azure/on-prem), encrypt the sensitive sections of config files using DPAPI or services like Azure Key Vault.

## Ensure that all admin interfaces are secured with strong credentials

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Any administrative interfaces that the device or field gateway exposes should be secured using strong credentials. Also, any other exposed interfaces like WiFi, SSH, File shares, FTP should be secured with strong credentials. Default weak passwords should not be used.

## Ensure that unknown code cannot execute on devices

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Enabling Secure Boot and bit-locker Device Encryption on Windows 10 IoT Core</a>

TITLE	DETAILS
<b>Steps</b>	UEFI Secure Boot restricts the system to only allow execution of binaries signed by a specified authority. This feature prevents unknown code from being executed on the platform and potentially weakening the security posture of it. Enable UEFI Secure Boot and restrict the list of certificate authorities that are trusted for signing code. Sign all code that is deployed on the device using one of the trusted authorities.

## Encrypt OS and additional partitions of IoT Device with bit-locker

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Windows 10 IoT Core implements a lightweight version of bit-locker Device Encryption, which has a strong dependency on the presence of a TPM on the platform, including the necessary preOS protocol in UEFI that conducts the necessary measurements. These preOS measurements ensure that the OS later has a definitive record of how the OS was launched. Encrypt OS partitions using bit-locker and any additional partitions also in case they store any sensitive data.

## Ensure that only the minimum services/features are enabled on devices

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Do not enable or turn off any features or services in the OS that is not required for the functioning of the solution. For e.g. if the device does not require a UI to be deployed, install Windows IoT Core in headless mode.

## Encrypt OS and additional partitions of IoT Field Gateway with bit-

## locker

TITLE	DETAILS
<b>Component</b>	IoT Field Gateway
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Windows 10 IoT Core implements a lightweight version of bit-locker Device Encryption, which has a strong dependency on the presence of a TPM on the platform, including the necessary preOS protocol in UEFI that conducts the necessary measurements. These preOS measurements ensure that the OS later has a definitive record of how the OS was launched. Encrypt OS partitions using bit-locker and any additional partitions also in case they store any sensitive data.

Ensure that the default login credentials of the field gateway are changed during installation

TITLE	DETAILS
<b>Component</b>	IoT Field Gateway
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Ensure that the default login credentials of the field gateway are changed during installation

Ensure that the Cloud Gateway implements a process to keep the connected devices firmware up to date

TITLE	DETAILS
<b>Component</b>	IoT Cloud Gateway
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic

TITLE	DETAILS
<b>Attributes</b>	Gateway choice - Azure IoT Hub
<b>References</b>	<a href="#">IoT Hub Device Management Overview</a> , <a href="#">How to update Device Firmware</a>
<b>Steps</b>	LWM2M is a protocol from the Open Mobile Alliance for IoT Device Management. Azure IoT device management allows to interact with physical devices using device jobs. Ensure that the Cloud Gateway implements a process to routinely keep the device and other configuration data up to date using Azure IoT Hub Device Management.

Ensure that devices have end-point security controls configured as per organizational policies

TITLE	DETAILS
<b>Component</b>	Machine Trust Boundary
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Ensure that devices have end-point security controls such as bit-locker for disk-level encryption, anti-virus with updated signatures, host based firewall, OS upgrades, group policies etc. are configured as per organizational security policies.

Ensure secure management of Azure storage access keys

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Azure Storage security guide - Managing Your Storage Account Keys</a>

TITLE	DETAILS
<b>Steps</b>	<p>Key Storage: It is recommended to store the Azure Storage access keys in Azure Key Vault as a secret and have the applications retrieve the key from key vault. This is recommended due to the following reasons:</p> <ul style="list-style-type: none"> <li>• The application will never have the storage key hardcoded in a configuration file, which removes that avenue of somebody getting access to the keys without specific permission</li> <li>• Access to the keys can be controlled using Azure Active Directory. This means an account owner can grant access to the handful of applications that need to retrieve the keys from Azure Key Vault. Other applications will not be able to access the keys without granting them permission specifically</li> <li>• Key Regeneration: It is recommended to have a process in place to regenerate Azure storage access keys for security reasons. Details on why and how to plan for key regeneration are documented in the Azure Storage Security Guide reference article</li> </ul>

## Ensure that only trusted origins are allowed if CORS is enabled on Azure storage

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">CORS Support for the Azure Storage Services</a>
<b>Steps</b>	<p>Azure Storage allows you to enable CORS – Cross Origin Resource Sharing. For each storage account, you can specify domains that can access the resources in that storage account. By default, CORS is disabled on all services. You can enable CORS by using the REST API or the storage client library to call one of the methods to set the service policies.</p>

## Enable WCF's service throttling feature

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	.NET Framework 3

TITLE	DETAILS
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a>
<b>Steps</b>	<p>Not placing a limit on the use of system resources could result in resource exhaustion and ultimately a denial of service.</p> <ul style="list-style-type: none"> <li>• <b>EXPLANATION:</b> Windows Communication Foundation (WCF) offers the ability to throttle service requests. Allowing too many client requests can flood a system and exhaust its resources. On the other hand, allowing only a small number of requests to a service can prevent legitimate users from using the service. Each service should be individually tuned to and configured to allow the appropriate amount of resources.</li> <li>• <b>RECOMMENDATIONS</b> Enable WCF's service throttling feature and set limits appropriate for your application.</li> </ul>

## Example

The following is an example configuration with throttling enabled:

```
<system.serviceModel>
 <behaviors>
 <serviceBehaviors>
 <behavior name="Throttled">
 <serviceThrottling maxConcurrentCalls="[YOUR SERVICE VALUE]" maxConcurrentSessions="[YOUR SERVICE VALUE]"
maxConcurrentInstances="[YOUR SERVICE VALUE]" />
 ...
 </behavior>
 </serviceBehaviors>
</system.serviceModel>
```

## WCF-Information disclosure through metadata

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	.NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a>
<b>Steps</b>	<p>Metadata can help attackers learn about the system and plan a form of attack. WCF services can be configured to expose metadata. Metadata gives detailed service description information and should not be broadcast in production environments. The <code>HttpGetEnabled</code> / <code>HttpsGetEnabled</code> properties of the <code>ServiceMetaData</code> class defines whether a service will expose the metadata</p>

## Example

The code below instructs WCF to broadcast a service's metadata

```
ServiceMetadataBehavior smb = new ServiceMetadataBehavior();
smb.HttpGetEnabled = true;
smb.HttpGetUrl = new Uri(EndPointAddress);
Host.Description.Behaviors.Add(smb);
```

Do not broadcast service metadata in a production environment. Set the `HttpGetEnabled` / `HttpsGetEnabled` properties of the `ServiceMetadata` class to false.

### Example

The code below instructs WCF to not broadcast a service's metadata.

```
ServiceMetadataBehavior smb = new ServiceMetadataBehavior();
smb.HttpGetEnabled = false;
smb.HttpGetUrl = new Uri(EndPointAddress);
Host.Description.Behaviors.Add(smb);
```

# Security Frame: Cryptography | Mitigations

8/22/2017 • 14 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
<b>Web Application</b>	<ul style="list-style-type: none"><li>• Use only approved symmetric block ciphers and key lengths</li><li>• Use approved block cipher modes and initialization vectors for symmetric ciphers</li><li>• Use approved asymmetric algorithms, key lengths, and padding</li><li>• Use approved random number generators</li><li>• Do not use symmetric stream ciphers</li><li>• Use approved MAC/HMAC/keyed hash algorithms</li><li>• Use only approved cryptographic hash functions</li></ul>
<b>Database</b>	<ul style="list-style-type: none"><li>• Use strong encryption algorithms to encrypt data in the database</li><li>• SSIS packages should be encrypted and digitally signed</li><li>• Add digital signature to critical database securables</li><li>• Use SQL server EKM to protect encryption keys</li><li>• Use AlwaysEncrypted feature if encryption keys should not be revealed to Database engine</li></ul>
<b>IoT Device</b>	<ul style="list-style-type: none"><li>• Store Cryptographic Keys securely on IoT Device</li></ul>
<b>IoT Cloud Gateway</b>	<ul style="list-style-type: none"><li>• Generate a random symmetric key of sufficient length for authentication to IoT Hub</li></ul>
<b>Dynamics CRM Mobile Client</b>	<ul style="list-style-type: none"><li>• Ensure a device management policy is in place that requires a use PIN and allows remote wiping</li></ul>
<b>Dynamics CRM Outlook Client</b>	<ul style="list-style-type: none"><li>• Ensure a device management policy is in place that requires a PIN/password/auto lock and encrypts all data (e.g. Bitlocker)</li></ul>
<b>Identity Server</b>	<ul style="list-style-type: none"><li>• Ensure that signing keys are rolled over when using Identity Server</li><li>• Ensure that cryptographically strong client ID, client secret are used in Identity Server</li></ul>

## Use only approved symmetric block ciphers and key lengths

TITLE	DETAILS
<b>Component</b>	Web Application

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>Products must use only those symmetric block ciphers and associated key lengths which have been explicitly approved by the Crypto Advisor in your organization. Approved symmetric algorithms at Microsoft include the following block ciphers:</p> <ul style="list-style-type: none"> <li>• For new code AES-128, AES-192, and AES-256 are acceptable</li> <li>• For backward compatibility with existing code, three-key 3DES is acceptable</li> <li>• For products using symmetric block ciphers: <ul style="list-style-type: none"> <li>◦ Advanced Encryption Standard (AES) is required for new code</li> <li>◦ Three-key triple Data Encryption Standard (3DES) is permissible in existing code for backward compatibility</li> <li>◦ All other block ciphers, including RC2, DES, 2 Key 3DES, DESX, and Skipjack, may only be used for decrypting old data, and must be replaced if used for encryption</li> </ul> </li> <li>• For symmetric block encryption algorithms, a minimum key length of 128 bits is required. The only block encryption algorithm recommended for new code is AES (AES-128, AES-192 and AES-256 are all acceptable)</li> <li>• Three-key 3DES is currently acceptable if already in use in existing code; transition to AES is recommended. DES, DESX, RC2, and Skipjack are no longer considered secure. These algorithms may only be used for decrypting existing data for the sake of backward-compatibility, and data should be re-encrypted using a recommended block cipher</li> </ul> <p>Please note that all symmetric block ciphers must be used with an approved cipher mode, which requires use of an appropriate initialization vector (IV). An appropriate IV, is typically a random number and never a constant value</p> <p>The use of legacy or otherwise unapproved crypto algorithms and smaller key lengths for reading existing data (as opposed to writing new data) may be permitted after your organization's Crypto Board review. However, you must file for an exception against this requirement. Additionally, in enterprise deployments, products should consider warning administrators when weak crypto is used to read data. Such warnings should be explanatory and actionable. In some cases, it may be appropriate to have Group Policy control the use of weak crypto</p> <p>Allowed .NET algorithms for managed crypto agility (in order of preference)</p> <ul style="list-style-type: none"> <li>• AesCng (FIPS compliant)</li> </ul>

TITLE	DETAILS
	<ul style="list-style-type: none"> <li>AuthenticatedAesCng (FIPS compliant)</li> <li>AESCryptoServiceProvider (FIPS compliant)</li> <li>AESManaged (non-FIPS-compliant)</li> </ul> <p>Please note that none of these algorithms can be specified via the <code>SymmetricAlgorithm.Create</code> or <code>CryptoConfig.CreateFromName</code> methods without making changes to the machine.config file. Also, note that AES in versions of .NET prior to .NET 3.5 is named <code>RijndaelManaged</code>, and <code>AesCng</code> and <code>AuthenticatedAesCng</code> are &gt; available through CodePlex and require CNG in the underlying OS</p>

## Use approved block cipher modes and initialization vectors for symmetric ciphers

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>All symmetric block ciphers must be used with an approved symmetric cipher mode. The only approved modes are CBC and CTS. In particular, the electronic code book (ECB) mode of operation should be avoided; use of ECB requires your organization's Crypto Board review. All usage of OFB, CFB, CTR, CCM, and GCM or any other encryption mode must be reviewed by your organization's Crypto Board. Reusing the same initialization vector (IV) with block ciphers in "streaming ciphers modes," such as CTR, may cause encrypted data to be revealed. All symmetric block ciphers must also be used with an appropriate initialization vector (IV). An appropriate IV is a cryptographically strong, random number and never a constant value.</p>

## Use approved asymmetric algorithms, key lengths, and padding

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	N/A
<b>Steps</b>	<p>The use of banned cryptographic algorithms introduces significant risk to product security and must be avoided. Products must use only those cryptographic algorithms and associated key lengths and padding that have been explicitly approved by your organization's Crypto Board.</p> <ul style="list-style-type: none"> <li>• <b>RSA-</b> may be used for encryption, key exchange and signature. RSA encryption must use only the OAEP or RSA-KEM padding modes. Existing code may use PKCS #1 v1.5 padding mode for compatibility only. Use of null padding is explicitly banned. Keys <math>\geq 2048</math> bits is required for new code. Existing code may support keys <math>&lt; 2048</math> bits only for backwards compatibility after a review by your organization's Crypto Board. Keys <math>&lt; 1024</math> bits may only be used for decrypting/verifying old data, and must be replaced if used for encryption or signing operations</li> <li>• <b>ECDSA-</b> may be used for signature only. ECDSA with <math>\geq 256</math>-bit keys is required for new code. ECDSA-based signatures must use one of the three NIST approved curves (P-256, P-384, or P521). Curves that have been thoroughly analyzed may be used only after a review with your organization's Crypto Board.</li> <li>• <b>ECDH-</b> may be used for key exchange only. ECDH with <math>\geq 256</math>-bit keys is required for new code. ECDH-based key exchange must use one of the three NIST approved curves (P-256, P-384, or P521). Curves that have been thoroughly analyzed may be used only after a review with your organization's Crypto Board.</li> <li>• <b>DSA-</b> may be acceptable after review and approval from your organization's Crypto Board. Contact your security advisor to schedule your organization's Crypto Board review. If your use of DSA is approved, note that you will need to prohibit use of keys less than 2048 bits in length. CNG supports 2048-bit and greater key lengths as of Windows 8.</li> <li>• <b>Diffie-Hellman-</b> may be used for session key management only. Key length <math>\geq 2048</math> bits is required for new code. Existing code may support key lengths <math>&lt; 2048</math> bits only for backwards compatibility after a review by your organization's Crypto Board. Keys <math>&lt; 1024</math> bits may not be used.</li> </ul>

## Use approved random number generators

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	N/A
<b>Steps</b>	<p>Products must use approved random number generators. Pseudorandom functions such as the C runtime function rand, the .NET Framework class System.Random, or system functions such as GetTickCount must, therefore, never be used in such code. Use of the dual elliptic curve random number generator (DUAL_EC_DRBG) algorithm is prohibited</p> <ul style="list-style-type: none"> <li>• <b>CNG-</b> BCryptGenRandom(use of the BCRYPT_USE_SYSTEM_PREFERRED_RNG flag recommended unless the caller might run at any IRQL greater than 0 [that is, PASSIVE_LEVEL])</li> <li>• <b>CAPI-</b> cryptGenRandom</li> <li>• <b>Win32/64-</b> RtlGenRandom (new implementations should use BCryptGenRandom or CryptGenRandom) * rand_s * SystemPrng (for kernel mode)</li> <li>• <b>.NET-</b> RNGCryptoServiceProvider or RNGCng</li> <li>• <b>Windows Store Apps-</b> Windows.Security.Cryptography.CryptographicBuffer.GenerateRandom or .GenerateRandomNumber</li> <li>• <b>Apple OS X (10.7+)/iOS(2.0+)-</b> int SecRandomCopyBytes (SecRandomRef random, size_t count, uint8_t bytes )</li> <li>• <b>Apple OS X (&lt; 10.7)-</b>* Use /dev/random to retrieve random numbers</li> <li>• <b>Java(including Google Android Java code)-</b> java.security.SecureRandom class. Note that for Android 4.3 (Jelly Bean), developers must follow the Android recommended workaround and update their applications to explicitly initialize the PRNG with entropy from /dev/urandom or /dev/random</li> </ul>

## Do not use symmetric stream ciphers

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Symmetric stream ciphers, such as RC4, must not be used. Instead of symmetric stream ciphers, products should use a block cipher, specifically AES with a key length of at least 128 bits.

## Use approved MAC/HMAC/keyed hash algorithms

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>Products must use only approved message authentication code (MAC) or hash-based message authentication code (HMAC) algorithms.</p> <p>A message authentication code (MAC) is a piece of information attached to a message that allows its recipient to verify both the authenticity of the sender and the integrity of the message using a secret key. The use of either a hash-based MAC (<a href="#">HMAC</a>) or <a href="#">block-cipher-based MAC</a> is permissible as long as all underlying hash or symmetric encryption algorithms are also approved for use; currently this includes the HMAC-SHA2 functions (HMAC-SHA256, HMAC-SHA384 and HMAC-SHA512) and the CMAC/OMAC1 and OMAC2 block cipher-based MACs (these are based on AES).</p> <p>Use of HMAC-SHA1 may be permissible for platform compatibility, but you will be required to file an exception to this procedure and undergo your organization's Crypto review. Truncation of HMACs to less than 128 bits is not permitted. Using customer methods to hash a key and data is not approved, and must undergo your organization's Crypto Board review prior to use.</p>

## Use only approved cryptographic hash functions

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A

TITLE	DETAILS
<b>Steps</b>	<p>Products must use the SHA-2 family of hash algorithms (SHA256, SHA384, and SHA512). If a shorter hash is needed, such as a 128-bit output length in order to fit a data structure designed with the shorter MD5 hash in mind, product teams may truncate one of the SHA2 hashes (typically SHA256). Note that SHA384 is a truncated version of SHA512. Truncation of cryptographic hashes for security purposes to less than 128 bits is not permitted. New code must not use the MD2, MD4, MD5, SHA-0, SHA-1, or RIPEMD hash algorithms. Hash collisions are computationally feasible for these algorithms, which effectively breaks them.</p> <p>Allowed .NET hash algorithms for managed crypto agility (in order of preference):</p> <ul style="list-style-type: none"> <li>• SHA512Cng (FIPS compliant)</li> <li>• SHA384Cng (FIPS compliant)</li> <li>• SHA256Cng (FIPS compliant)</li> <li>• SHA512Managed (non-FIPS-compliant) (use SHA512 as algorithm name in calls to HashAlgorithm.Create or CryptoConfig.CreateFromName)</li> <li>• SHA384Managed (non-FIPS-compliant) (use SHA384 as algorithm name in calls to HashAlgorithm.Create or CryptoConfig.CreateFromName)</li> <li>• SHA256Managed (non-FIPS-compliant) (use SHA256 as algorithm name in calls to HashAlgorithm.Create or CryptoConfig.CreateFromName)</li> <li>• SHA512CryptoServiceProvider (FIPS compliant)</li> <li>• SHA256CryptoServiceProvider (FIPS compliant)</li> <li>• SHA384CryptoServiceProvider (FIPS compliant)</li> </ul>

## Use strong encryption algorithms to encrypt data in the database

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Choosing an encryption algorithm</a>
<b>Steps</b>	Encryption algorithms define data transformations that cannot be easily reversed by unauthorized users. SQL Server allows administrators and developers to choose from among several algorithms, including DES, Triple DES, TRIPLE_DES_3KEY, RC2, RC4, 128-bit RC4, DESX, 128-bit AES, 192-bit AES, and 256-bit AES

## SSIS packages should be encrypted and digitally signed

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Identify the Source of Packages with Digital Signatures, Threat and Vulnerability Mitigation (Integration Services)</a>
<b>Steps</b>	The source of a package is the individual or organization that created the package. Running a package from an unknown or untrusted source might be risky. To prevent unauthorized tampering of SSIS packages, digital signatures should be used. Also, to ensure the confidentiality of the packages during storage/transit, SSIS packages have to be encrypted

## Add digital signature to critical database securables

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">ADD SIGNATURE (Transact-SQL)</a>
<b>Steps</b>	In cases where the integrity of a critical database securable has to be verified, digital signatures should be used. Database securables such as a stored procedure, function, assembly, or trigger can be digitally signed. Below is an example of when this can be useful: Let us say an ISV (Independent Software Vendor) has provided support to a software delivered to one of their customers. Before providing support, the ISV would want to ensure that a database securable in the software was not tampered either by mistake or by a malicious attempt. If the securable is digitally signed, the ISV can verify its digital signature and validate its integrity.

## Use SQL server EKM to protect encryption keys

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build

TITLE	DETAILS
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">SQL Server Extensible Key Management (EKM)</a> , <a href="#">Extensible Key Management Using Azure Key Vault (SQL Server)</a>
<b>Steps</b>	SQL Server Extensible Key Management enables the encryption keys that protect the database files to be stored in an off-box device such as a smartcard, USB device, or EKM/HSM module. This also enables data protection from database administrators (except members of the sysadmin group). Data can be encrypted by using encryption keys that only the database user has access to on the external EKM/HSM module.

Use AlwaysEncrypted feature if encryption keys should not be revealed to Database engine

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	SQL Azure, OnPrem
<b>Attributes</b>	SQL Version - V12, MsSQL2016
<b>References</b>	<a href="#">Always Encrypted (Database Engine)</a>
<b>Steps</b>	Always Encrypted is a feature designed to protect sensitive data, such as credit card numbers or national identification numbers (e.g. U.S. social security numbers), stored in Azure SQL Database or SQL Server databases. Always Encrypted allows clients to encrypt sensitive data inside client applications and never reveal the encryption keys to the Database Engine (SQL Database or SQL Server). As a result, Always Encrypted provides a separation between those who own the data (and can view it) and those who manage the data (but should have no access)

Store Cryptographic Keys securely on IoT Device

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic

TITLE	DETAILS
<b>Attributes</b>	Device OS - Windows IoT Core, Device Connectivity - Azure IoT device SDKs
<b>References</b>	<a href="#">TPM on Windows IoT Core</a> , <a href="#">Set up TPM on Windows IoT Core</a> , <a href="#">Azure IoT Device SDK TPM</a>
<b>Steps</b>	Symmetric or Certificate Private keys securely in a hardware protected storage like TPM or Smart Card chips. Windows 10 IoT Core supports the user of a TPM and there are several compatible TPMs that can be used: <a href="https://developer.microsoft.com/windows/iot/win10/tpm">https://developer.microsoft.com/windows/iot/win10/tpm</a> . It is recommended to use a Firmware or Discrete TPM. A Software TPM should only be used for development and testing purposes. Once a TPM is available and the keys are provisioned in it, the code that generates the token should be written without hard coding any sensitive information in it.

### Example

```

TpmDevice myDevice = new TpmDevice(0);
// Use logical device 0 on the TPM
string hubUri = myDevice.GetHostName();
string deviceId = myDevice.GetDeviceId();
string sasToken = myDevice.GetSASToken();

var deviceClient = DeviceClient.Create(hubUri, AuthenticationMethodFactory.
CreateAuthenticationWithToken(deviceId, sasToken), TransportType.Amqp);

```

As can be seen, the device primary key is not present in the code. Instead, it is stored in the TPM at slot 0. TPM device generates a short-lived SAS token that is then used to connect to the IoT Hub.

## Generate a random symmetric key of sufficient length for authentication to IoT Hub

TITLE	DETAILS
<b>Component</b>	IoT Cloud Gateway
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	Gateway choice - Azure IoT Hub
<b>References</b>	N/A
<b>Steps</b>	IoT Hub contains a device Identity Registry and while provisioning a device, automatically generates a random Symmetric key. It is recommended to use this feature of the Azure IoT Hub Identity Registry to generate the key used for authentication. IoT Hub also allows for a key to be specified while creating the device. If a key is generated outside of IoT Hub during device provisioning, it is recommended to create a random symmetric key or at least 256 bits.

Ensure a device management policy is in place that requires a use PIN and allows remote wiping

TITLE	DETAILS
<b>Component</b>	Dynamics CRM Mobile Client
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Ensure a device management policy is in place that requires a use PIN and allows remote wiping

Ensure a device management policy is in place that requires a PIN/password/auto lock and encrypts all data (e.g. Bitlocker)

TITLE	DETAILS
<b>Component</b>	Dynamics CRM Outlook Client
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Ensure a device management policy is in place that requires a PIN/password/auto lock and encrypts all data (e.g. Bitlocker)

Ensure that signing keys are rolled over when using Identity Server

TITLE	DETAILS
<b>Component</b>	Identity Server
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Identity Server - Keys, Signatures and Cryptography</a>

TITLE	DETAILS
<b>Steps</b>	Ensure that signing keys are rolled over when using Identity Server. The link in the references section explains how this should be planned without causing outages to applications relying on Identity Server.

## Ensure that cryptographically strong client ID, client secret are used in Identity Server

TITLE	DETAILS
<b>Component</b>	Identity Server
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>Ensure that cryptographically strong client ID, client secret are used in Identity Server. The following guidelines should be used while generating a client ID and secret:</p> <ul style="list-style-type: none"> <li>• Generate a random GUID as the client ID</li> <li>• Generate a cryptographically random 256-bit key as the secret</li> </ul>

# Security Frame: Exception Management | Mitigations

8/25/2017 • 7 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
<b>WCF</b>	<ul style="list-style-type: none"><li>• <a href="#">WCF- Do not include serviceDebug node in configuration file</a></li><li>• <a href="#">WCF- Do not include serviceMetadata node in configuration file</a></li></ul>
<b>Web API</b>	<ul style="list-style-type: none"><li>• <a href="#">Ensure that proper exception handling is done in ASP.NET Web API</a></li></ul>
<b>Web Application</b>	<ul style="list-style-type: none"><li>• <a href="#">Do not expose security details in error messages</a></li><li>• <a href="#">Implement Default error handling page</a></li><li>• <a href="#">Set Deployment Method to Retail in IIS</a></li><li>• <a href="#">Exceptions should fail safely</a></li></ul>

## WCF- Do not include serviceDebug node in configuration file

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, .NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a>
<b>Steps</b>	Windows Communication Framework (WCF) services can be configured to expose debugging information. Debug information should not be used in production environments. The <code>&lt;serviceDebug&gt;</code> tag defines whether the debug information feature is enabled for a WCF service. If the attribute <code>includeExceptionDetailInFaults</code> is set to true, exception information from the application will be returned to clients. Attackers can leverage the additional information they gain from debugging output to mount attacks targeted on the framework, database, or other resources used by the application.

### Example

The following configuration file includes the `<serviceDebug>` tag:

```

<configuration>
<system.serviceModel>
<behaviors>
<serviceBehaviors>
<behavior name=""MyServiceBehavior"">
<serviceDebug includeExceptionDetailInFaults=""True"" httpHelpPageEnabled=""True""/>
...

```

Disable debugging information in the service. This can be accomplished by removing the `<serviceDebug>` tag from your application's configuration file.

## WCF- Do not include serviceMetadata node in configuration file

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	Generic, .NET Framework 3
<b>References</b>	<a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a>
<b>Steps</b>	Publicly exposing information about a service can provide attackers with valuable insight into how they might exploit the service. The <code>&lt;serviceMetadata&gt;</code> tag enables the metadata publishing feature. Service metadata could contain sensitive information that should not be publicly accessible. At a minimum, only allow trusted users to access the metadata and ensure that unnecessary information is not exposed. Better yet, entirely disable the ability to publish metadata. A safe WCF configuration will not contain the <code>&lt;serviceMetadata&gt;</code> tag.

## Ensure that proper exception handling is done in ASP.NET Web API

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC 5, MVC 6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Exception Handling in ASP.NET Web API</a> , <a href="#">Model Validation in ASP.NET Web API</a>
<b>Steps</b>	By default, most uncaught exceptions in ASP.NET Web API are translated into an HTTP response with status code <code>500, Internal Server Error</code>

## Example

To control the status code returned by the API, `HttpResponseException` can be used as shown below:

```
public Product GetProduct(int id)
{
 Product item = repository.Get(id);
 if (item == null)
 {
 throw new HttpResponseException(HttpStatusCode.NotFound);
 }
 return item;
}
```

## Example

For further control on the exception response, the `HttpResponseMessage` class can be used as shown below:

```
public Product GetProduct(int id)
{
 Product item = repository.Get(id);
 if (item == null)
 {
 var resp = new HttpResponseMessage(HttpStatusCode.NotFound)
 {
 Content = new StringContent(string.Format("No product with ID = {0}", id)),
 ReasonPhrase = "Product ID Not Found"
 }
 throw new HttpResponseException(resp);
 }
 return item;
}
```

To catch unhandled exceptions that are not of the type `HttpResponseException`, Exception Filters can be used.

Exception filters implement the `System.Web.Http.Filters.IExceptionFilter` interface. The simplest way to write an exception filter is to derive from the `System.Web.Http.Filters.ExceptionFilterAttribute` class and override the `OnException` method.

## Example

Here is a filter that converts `NotImplementedException` exceptions into HTTP status code `501, Not Implemented`:

```
namespace ProductStore.Filters
{
 using System;
 using System.Net;
 using System.Net.Http;
 using System.Web.Http.Filters;

 public class NotImplExceptionFilterAttribute : ExceptionFilterAttribute
 {
 public override void OnException(HttpActionExecutedContext context)
 {
 if (context.Exception is NotImplementedException)
 {
 context.Response = new HttpResponseMessage(HttpStatusCode.NotImplemented);
 }
 }
 }
}
```

There are several ways to register a Web API exception filter:

- By action
- By controller
- Globally

### Example

To apply the filter to a specific action, add the filter as an attribute to the action:

```
public class ProductsController : ApiController
{
 [NotImplExceptionFilter]
 public Contact GetContact(int id)
 {
 throw new NotImplementedException("This method is not implemented");
 }
}
```

### Example

To apply the filter to all of the actions on a `controller`, add the filter as an attribute to the `controller` class:

```
[NotImplExceptionFilter]
public class ProductsController : ApiController
{
 // ...
}
```

### Example

To apply the filter globally to all Web API controllers, add an instance of the filter to the `GlobalConfiguration.Configuration.Filters` collection. Exception filters in this collection apply to any Web API controller action.

```
GlobalConfiguration.Configuration.Filters.Add(
 new ProductStore.NotImplExceptionFilterAttribute());
```

### Example

For model validation, the model state can be passed to `CreateErrorResponse` method as shown below:

```
public HttpResponseMessage PostProduct(Product item)
{
 if (!ModelState.IsValid)
 {
 return Request.CreateErrorResponse(HttpStatusCode.BadRequest, ModelState);
 }
 // Implementation not shown...
}
```

Check the links in the references section for additional details about exceptional handling and model validation in ASP.NET Web API

## Do not expose security details in error messages

TITLE	DETAILS
Component	Web Application

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>Generic error messages are provided directly to the user without including sensitive application data. Examples of sensitive data include:</p> <ul style="list-style-type: none"> <li>• Server names</li> <li>• Connection strings</li> <li>• Usernames</li> <li>• Passwords</li> <li>• SQL procedures</li> <li>• Details of dynamic SQL failures</li> <li>• Stack trace and lines of code</li> <li>• Variables stored in memory</li> <li>• Drive and folder locations</li> <li>• Application install points</li> <li>• Host configuration settings</li> <li>• Other internal application details</li> </ul> <p>Trapping all errors within an application and providing generic error messages, as well as enabling custom errors within IIS will help prevent information disclosure. SQL Server database and .NET Exception handling, among other error handling architectures, are especially verbose and extremely useful to a malicious user profiling your application. Do not directly display the contents of a class derived from the .NET Exception class, and ensure that you have proper exception handling so that an unexpected exception isn't inadvertently raised directly to the user.</p> <ul style="list-style-type: none"> <li>• Provide generic error messages directly to the user that abstract away specific details found directly in the exception/error message</li> <li>• Do not display the contents of a .NET exception class directly to the user</li> <li>• Trap all error messages and if appropriate inform the user via a generic error message sent to the application client</li> <li>• Do not expose the contents of the Exception class directly to the user, especially the return value from <code>.ToString()</code>, or the values of the Message or StackTrace properties. Securely log this information and display a more innocuous message to the user</li> </ul>

## Implement Default error handling page

TITLE	DETAILS
<b>Component</b>	Web Application

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Edit ASP.NET Error Pages Settings Dialog Box</a>
<b>Steps</b>	<p>When an ASP.NET application fails and causes an HTTP/1.x 500 Internal Server Error, or a feature configuration (such as Request Filtering) prevents a page from being displayed, an error message will be generated. Administrators can choose whether or not the application should display a friendly message to the client, detailed error message to the client, or detailed error message to localhost only. The tag in the web.config has three modes:</p> <ul style="list-style-type: none"> <li>• <b>On:</b> Specifies that custom errors are enabled. If no defaultRedirect attribute is specified, users see a generic error. The custom errors are shown to the remote clients and to the local host</li> <li>• <b>Off:</b> Specifies that custom errors are disabled. The detailed ASP.NET errors are shown to the remote clients and to the local host</li> <li>• <b>RemoteOnly:</b> Specifies that custom errors are shown only to the remote clients, and that ASP.NET errors are shown to the local host. This is the default value</li> </ul> <p>Open the <code>web.config</code> file for the application/site and ensure that the tag has either  <code>&lt;customErrors mode="RemoteOnly" /&gt;</code> or  <code>&lt;customErrors mode="On" /&gt;</code> defined.</p>

## Set Deployment Method to Retail in IIS

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">deployment Element (ASP.NET Settings Schema)</a>

TITLE	DETAILS
<b>Steps</b>	<p>The <code>&lt;deployment retail&gt;</code> switch is intended for use by production IIS servers. This switch is used to help applications run with the best possible performance and least possible security information leakages by disabling the application's ability to generate trace output on a page, disabling the ability to display detailed error messages to end users, and disabling the debug switch.</p> <p>Often times, switches and options that are developer-focused, such as failed request tracing and debugging, are enabled during active development. It is recommended that the deployment method on any production server be set to retail. Open the machine.config file and ensure that <code>&lt;deployment retail="true" /&gt;</code> remains set to true.</p>

## Exceptions should fail safely

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Fail securely</a>
<b>Steps</b>	Application should fail safely. Any method that returns a Boolean value, based on which certain decision is made, should have exception block carefully created. There are lot of logical errors due to which security issues creep in, when the exception block is written carelessly.

### Example

```

public static bool ValidateDomain(string pathToValidate, Uri currentUrl)
{
 try
 {
 if (!string.IsNullOrWhiteSpace(pathToValidate))
 {
 var domain = RetrieveDomain(currentUrl);
 var replyPath = new Uri(pathToValidate);
 var replyDomain = RetrieveDomain(replyPath);

 if (string.Compare(domain, replyDomain, StringComparison.OrdinalIgnoreCase) != 0)
 {
 //// Adding additional check to enable CMS urls if they are not hosted on same domain.
 if (!string.IsNullOrWhiteSpace(Utilities.CmsBase))
 {
 var cmsDomain = RetrieveDomain(new Uri(Utilities.Base.Trim()));
 if (string.Compare(cmsDomain, replyDomain, StringComparison.OrdinalIgnoreCase) != 0)
 {
 return false;
 }
 else
 {
 return true;
 }
 }
 }

 return false;
 }
 }

 return true;
}
catch (UriFormatException ex)
{
 LogHelper.LogError("Utilities:ValidateDomain", ex);
 return true;
}
}

```

The above method will always return True, if some exception happens. If the end user provides a malformed URL, that the browser respects, but the `Uri()` constructor doesn't, this will throw an exception, and the victim will be taken to the valid but malformed URL.

# Security Frame: Input Validation | Mitigations

8/22/2017 • 31 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
<b>Web Application</b>	<ul style="list-style-type: none"><li>• Disable XSLT scripting for all transforms using untrusted style sheets</li><li>• Ensure that each page that could contain user controllable content opts out of automatic MIME sniffing</li><li>• Harden or Disable XML Entity Resolution</li><li>• Applications utilizing http.sys perform URL canonicalization verification</li><li>• Ensure appropriate controls are in place when accepting files from users</li><li>• Ensure that type-safe parameters are used in Web Application for data access</li><li>• Use separate model binding classes or binding filter lists to prevent MVC mass assignment vulnerability</li><li>• Encode untrusted web output prior to rendering</li><li>• Perform input validation and filtering on all string type Model properties</li><li>• Sanitization should be applied on form fields that accept all characters, e.g. rich text editor</li><li>• Do not assign DOM elements to sinks that do not have inbuilt encoding</li><li>• Validate all redirects within the application are closed or done safely</li><li>• Implement input validation on all string type parameters accepted by Controller methods</li><li>• Set upper limit timeout for regular expression processing to prevent DoS due to bad regular expressions</li><li>• Avoid using Html.Raw in Razor views</li></ul>
<b>Database</b>	<ul style="list-style-type: none"><li>• Do not use dynamic queries in stored procedures</li></ul>
<b>Web API</b>	<ul style="list-style-type: none"><li>• Ensure that model validation is done on Web API methods</li><li>• Implement input validation on all string type parameters accepted by Web API methods</li><li>• Ensure that type-safe parameters are used in Web API for data access</li></ul>
<b>Azure Document DB</b>	<ul style="list-style-type: none"><li>• Use parametrized SQL queries for DocumentDB</li></ul>
<b>WCF</b>	<ul style="list-style-type: none"><li>• WCF Input validation through Schema binding</li><li>• WCF- Input validation through Parameter Inspectors</li></ul>

Disable XSLT scripting for all transforms using untrusted style sheets

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">XSLT Security</a> , <a href="#">XsltSettings.EnableScript Property</a>
<b>Steps</b>	XSLT supports scripting inside style sheets using the <code>&lt;msxml:script&gt;</code> element. This allows custom functions to be used in an XSLT transformation. The script is executed under the context of the process performing the transform. XSLT script must be disabled when in an untrusted environment to prevent execution of untrusted code. <i>If using .NET:</i> XSLT scripting is disabled by default; however, you must ensure that it has not been explicitly enabled through the <code>XsltSettings.EnableScript</code> property.

## Example

```
XsltSettings settings = new XsltSettings();
settings.EnableScript = true; // WRONG: THIS SHOULD BE SET TO false
```

## Example

If you are using MSXML 6.0, XSLT scripting is disabled by default; however, you must ensure that it has not been explicitly enabled through the XML DOM object property `AllowXsltScript`.

```
doc.setProperty("AllowXsltScript", true); // WRONG: THIS SHOULD BE SET TO false
```

## Example

If you are using MSXML 5 or below, XSLT scripting is enabled by default and you must explicitly disable it. Set the XML DOM object property `AllowXsltScript` to false.

```
doc.setProperty("AllowXsltScript", false); // CORRECT. Setting to false disables XSLT scripting.
```

Ensure that each page that could contain user controllable content opts out of automatic MIME sniffing

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	IE8 Security Part V - Comprehensive Protection
<b>Steps</b>	<p>For each page that could contain user controllable content, you must use the HTTP Header <code>X-Content-Type-Options:nosniff</code>. To comply with this requirement, you can either set the required header page by page for only those pages that might contain user-controllable content, or you can set it globally for all pages in the application.</p> <p>Each type of file delivered from a web server has an associated <a href="#">MIME type</a> (also called a <i>content-type</i>) that describes the nature of the content (that is, image, text, application, etc.)</p> <p>The X-Content-Type-Options header is an HTTP header that allows developers to specify that their content should not be MIME-sniffed. This header is designed to mitigate MIME-Sniffing attacks. Support for this header was added in Internet Explorer 8 (IE8)</p> <p>Only users of Internet Explorer 8 (IE8) will benefit from X-Content-Type-Options. Previous versions of Internet Explorer do not currently respect the X-Content-Type-Options header</p> <p>Internet Explorer 8 (and later) are the only major browsers to implement a MIME-sniffing opt-out feature. If and when other major browsers (Firefox, Safari, Chrome) implement similar features, this recommendation will be updated to include syntax for those browsers as well</p>

## Example

To enable the required header globally for all pages in the application, you can do one of the following:

- Add the header in the web.config file if the application is hosted by Internet Information Services (IIS) 7

```
<system.webServer>
 <httpProtocol>
 <customHeaders>
 <add name=""X-Content-Type-Options"" value=""nosniff""/>
 </customHeaders>
 </httpProtocol>
</system.webServer>
```

- Add the header through the global Application\_BeginRequest

```
void Application_BeginRequest(object sender, EventArgs e)
{
 this.Response.Headers["X-Content-Type-Options"] = "nosniff";
}
```

- Implement custom HTTP module

```

public class XContentTypeOptionsModule : IHttpModule
{
 #region IHttpModule Members
 public void Dispose()
 {

 }
 public void Init(HttpApplication context)
 {
 context.PreSendRequestHeaders += new EventHandler(context_PreSendRequestHeaders);
 }
 #endregion
 void context_PreSendRequestHeaders(object sender, EventArgs e)
 {
 HttpApplication application = sender as HttpApplication;
 if (application == null)
 return;
 if (application.Response.Headers["X-Content-Type-Options"] != null)
 return;
 application.Response.Headers.Add("X-Content-Type-Options", "nosniff");
 }
}

```

- You can enable the required header only for specific pages by adding it to individual responses:

```
this.Response.Headers["X-Content-Type-Options"] = "nosniff";
```

## Harden or Disable XML Entity Resolution

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">XML Entity Expansion</a> , <a href="#">XML Denial of Service Attacks and Defenses</a> , <a href="#">MSXML Security Overview</a> , <a href="#">Best Practices for Securing MSXML Code</a> , <a href="#">NSXMLParserDelegate Protocol Reference</a> , <a href="#">Resolving External References</a>

TITLE	DETAILS
<b>Steps</b>	<p>Although it is not widely used, there is a feature of XML that allows the XML parser to expand macro entities with values defined either within the document itself or from external sources. For example, the document might define an entity "companyname" with the value "Microsoft," so that every time the text "&amp;companyname;" appears in the document, it is automatically replaced with the text Microsoft. Or, the document might define an entity "MSFTStock" that references an external web service to fetch the current value of Microsoft stock.</p> <p>Then any time "&amp;MSFTStock;" appears in the document, it is automatically replaced with the current stock price. However, this functionality can be abused to create denial of service (DoS) conditions. An attacker can nest multiple entities to create an exponential expansion XML bomb that consumes all available memory on the system.</p> <p>Alternatively, he can create an external reference that streams back an infinite amount of data or that simply hangs the thread. As a result, all teams must disable internal and/or external XML entity resolution entirely if their application does not use it, or manually limit the amount of memory and time that the application can consume for entity resolution if this functionality is absolutely necessary. If entity resolution is not required by your application, then disable it.</p>

## Example

For .NET Framework code, you can use the following approaches:

```

XmlTextReader reader = new XmlTextReader(stream);
reader.ProhibitDtd = true;

XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = true;
XmlReader reader = XmlReader.Create(stream, settings);

// for .NET 4
XmlReaderSettings settings = new XmlReaderSettings();
settings.DtdProcessing = DtdProcessing.Prohibit;
XmlReader reader = XmlReader.Create(stream, settings);

```

Note that the default value of `ProhibitDtd` in `XmlReaderSettings` is true, but in `XmlTextReader` it is false. If you are using `XmlReaderSettings`, you do not need to set `ProhibitDtd` to true explicitly, but it is recommended for safety sake that you do. Also note that the  `XmlDocument` class allows entity resolution by default.

## Example

To disable entity resolution for  `XmlDocument`s, use the  `XmlDocument.Load(XmlReader)` overload of the `Load` method and set the appropriate properties in the `XmlReader` argument to disable resolution, as illustrated in the following code:

```

XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = true;
XmlReader reader = XmlReader.Create(stream, settings);
 XmlDocument doc = new XmlDocument();
doc.Load(reader);

```

## Example

If disabling entity resolution is not possible for your application, set the `XmlReaderSettings.MaxCharactersFromEntities` property to a reasonable value according to your application's needs. This will limit the impact of potential exponential expansion DoS attacks. The following code provides an example of this approach:

```
XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = false;
settings.MaxCharactersFromEntities = 1000;
XmlReader reader = XmlReader.Create(stream, settings);
```

## Example

If you need to resolve inline entities but do not need to resolve external entities, set the `XmlReaderSettings.XmlResolver` property to null. For example:

```
XmlReaderSettings settings = new XmlReaderSettings();
settings.ProhibitDtd = false;
settings.MaxCharactersFromEntities = 1000;
settings.XmlResolver = null;
XmlReader reader = XmlReader.Create(stream, settings);
```

Note that in MSXML6, `ProhibitDTD` is set to true (disabling DTD processing) by default. For Apple OSX/iOS code, there are two XML parsers you can use: NSXMLParser and libXML2.

## Applications utilizing http.sys perform URL canonicalization verification

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A

TITLE	DETAILS
<b>Steps</b>	<p>Any application that uses http.sys should follow these guidelines:</p> <ul style="list-style-type: none"> <li>• Limit the URL length to no more than 16,384 characters (ASCII or Unicode). This is the absolute maximum URL length based on the default Internet Information Services (IIS) 6 setting. Websites should strive for a length shorter than this if possible</li> <li>• Use the standard .NET Framework file I/O classes (such as FileStream) as these will take advantage of the canonicalization rules in the .NET FX</li> <li>• Explicitly build an allow-list of known filenames</li> <li>• Explicitly reject known filetypes you will not serve UrlScan rejects: exe, bat, cmd, com, htw, ida, idq, htr, idc, shtm[], stm, printer, ini, pol, dat files</li> <li>• Catch the following exceptions: <ul style="list-style-type: none"> <li>◦ System.ArgumentException (for device names)</li> <li>◦ System.NotSupportedException (for data streams)</li> <li>◦ System.IO.FileNotFoundException (for invalid escaped filenames)</li> <li>◦ System.IO.DirectoryNotFoundException (for invalid escaped dirs)</li> </ul> </li> <li>• <i>Do not</i> call out to Win32 file I/O APIs. On an invalid URL gracefully return a 400 error to the user, and log the real error.</li> </ul>

## Ensure appropriate controls are in place when accepting files from users

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Unrestricted File Upload</a> , <a href="#">File Signature Table</a>

TITLE	DETAILS
<b>Steps</b>	<p>Uploaded files represent a significant risk to applications. The first step in many attacks is to get some code to the system to be attacked. Then the attack only needs to find a way to get the code executed. Using a file upload helps the attacker accomplish the first step. The consequences of unrestricted file upload can vary, including complete system takeover, an overloaded file system or database, forwarding attacks to back-end systems, and simple defacement.</p> <p>It depends on what the application does with the uploaded file and especially where it is stored. Server side validation of file uploads is missing. Following security controls should be implemented for File Upload functionality:</p> <ul style="list-style-type: none"> <li>• File Extension check (only a valid set of allowed file type should be accepted)</li> <li>• Maximum file size limit</li> <li>• File should not be uploaded to webroot; the location should be a directory on non-system drive</li> <li>• Naming convention should be followed, such that the uploaded file name have some randomness, so as to prevent file overwrites</li> <li>• Files should be scanned for anti-virus before writing to the disk</li> <li>• Ensure that the file name and any other metadata (e.g., file path) are validated for malicious characters</li> <li>• File format signature should be checked, to prevent a user from uploading a masqueraded file (e.g., uploading an exe file by changing extension to txt)</li> </ul>

## Example

For the last point regarding file format signature validation, refer to the class below for details:

```

private static Dictionary<string, List<byte[]>> fileSignature = new Dictionary<string, List<byte[]>>
{
 { ".DOC", new List<byte[]> { new byte[] { 0xD0, 0xCF, 0x11, 0xE0, 0xA1, 0xB1, 0x1A, 0xE1 } } },
 { ".DOCX", new List<byte[]> { new byte[] { 0x50, 0x4B, 0x03, 0x04 } } },
 { ".PDF", new List<byte[]> { new byte[] { 0x25, 0x50, 0x44, 0x46 } } },
 { ".ZIP", new List<byte[]>
 {
 new byte[] { 0x50, 0x4B, 0x03, 0x04 },
 new byte[] { 0x50, 0x4B, 0x4C, 0x49, 0x54, 0x55 },
 new byte[] { 0x50, 0x4B, 0x53, 0x70, 0x58 },
 new byte[] { 0x50, 0x4B, 0x05, 0x06 },
 new byte[] { 0x50, 0x4B, 0x07, 0x08 },
 new byte[] { 0x57, 0x69, 0x6E, 0x5A, 0x69, 0x70 }
 } },
 { ".PNG", new List<byte[]> { new byte[] { 0x89, 0x50, 0x4E, 0x47, 0x0D, 0x0A, 0x1A, 0x0A } } },
 { ".JPG", new List<byte[]>
 {
 new byte[] { 0xFF, 0xD8, 0xFF, 0xE0 },
 new byte[] { 0xFF, 0xD8, 0xFF, 0xE1 },
 new byte[] { 0xFF, 0xD8, 0xFF, 0xE8 }
 } },
 { ".JPEG" -> new List<byte[]>{ }
}

```

```

 {
 new byte[] { 0xFF, 0xD8, 0xFF, 0xE0 },
 new byte[] { 0xFF, 0xD8, 0xFF, 0xE2 },
 new byte[] { 0xFF, 0xD8, 0xFF, 0xE3 }
 }
 },
 { ".XLS", new List<byte[]>
 {
 new byte[] { 0xD0, 0xCF, 0x11, 0xE0, 0xA1, 0xB1, 0x1A, 0xE1 },
 new byte[] { 0x09, 0x08, 0x10, 0x00, 0x00, 0x06, 0x05, 0x00 },
 new byte[] { 0xFD, 0xFF, 0xFF, 0xFF }
 }
 },
 { ".XLSX", new List<byte[]> { new byte[] { 0x50, 0x4B, 0x03, 0x04 } } },
 { ".GIF", new List<byte[]> { new byte[] { 0x47, 0x49, 0x46, 0x38 } } }
};

public static bool IsValidFileExtension(string fileName, byte[] fileData, byte[] allowedChars)
{
 if (string.IsNullOrEmpty(fileName) || fileData == null || fileData.Length == 0)
 {
 return false;
 }

 bool flag = false;
 string ext = Path.GetExtension(fileName);
 if (string.IsNullOrEmpty(ext))
 {
 return false;
 }

 ext = ext.ToUpperInvariant();

 if (ext.Equals(".TXT") || ext.Equals(".CSV") || ext.Equals(".PRN"))
 {
 foreach (byte b in fileData)
 {
 if (b > 0x7F)
 {
 if (allowedChars != null)
 {
 if (!allowedChars.Contains(b))
 {
 return false;
 }
 }
 else
 {
 return false;
 }
 }
 }
 }

 return true;
}

if (!fileSignature.ContainsKey(ext))
{
 return true;
}

List<byte[]> sig = fileSignature[ext];
foreach (byte[] b in sig)
{
 var curFileSig = new byte[b.Length];
 Array.Copy(fileData, curFileSig, b.Length);
 if (curFileSig.SequenceEqual(b))
 {
 flag = true;
 }
}
}

```

```

 flag = true;
 break;
 }

 return flag;
}

```

## Ensure that type-safe parameters are used in Web Application for data access

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>If you use the Parameters collection, SQL treats the input as a literal value rather than as executable code. The Parameters collection can be used to enforce type and length constraints on input data. Values outside of the range trigger an exception. If type-safe SQL parameters are not used, attackers might be able to execute injection attacks that are embedded in the unfiltered input.</p> <p>Use type safe parameters when constructing SQL queries to avoid possible SQL injection attacks that can occur with unfiltered input. You can use type safe parameters with stored procedures and with dynamic SQL statements. Parameters are treated as literal values by the database and not as executable code. Parameters are also checked for type and length.</p>

### Example

The following code shows how to use type safe parameters with the SqlParameterCollection when calling a stored procedure.

```

using System.Data;
using System.Data.SqlClient;

using (SqlConnection connection = new SqlConnection(connectionString))
{
 DataSet userDataset = new DataSet();
 SqlDataAdapter myCommand = new SqlDataAdapter("LoginStoredProcedure", connection);
 myCommand.SelectCommand.CommandType = CommandType.StoredProcedure;
 myCommand.SelectCommand.Parameters.Add("@au_id", SqlDbType.VarChar, 11);
 myCommand.SelectCommand.Parameters["@au_id"].Value = SSN.Text;
 myCommand.Fill(userDataset);
}

```

In the preceding code example, the input value cannot be longer than 11 characters. If the data does not conform to

the type or length defined by the parameter, the SqlParameter class throws an exception.

## Use separate model binding classes or binding filter lists to prevent MVC mass assignment vulnerability

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC5, MVC6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Metadata Attributes, Public Key Security Vulnerability And Mitigation, Complete Guide to Mass Assignment in ASP.NET MVC, Getting Started with EF using MVC</a>
<b>Steps</b>	<ul style="list-style-type: none"><li><b>When should I look for over-posting vulnerabilities?</b> - Over-posting vulnerabilities can occur any place you bind model classes from user input. Frameworks like MVC can represent user data in custom .NET classes, including Plain Old CLR Objects (POCOs). MVC automatically populates these model classes with data from the request, providing a convenient representation for dealing with user input. When these classes include properties that should not be set by the user, the application can be vulnerable to over-posting attacks, which allow user control of data that the application never intended. Like MVC model binding, database access technologies such as object/relational mappers like Entity Framework often also support using POCO objects to represent database data. These data model classes provide the same convenience in dealing with database data as MVC does in dealing with user input. Because both MVC and the database support similar models, like POCO objects, it seems easy to reuse the same classes for both purposes. This practice fails to preserve separation of concerns, and it is one common area where unintended properties are exposed to model binding, enabling over-posting attacks.</li><li><b>Why shouldn't I use my unfiltered database model classes as parameters to my MVC actions?</b> - Because MVC model binding will bind anything in that class. Even if the data does not appear in your view, a malicious user can send an HTTP request with this data included, and MVC will gladly bind it because your action says that database class is the shape of data it should accept for user input.</li><li><b>Why should I care about the shape used for model binding?</b> - Using ASP.NET MVC model binding with overly broad models exposes an application to over-posting attacks. Over-posting could enable attackers to change application data beyond what the developer intended, such as overriding the price for an item or the security privileges for an account. Applications should use action-specific binding models (or specific allowed property filter lists) to provide an explicit</li></ul>

TITLE	DETAILS
	<p>contract for what untrusted input to allow via model binding.</p> <ul style="list-style-type: none"> <li><b>Is having separate binding models just duplicating code?</b> - No, it is a matter of separation of concerns. If you reuse database models in action methods, you are saying any property (or sub-property) in that class can be set by the user in an HTTP request. If that is not what you want MVC to do, you need a filter list or a separate class shape to show MVC what data can come from user input instead.</li> <li><b>If I have separate binding models for user input, do I have to duplicate all my data annotation attributes?</b> - Not necessarily. You can use MetadataTypeAttribute on the database model class to link to the metadata on a model binding class. Just note that the type referenced by the MetadataTypeAttribute must be a subset of the referencing type (it can have fewer properties, but not more).</li> <li><b>Moving data back and forth between user input models and database models is tedious. Can I just copy over all properties using reflection?</b> - Yes. The only properties that appear in the binding models are the ones you have determined to be safe for user input. There is no security reason that prevents using reflection to copy over all properties that exist in common between these two models.</li> <li><b>What about [Bind(Exclude = "")]. Can I use that instead of having separate binding models?</b> - This approach is not recommended. Using [Bind(Exclude = "")] means that any new property is bindable by default. When a new property is added, there is an extra step to remember to keep things secure, rather than having the design be secure by default. Depending on the developer checking this list every time a property is added is risky.</li> <li><b>Is [Bind(Include = "")] useful for Edit operations?</b> - No. [Bind(Include = "")] is only suitable for INSERT-style operations (adding new data). For UPDATE-style operations (revising existing data), use another approach, like having separate binding models or passing an explicit list of allowed properties to UpdateModel or TryUpdateModel. Adding a [Bind(Include = "")] attribute on an Edit operation means that MVC will create an object instance and set only the listed properties, leaving all others at their default values. When the data is persisted, it will entirely replace the existing entity, resetting the values for any omitted properties to their defaults. For example, if IsAdmin was omitted from a [Bind(Include = "")] attribute on an Edit operation, any user whose name was edited via this action would be reset to IsAdmin = false (any edited user would lose administrator status). If you want to prevent updates to certain properties, use one of the other approaches above. Note that some versions of MVC tooling generate controller classes with [Bind(Include = "")] on Edit actions and imply that removing a property from that list will prevent over-posting attacks. However, as described above, that approach does not work as intended and instead will reset any data in the omitted properties to their default values.</li> <li><b>For Create operations, are there any caveats using [Bind(Include = "")] rather than separate</b></li> </ul>

TITLE	DETAILS
	<p><b>binding models?</b> - Yes. First this approach does not work for Edit scenarios, requiring maintaining two separate approaches for mitigating all over-posting vulnerabilities. Second, separate binding models enforce separation of concerns between the shape used for user input and the shape used for persistence, something [Bind(Include = "")] does not do. Third, note that [Bind(Include = "")] can only handle top-level properties; you cannot allow only portions of sub-properties (such as "Details.Name") in the attribute. Finally, and perhaps most importantly, using [Bind(Include = "")] adds an extra step that must be remembered any time the class is used for model binding. If a new action method binds to the data class directly and forgets to include a [Bind(Include = "")] attribute, it can be vulnerable to over-posting attacks, so the [Bind(Include = "")] approach is somewhat less secure by default. If you use [Bind(Include = "")], take care always to remember to specify it every time your data classes appear as action method parameters.</p> <ul style="list-style-type: none"> <li>• <b>For Create operations, what about putting the [Bind(Include = "")] attribute on the model class itself? Does not this approach avoid the need to remember putting the attribute on every action method?</b> - This approach works in some cases. Using [Bind(Include = "")] on the model type itself (rather than on action parameters using this class), does avoid the need to remember to include the [Bind(Include = "")] attribute on every action method. Using the attribute directly on the class effectively creates a separate surface area of this class for model binding purposes. However, this approach only allows for one model binding shape per model class. If one action method needs to allow model binding of a field (for example, an administrator-only action that updates user roles) and other actions need to prevent model binding of this field, this approach will not work. Each class can only have one model binding shape; if different actions need different model binding shapes, they need to represent these separate shapes using either separate model binding classes or separate [Bind(Include = "")] attributes on the action methods.</li> <li>• <b>What are binding models? Are they the same thing as view models?</b> - These are two related concepts. The term binding model refers to a model class used in an action parameter list (the shape passed from MVC model binding to the action method). The term view model refers to a model class passed from an action method to a view. Using a view-specific model is a common approach for passing data from an action method to a view. Often, this shape is also suitable for model binding, and the term view model can be used to refer the same model used in both places. To be precise, this procedure talks specifically about binding models, focusing on the shape passed to the action, which is what matters for mass assignment purposes.</li> </ul>

## Encode untrusted web output prior to rendering

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, Web Forms, MVC5, MVC6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">How to prevent Cross-site scripting in ASP.NET</a> , <a href="#">Cross-site Scripting, XSS (Cross Site Scripting) Prevention Cheat Sheet</a>
<b>Steps</b>	Cross-site scripting (commonly abbreviated as XSS) is an attack vector for online services or any application/component that consumes input from the web. XSS vulnerabilities may allow an attacker to execute script on another user's machine through a vulnerable web application. Malicious scripts can be used to steal cookies and otherwise tamper with a victim's machine through JavaScript. XSS is prevented by validating user input, ensuring it is well formed and encoding before it is rendered in a web page. Input validation and output encoding can be done by using Web Protection Library. For Managed code (C#, VB.net, etc.), use one or more appropriate encoding methods from the Web Protection (Anti-XSS) Library, depending on the context where the user input gets manifested:

### Example

```
* Encoder.HtmlEncode
* Encoder.HtmlAttributeEncode
* Encoder.JavaScriptEncode
* Encoder.UrlEncode
* Encoder.VisualBasicScriptEncode
* Encoder.XmlEncode
* Encoder.XmlAttributeEncode
* Encoder.CssEncode
* Encoder.LdapEncode
```

## Perform input validation and filtering on all string type Model properties

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, MVC5, MVC6
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	<a href="#">Adding Validation, Validating Model Data in an MVC Application, Guiding Principles For Your ASP.NET MVC Applications</a>
<b>Steps</b>	<p>All the input parameters must be validated before they are used in the application to ensure that the application is safeguarded against malicious user inputs. Validate the input values using regular expression validations on server side with a whitelist validation strategy. Unsanitized user inputs / parameters passed to the methods can cause code injection vulnerabilities.</p> <p>For web applications, entry points can also include form fields, QueryStrings, cookies, HTTP headers, and web service parameters.</p> <p>The following input validation checks must be performed upon model binding:</p> <ul style="list-style-type: none"> <li>• The model properties should be annotated with RegularExpression annotation, for accepting allowed characters and maximum permissible length</li> <li>• The controller methods should perform ModelState validity</li> </ul>

Sanitization should be applied on form fields that accept all characters, e.g, rich text editor

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Encode Unsafe Input, HTML Sanitizer</a>

TITLE	DETAILS
<b>Steps</b>	<p>Identify all static markup tags that you want to use. A common practice is to restrict formatting to safe HTML elements, such as <code>&lt;b&gt;</code> (bold) and <code>&lt;i&gt;</code> (italic).</p> <p>Before writing the data, HTML-encode it. This makes any malicious script safe by causing it to be handled as text, not as executable code.</p> <ol style="list-style-type: none"> <li>1. Disable ASP.NET request validation by adding the <code>ValidateRequest="false"</code> attribute to the <code>@ Page</code> directive</li> <li>2. Encode the string input with the <code>HtmlEncode</code> method</li> <li>3. Use a <code>StringBuilder</code> and call its <code>Replace</code> method to selectively remove the encoding on the HTML elements that you want to permit</li> </ol> <p>The page-in the references disables ASP.NET request validation by setting <code>ValidateRequest="false"</code>. It HTML-encodes the input and selectively allows the <code>&lt;b&gt;</code> and <code>&lt;i&gt;</code>. Alternatively, a .NET library for HTML sanitization may also be used.</p> <p><code>HtmlSanitizer</code> is a .NET library for cleaning HTML fragments and documents from constructs that can lead to XSS attacks. It uses AngleSharp to parse, manipulate, and render HTML and CSS. <code>HtmlSanitizer</code> can be installed as a NuGet package, and the user input can be passed through relevant HTML or CSS sanitization methods, as applicable, on the server side. Please note that Sanitization as a security control should be considered only as a last option.</p> <p>Input validation and Output Encoding are considered better security controls.</p>

## Do not assign DOM elements to sinks that do not have inbuilt encoding

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Many javascript functions don't do encoding by default. When assigning untrusted input to DOM elements via such functions, may result in cross site script (XSS) executions.

### Example

Following are insecure examples:

```

document.getElementById("div1").innerHTML = value;
$("#userName").html(res.Name);
return $('<div>').html(value)
$('body').append(resHTML);

```

Don't use `innerHTML`; instead use `innerText`. Similarly, instead of `$("#elm").html()`, use `$("#elm").text()`

## Validate all redirects within the application are closed or done safely

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">The OAuth 2.0 Authorization Framework - Open Redirectors</a>
<b>Steps</b>	<p>Application design requiring redirection to a user-supplied location must constrain the possible redirection targets to a predefined "safe" list of sites or domains. All redirects in the application must be closed/safe.</p> <p>To do this:</p> <ul style="list-style-type: none"> <li>• Identify all redirects</li> <li>• Implement an appropriate mitigation for each redirect. Appropriate mitigations include redirect whitelist or user confirmation. If a web site or service with an open redirect vulnerability uses Facebook/OAuth/OpenID identity providers, an attacker can steal a user's logon token and impersonate that user. This is an inherent risk when using OAuth, which is documented in RFC 6749 "The OAuth 2.0 Authorization Framework", Section 10.15 "Open Redirects". Similarly, users' credentials can be compromised by spear phishing attacks using open redirects</li> </ul>

## Implement input validation on all string type parameters accepted by Controller methods

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, MVC5, MVC6
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	<a href="#">Validating Model Data in an MVC Application, Guiding Principles For Your ASP.NET MVC Applications</a>
<b>Steps</b>	For methods that just accept primitive data type, and not models as argument, input validation using Regular Expression should be done. Here Regex.IsMatch should be used with a valid regex pattern. If the input doesn't match the specified Regular Expression, control should not proceed further, and an adequate warning regarding validation failure should be displayed.

Set upper limit timeout for regular expression processing to prevent DoS due to bad regular expressions

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, Web Forms, MVC5, MVC6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">DefaultRegexMatchTimeout Property</a>
<b>Steps</b>	To ensure denial of service attacks against badly created regular expressions, that cause a lot of backtracking, set the global default timeout. If the processing time takes longer than the defined upper limit, it would throw a Timeout exception. If nothing is configured, the timeout would be infinite.

### Example

For example, the following configuration will throw a RegexMatchTimeoutException, if the processing takes more than 5 seconds:

```
<httpRuntime targetFramework="4.5" defaultRegexMatchTimeout="00:00:05" />
```

### Avoid using Html.Raw in Razor views

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC5, MVC6
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	N/A
Step	ASP.Net WebPages (Razor) perform automatic HTML encoding. All strings printed by embedded code nuggets (@ blocks) are automatically HTML-encoded. However, when <code>HtmlHelper.Raw</code> Method is invoked, it returns markup that is not HTML encoded. If <code>Html.Raw()</code> helper method is used, it bypasses the automatic encoding protection that Razor provides.

### Example

Following is an insecure example:

```
<div class="form-group">
 @Html.Raw(Model.AccountConfirmText)
</div>
<div class="form-group">
 @Html.Raw(Model.PaymentConfirmText)
</div>
</div>
```

Do not use `Html.Raw()` unless you need to display markup. This method does not perform output encoding implicitly. Use other ASP.NET helpers e.g., `@Html.DisplayFor()`

## Do not use dynamic queries in stored procedures

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	A SQL injection attack exploits vulnerabilities in input validation to run arbitrary commands in the database. It can occur when your application uses input to construct dynamic SQL statements to access the database. It can also occur if your code uses stored procedures that are passed strings that contain raw user input. Using the SQL injection attack, the attacker can execute arbitrary commands in the database. All SQL statements (including the SQL statements in stored procedures) must be parameterized. Parameterized SQL statements will accept characters that have special meaning to SQL (like single quote) without problems because they are strongly typed.

### Example

Following is an example of insecure dynamic Stored Procedure:

```

CREATE PROCEDURE [dbo].[uspGetProductsByCriteria]
(
 @productName nvarchar(200) = NULL,
 @startPrice float = NULL,
 @endPrice float = NULL
)
AS
BEGIN
 DECLARE @sql nvarchar(max)
 SELECT @sql = ' SELECT ProductID, ProductName, Description, UnitPrice, ImagePath' +
 ' FROM dbo.Products WHERE 1 = 1 '
 PRINT @sql
 IF @productName IS NOT NULL
 SELECT @sql = @sql + ' AND ProductName LIKE ''%' + @productName + '%''''
 IF @startPrice IS NOT NULL
 SELECT @sql = @sql + ' AND UnitPrice > ''' + CONVERT(VARCHAR(10),@startPrice) + ''''
 IF @endPrice IS NOT NULL
 SELECT @sql = @sql + ' AND UnitPrice < ''' + CONVERT(VARCHAR(10),@endPrice) + ''''

 PRINT @sql
 EXEC(@sql)
END

```

## Example

Following is the same stored procedure implemented securely:

```

CREATE PROCEDURE [dbo].[uspGetProductsByCriteriaSecure]
(
 @productName nvarchar(200) = NULL,
 @startPrice float = NULL,
 @endPrice float = NULL
)
AS
BEGIN
 SELECT ProductID, ProductName, Description, UnitPrice, ImagePath
 FROM dbo.Products where
 (@productName IS NULL or ProductName like '%' + @productName + '%')
 AND
 (@startPrice IS NULL or UnitPrice > @startPrice)
 AND
 (@endPrice IS NULL or UnitPrice < @endPrice)
END

```

## Ensure that model validation is done on Web API methods

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC5, MVC6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Model Validation in ASP.NET Web API</a>

TITLE	DETAILS
<b>Steps</b>	When a client sends data to a web API, it is mandatory to validate the data before doing any processing. For ASP.NET Web APIs which accept models as input, use data annotations on models to set validation rules on the properties of the model.

### Example

The following code demonstrates the same:

```
using System.ComponentModel.DataAnnotations;

namespace MyApi.Models
{
 public class Product
 {
 public int Id { get; set; }
 [Required]
 [RegularExpression(@"^[\w-]*$", ErrorMessage="Only alphanumeric characters are allowed.")]
 public string Name { get; set; }
 public decimal Price { get; set; }
 [Range(0, 999)]
 public double Weight { get; set; }
 }
}
```

### Example

In the action method of the API controllers, validity of the model has to be explicitly checked as shown below:

```
namespace MyApi.Controllers
{
 public class ProductsController : ApiController
 {
 public HttpResponseMessage Post(Product product)
 {
 if (ModelState.IsValid)
 {
 // Do something with the product (not shown).

 return new HttpResponseMessage(HttpStatusCode.OK);
 }
 else
 {
 return Request.CreateErrorResponse(HttpStatusCode.BadRequest, ModelState);
 }
 }
 }
}
```

## Implement input validation on all string type parameters accepted by Web API methods

TITLE	DETAILS
<b>Component</b>	Web API

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, MVC 5, MVC 6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Validating Model Data in an MVC Application, Guiding Principles For Your ASP.NET MVC Applications</a>
<b>Steps</b>	For methods that just accept primitive data type, and not models as argument, input validation using Regular Expression should be done. Here Regex.IsMatch should be used with a valid regex pattern. If the input doesn't match the specified Regular Expression, control should not proceed further, and an adequate warning regarding validation failure should be displayed.

## Ensure that type-safe parameters are used in Web API for data access

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	<p>If you use the Parameters collection, SQL treats the input as a literal value rather than as executable code. The Parameters collection can be used to enforce type and length constraints on input data. Values outside of the range trigger an exception. If type-safe SQL parameters are not used, attackers might be able to execute injection attacks that are embedded in the unfiltered input.</p> <p>Use type safe parameters when constructing SQL queries to avoid possible SQL injection attacks that can occur with unfiltered input. You can use type safe parameters with stored procedures and with dynamic SQL statements. Parameters are treated as literal values by the database and not as executable code. Parameters are also checked for type and length.</p>

### Example

The following code shows how to use type safe parameters with the SqlParameterCollection when calling a stored procedure.

```

using System.Data;
using System.Data.SqlClient;

using (SqlConnection connection = new SqlConnection(connectionString))
{
 DataSet userDataset = new DataSet();
 SqlDataAdapter myCommand = new SqlDataAdapter("LoginStoredProcedure", connection);
 myCommand.SelectCommand.CommandType = CommandType.StoredProcedure;
 myCommand.SelectCommand.Parameters.Add("@au_id", SqlDbType.VarChar, 11);
 myCommand.SelectCommand.Parameters["@au_id"].Value = SSN.Text;
 myCommand.Fill(userDataset);
}

```

In the preceding code example, the input value cannot be longer than 11 characters. If the data does not conform to the type or length defined by the parameter, the SqlParameter class throws an exception.

## Use parametrized SQL queries for Cosmos DB

TITLE	DETAILS
<b>Component</b>	Azure Document DB
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Announcing SQL Parameterization in DocumentDB</a>
<b>Steps</b>	Although DocumentDB only supports read-only queries, SQL injection is still possible if queries are constructed by concatenating with user input. It might be possible for a user to gain access to data they shouldn't be accessing within the same collection by crafting malicious SQL queries. Use parameterized SQL queries if queries are constructed based on user input.

## WCF Input validation through Schema binding

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, .NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a>

TITLE	DETAILS
<b>Steps</b>	<p>Lack of validation leads to different type injection attacks.</p> <p>Message validation represents one line of defense in the protection of your WCF application. With this approach, you validate messages using schemas to protect WCF service operations from attack by a malicious client.</p> <p>Validate all messages received by the client to protect the client from attack by a malicious service. Message validation makes it possible to validate messages when operations consume message contracts or data contracts, which cannot be done using parameter validation.</p> <p>Message validation allows you to create validation logic inside schemas, thereby providing more flexibility and reducing development time. Schemas can be reused across different applications inside the organization, creating standards for data representation. Additionally, message validation allows you to protect operations when they consume more complex data types involving contracts representing business logic.</p> <p>To perform message validation, you first build a schema that represents the operations of your service and the data types consumed by those operations. You then create a .NET class that implements a custom client message inspector and custom dispatcher message inspector to validate the messages sent/received to/from the service. Next, you implement a custom endpoint behavior to enable message validation on both the client and the service. Finally, you implement a custom configuration element on the class that allows you to expose the extended custom endpoint behavior in the configuration file of the service or the client"</p>

## WCF- Input validation through Parameter Inspectors

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, .NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN</a>

TITLE	DETAILS
<b>Steps</b>	<p>Input and data validation represents one important line of defense in the protection of your WCF application. You should validate all parameters exposed in WCF service operations to protect the service from attack by a malicious client. Conversely, you should also validate all return values received by the client to protect the client from attack by a malicious service.</p> <p>WCF provides different extensibility points that allow you to customize the WCF runtime behavior by creating custom extensions. Message Inspectors and Parameter Inspectors are two extensibility mechanisms used to gain greater control over the data passing between a client and a service. You should use parameter inspectors for input validation and use message inspectors only when you need to inspect the entire message flowing in and out of a service.</p> <p>To perform input validation, you will build a .NET class and implement a custom parameter inspector in order to validate parameters on operations in your service. You will then implement a custom endpoint behavior to enable validation on both the client and the service. Finally, you will implement a custom configuration element on the class that allows you to expose the extended custom endpoint behavior in the configuration file of the service or the client.</p>

# Security Frame: Sensitive Data | Mitigations

8/22/2017 • 16 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
<b>Machine Trust Boundary</b>	<ul style="list-style-type: none"><li>• Ensure that binaries are obfuscated if they contain sensitive information</li><li>• Consider using Encrypted File System (EFS) is used to protect confidential user-specific data</li><li>• Ensure that sensitive data stored by the application on the file system is encrypted</li></ul>
<b>Web Application</b>	<ul style="list-style-type: none"><li>• Ensure that sensitive content is not cached on the browser</li><li>• Encrypt sections of Web App's configuration files that contain sensitive data</li><li>• Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs</li><li>• Ensure that sensitive data displayed on the user screen is masked</li></ul>
<b>Database</b>	<ul style="list-style-type: none"><li>• Implement dynamic data masking to limit sensitive data exposure non privileged users</li><li>• Ensure that passwords are stored in salted hash format</li><li>• Ensure that sensitive data in database columns is encrypted</li><li>• Ensure that database-level encryption (TDE) is enabled</li><li>• Ensure that database backups are encrypted</li></ul>
<b>Web API</b>	<ul style="list-style-type: none"><li>• Ensure that sensitive data relevant to Web API is not stored in browser's storage</li></ul>
Azure Document DB	<ul style="list-style-type: none"><li>• Encrypt sensitive data stored in DocumentDB</li></ul>
<b>Azure IaaS VM Trust Boundary</b>	<ul style="list-style-type: none"><li>• Use Azure Disk Encryption to encrypt disks used by Virtual Machines</li></ul>
<b>Service Fabric Trust Boundary</b>	<ul style="list-style-type: none"><li>• Encrypt secrets in Service Fabric applications</li></ul>
<b>Dynamics CRM</b>	<ul style="list-style-type: none"><li>• Perform security modeling and use Business Units/Teams where required</li><li>• Minimize access to share feature on critical entities</li><li>• Train users on the risks associated with the Dynamics CRM Share feature and good security practices</li><li>• Include a development standards rule proscribing showing config details in exception management</li></ul>

PRODUCT/SERVICE	ARTICLE
Azure Storage	<ul style="list-style-type: none"> <li>• Use Azure Storage Service Encryption (SSE) for Data at Rest (Preview)</li> <li>• Use Client-Side Encryption to store sensitive data in Azure Storage</li> </ul>
Mobile Client	<ul style="list-style-type: none"> <li>• Encrypt sensitive or PII data written to phones local storage</li> <li>• Obfuscate generated binaries before distributing to end users</li> </ul>
WCF	<ul style="list-style-type: none"> <li>• Set clientCredentialType to Certificate or Windows</li> <li>• WCF-Security Mode is not enabled</li> </ul>

Ensure that binaries are obfuscated if they contain sensitive information

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Deployment
Applicable Technologies	Generic
Attributes	N/A
References	N/A
Steps	Ensure that binaries are obfuscated if they contain sensitive information such as trade secrets, sensitive business logic that should not be reversed. This is to stop reverse engineering of assemblies. Tools like <a href="#">CryptoObfuscator</a> may be used for this purpose.

Consider using Encrypted File System (EFS) is used to protect confidential user-specific data

TITLE	DETAILS
Component	Machine Trust Boundary
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
<b>Steps</b>	Consider using Encrypted File System (EFS) is used to protect confidential user-specific data from adversaries with physical access to the computer.

## Ensure that sensitive data stored by the application on the file system is encrypted

TITLE	DETAILS
<b>Component</b>	Machine Trust Boundary
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Ensure that sensitive data stored by the application on the file system is encrypted (e.g., using DPAPI), if EFS cannot be enforced

## Ensure that sensitive content is not cached on the browser

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, Web Forms, MVC5, MVC6
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Browsers can store information for purposes of caching and history. These cached files are stored in a folder, like the Temporary Internet Files folder in the case of Internet Explorer. When these pages are referred again, the browser displays them from its cache. If sensitive information is displayed to the user (such as their address, credit card details, Social Security Number, or username), then this information could be stored in browser's cache, and therefore retrievable through examining the browser's cache or by simply pressing the browser's "Back" button. Set cache-control response header value to "no-store" for all pages.

### Example

```

<configuration>
 <system.webServer>
 <httpProtocol>
 <customHeaders>
 <add name="Cache-Control" value="no-cache" />
 <add name="Pragma" value="no-cache" />
 <add name="Expires" value="-1" />
 </customHeaders>
 </httpProtocol>
 </system.webServer>
</configuration>

```

## Example

This may be implemented through a filter. Following example may be used:

```

public override void OnActionExecuting(ActionExecutingContext filterContext)
{
 if (filterContext == null || (filterContext.HttpContext != null &&
filterContext.HttpContext.Response != null && filterContext.HttpContext.Response.IsRequestBeingRedirected))
 {
 // Since this is MVC pipeline, this should never be null.
 return;
 }

 var attributes =
filterContext.ActionDescriptor.GetCustomAttributes(typeof(System.Web.Mvc.OutputCacheAttribute), false);
 if (attributes == null || **Attributes**.Count() == 0)
 {
 filterContext.HttpContext.Response.Cache.SetNoStore();
 filterContext.HttpContext.Response.Cache.SetCacheability(HttpCacheability.NoCache);
 filterContext.HttpContext.Response.Cache.SetExpires(DateTime.UtcNow.AddHours(-1));
 if (!filterContext.IsChildAction)
 {
 filterContext.HttpContext.Response.AppendHeader("Pragma", "no-cache");
 }
 }

 base.OnActionExecuting(filterContext);
}

```

## Encrypt sections of Web App's configuration files that contain sensitive data

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">How To: Encrypt Configuration Sections in ASP.NET 2.0 Using DPAPI, Specifying a Protected Configuration Provider, Using Azure Key Vault to protect application secrets</a>

TITLE	DETAILS
<b>Steps</b>	Configuration files such as the Web.config, appsettings.json are often used to hold sensitive information, including user names, passwords, database connection strings, and encryption keys. If you do not protect this information, your application is vulnerable to attackers or malicious users obtaining sensitive information such as account user names and passwords, database names and server names. Based on the deployment type (azure/on-prem), encrypt the sensitive sections of config files using DPAPI or services like Azure Key Vault.

## Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">MSDN: autocomplete attribute, Using AutoComplete in HTML</a> , <a href="#">HTML Sanitization Vulnerability, Autocomplete.,again?!</a>
<b>Steps</b>	The autocomplete attribute specifies whether a form should have autocomplete on or off. When autocomplete is on, the browser automatically complete values based on values that the user has entered before. For example, when a new name and password is entered in a form and the form is submitted, the browser asks if the password should be saved. Thereafter when the form is displayed, the name and password are filled in automatically or are completed as the name is entered. An attacker with local access could obtain the clear text password from the browser cache. By default autocomplete is enabled, and it must explicitly be disabled.

### Example

```
<form action="Login.aspx" method="post" autocomplete="off">
 Social Security Number: <input type="text" name="ssn" />
 <input type="submit" value="Submit" />
</form>
```

## Ensure that sensitive data displayed on the user screen is masked

TITLE	DETAILS
<b>Component</b>	Web Application

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Sensitive data such as passwords, credit card numbers, SSN etc. should be masked when displayed on the screen. This is to prevent unauthorized personnel from accessing the data (e.g., shoulder-surfing passwords, support personnel viewing SSN numbers of users) . Ensure that these data elements are not visible in plain text and are appropriately masked. This has to be taken care while accepting them as input (e.g., input type="password") as well as displaying back on the screen (e.g., display only the last 4 digits of the credit card number).

## Implement dynamic data masking to limit sensitive data exposure non privileged users

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Sql Azure, OnPrem
<b>Attributes</b>	SQL Version - V12, SQL Version - MsSQL2016
<b>References</b>	<a href="#">Dynamic Data Masking</a>
<b>Steps</b>	The purpose of dynamic data masking is to limit exposure of sensitive data, preventing users who should not have access to the data from viewing it. Dynamic data masking does not aim to prevent database users from connecting directly to the database and running exhaustive queries that expose pieces of the sensitive data. Dynamic data masking is complementary to other SQL Server security features (auditing, encryption, row level security...) and it is highly recommended to use this feature in conjunction with them in addition in order to better protect the sensitive data in the database. Please note that this feature is supported only by SQL Server starting with 2016 and Azure SQL Database.

## Ensure that passwords are stored in salted hash format

TITLE	DETAILS
<b>Component</b>	Database

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Password Hashing using .NET Crypto APIs</a>
<b>Steps</b>	Passwords should not be stored in custom user store databases. Password hashes should be stored with salt values instead. Make sure the salt for the user is always unique and you apply b-crypt, s-crypt or PBKDF2 before storing the password, with a minimum work factor iteration count of 150,000 loops to eliminate the possibility of brute forcing.

## Ensure that sensitive data in database columns is encrypted

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	SQL Version - All
<b>References</b>	<a href="#">Encrypting sensitive data in SQL server</a> , <a href="#">How to: Encrypt a Column of Data in SQL Server, Encrypt by Certificate</a>
<b>Steps</b>	Sensitive data such as credit card numbers has to be encrypted in the database. Data can be encrypted using column-level encryption or by an application function using the encryption functions.

## Ensure that database-level encryption (TDE) is enabled

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Understanding SQL Server Transparent Data Encryption (TDE)</a>

TITLE	DETAILS
<b>Steps</b>	Transparent Data Encryption (TDE) feature in SQL server helps in encrypting sensitive data in a database and protect the keys that are used to encrypt the data with a certificate. This prevents anyone without the keys from using the data. TDE protects data "at rest", meaning the data and log files. It provides the ability to comply with many laws, regulations, and guidelines established in various industries.

## Ensure that database backups are encrypted

TITLE	DETAILS
<b>Component</b>	Database
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	SQL Azure, OnPrem
<b>Attributes</b>	SQL Version - V12, SQL Version - MsSQL2014
<b>References</b>	<a href="#">SQL database backup encryption</a>
<b>Steps</b>	SQL Server has the ability to encrypt the data while creating a backup. By specifying the encryption algorithm and the encryptor (a Certificate or Asymmetric Key) when creating a backup, one can create an encrypted backup file.

## Ensure that sensitive data relevant to Web API is not stored in browser's storage

TITLE	DETAILS
<b>Component</b>	Web API
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC 5, MVC 6
<b>Attributes</b>	Identity Provider - ADFS, Identity Provider - Azure AD
<b>References</b>	N/A

TITLE	DETAILS
<b>Steps</b>	<p>In certain implementations, sensitive artifacts relevant to Web API's authentication are stored in browser's local storage. E.g., Azure AD authentication artifacts like adal.idtoken, adal.nonce.idtoken, adal.access.token.key, adal.token.keys, adal.state.login, adal.session.state, adal.expiration.key etc.</p> <p>All these artifacts are available even after sign out or browser is closed. If an adversary gets access to these artifacts, he/she can reuse them to access the protected resources (APIs). Ensure that all sensitive artifacts related to Web API is not stored in browser's storage. In cases where client-side storage is unavoidable (e.g., Single Page Applications (SPA) that leverage Implicit OpenIdConnect/OAuth flows need to store access tokens locally), use storage choices with do not have persistence. e.g., prefer SessionStorage to LocalStorage.</p>

### Example

The below JavaScript snippet is from a custom authentication library which stores authentication artifacts in local storage. Such implementations should be avoided.

```
ns.AuthHelper.Authenticate = function () {
 window.config = {
 instance: 'https://login.microsoftonline.com/',
 tenant: ns.Configurations.Tenant,
 clientId: ns.Configurations.AADApplicationClientID,
 postLogoutRedirectUri: window.location.origin,
 cacheLocation: 'localStorage', // enable this for IE, as sessionStorage does not work for localhost.
 };
}
```

## Encrypt sensitive data stored in Cosmos DB

TITLE	DETAILS
<b>Component</b>	Azure Document DB
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Encrypt sensitive data at application level before storing in document DB or store any sensitive data in other storage solutions like Azure Storage or Azure SQL

## Use Azure Disk Encryption to encrypt disks used by Virtual Machines

TITLE	DETAILS
<b>Component</b>	Azure IaaS VM Trust Boundary
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Using Azure Disk Encryption to encrypt disks used by your virtual machines</a>
<b>Steps</b>	<p>Azure Disk Encryption is a new feature that is currently in preview. This feature allows you to encrypt the OS disks and Data disks used by an IaaS Virtual Machine. For Windows, the drives are encrypted using industry-standard BitLocker encryption technology. For Linux, the disks are encrypted using the DM-Crypt technology. This is integrated with Azure Key Vault to allow you to control and manage the disk encryption keys. The Azure Disk Encryption solution supports the following three customer encryption scenarios:</p> <ul style="list-style-type: none"> <li>• Enable encryption on new IaaS VMs created from customer-encrypted VHD files and customer-provided encryption keys, which are stored in Azure Key Vault.</li> <li>• Enable encryption on new IaaS VMs created from the Azure Marketplace.</li> <li>• Enable encryption on existing IaaS VMs already running in Azure.</li> </ul>

## Encrypt secrets in Service Fabric applications

TITLE	DETAILS
<b>Component</b>	Service Fabric Trust Boundary
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	Environment - Azure
<b>References</b>	<a href="#">Managing secrets in Service Fabric applications</a>
<b>Steps</b>	<p>Secrets can be any sensitive information, such as storage connection strings, passwords, or other values that should not be handled in plain text. Use Azure Key Vault to manage keys and secrets in service fabric applications.</p>

Perform security modeling and use Business Units/Teams where required

TITLE	DETAILS
<b>Component</b>	Dynamics CRM
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Perform security modeling and use Business Units/Teams where required

## Minimize access to share feature on critical entities

TITLE	DETAILS
<b>Component</b>	Dynamics CRM
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Minimize access to share feature on critical entities

## Train users on the risks associated with the Dynamics CRM Share feature and good security practices

TITLE	DETAILS
<b>Component</b>	Dynamics CRM
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Train users on the risks associated with the Dynamics CRM Share feature and good security practices

## Include a development standards rule proscribing showing config

## details in exception management

TITLE	DETAILS
<b>Component</b>	Dynamics CRM
<b>SDL Phase</b>	Deployment
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Include a development standards rule proscribing showing config details in exception management outside development. Test for this as part of code reviews or periodic inspection.

## Use Azure Storage Service Encryption (SSE) for Data at Rest (Preview)

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	StorageType - Blob
<b>References</b>	<a href="#">Azure Storage Service Encryption for Data at Rest (Preview)</a>

TITLE	DETAILS
<b>Steps</b>	<p>Azure Storage Service Encryption (SSE) for Data at Rest helps you protect and safeguard your data to meet your organizational security and compliance commitments. With this feature, Azure Storage automatically encrypts your data prior to persisting to storage and decrypts prior to retrieval. The encryption, decryption and key management is totally transparent to users. SSE applies only to block blobs, page blobs, and append blobs. The other types of data, including tables, queues, and files, will not be encrypted.</p> <p>Encryption and Decryption Workflow:</p> <ul style="list-style-type: none"> <li>• The customer enables encryption on the storage account</li> <li>• When the customer writes new data (PUT Blob, PUT Block, PUT Page, etc.) to Blob storage; every write is encrypted using 256-bit AES encryption, one of the strongest block ciphers available</li> <li>• When the customer needs to access data (GET Blob, etc.), data is automatically decrypted before returning to the user</li> <li>• If encryption is disabled, new writes are no longer encrypted and existing encrypted data remains encrypted until rewritten by the user. While encryption is enabled, writes to Blob storage will be encrypted. The state of data does not change with the user toggling between enabling/disabling encryption for the storage account</li> <li>• All encryption keys are stored, encrypted, and managed by Microsoft</li> </ul> <p>Please note that at this time, the keys used for the encryption are managed by Microsoft. Microsoft generates the keys originally, and manage the secure storage of the keys as well as the regular rotation as defined by internal Microsoft policy. In the future, customers will get the ability to manage their own encryption keys, and provide a migration path from Microsoft-managed keys to customer-managed keys.</p>

## Use Client-Side Encryption to store sensitive data in Azure Storage

TITLE	DETAILS
<b>Component</b>	Azure Storage
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage, Tutorial: Encrypt and decrypt blobs in Microsoft Azure Storage using Azure Key Vault, Storing Data Securely in Azure Blob Storage with Azure Encryption Extensions</a>

TITLE	DETAILS
<b>Steps</b>	<p>The Azure Storage Client Library for .NET Nuget package supports encrypting data within client applications before uploading to Azure Storage, and decrypting data while downloading to the client. The library also supports integration with Azure Key Vault for storage account key management. Here is a brief description of how client side encryption works:</p> <ul style="list-style-type: none"> <li>• The Azure Storage client SDK generates a content encryption key (CEK), which is a one-time-use symmetric key</li> <li>• Customer data is encrypted using this CEK</li> <li>• The CEK is then wrapped (encrypted) using the key encryption key (KEK). The KEK is identified by a key identifier and can be an asymmetric key pair or a symmetric key and can be managed locally or stored in Azure Key Vault. The Storage client itself never has access to the KEK. It just invokes the key wrapping algorithm that is provided by Key Vault. Customers can choose to use custom providers for key wrapping/unwrapping if they want</li> <li>• The encrypted data is then uploaded to the Azure Storage service. Check the links in the references section for low-level implementation details.</li> </ul>

## Encrypt sensitive or PII data written to phones local storage

TITLE	DETAILS
<b>Component</b>	Mobile Client
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, Xamarin
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Manage settings and features on your devices with Microsoft Intune policies</a> , <a href="#">Keychain Valet</a>
<b>Steps</b>	<p>If the application writes sensitive information like user's PII (email, phone number, first name, last name, preferences etc.)- on mobile's file system, then it should be encrypted before writing to the local file system. If the application is an enterprise application, then explore the possibility of publishing application using Windows Intune.</p>

### Example

Intune can be configured with following security policies to safeguard sensitive data:

```
Require encryption on mobile device
Require encryption on storage cards
Allow screen capture
```

## Example

If the application is not an enterprise application, then use platform provided keystore, keychains to store encryption keys, using which cryptographic operation may be performed on the file system. Following code snippet shows how to access key from keychain using xamarin:

```
protected static string EncryptionKey
{
 get
 {
 if (String.IsNullOrEmpty(_Key))
 {
 var query = new SecRecord(SecKind.GenericPassword);
 query.Service = NSBundle.MainBundle.BundleIdentifier;
 query.Account = "UniqueID";

 NSData uniqueId = SecKeyChain.QueryAsData(query);
 if (uniqueId == null)
 {
 query.ValueData = NSData.FromString(System.Guid.NewGuid().ToString());
 var err = SecKeyChain.Add(query);
 _Key = query.ValueData.ToString();
 }
 else
 {
 _Key = uniqueId.ToString();
 }
 }

 return _Key;
 }
}
```

## Obfuscate generated binaries before distributing to end users

TITLE	DETAILS
<b>Component</b>	Mobile Client
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Crypto Obfuscation For .Net</a>
<b>Steps</b>	Generated binaries (assemblies within apk) should be obfuscated to stop reverse engineering of assemblies. Tools like <a href="#">Cryptoobfuscator</a> may be used for this purpose.

## Set clientCredentialType to Certificate or Windows

TITLE	DETAILS
<b>Component</b>	WCF

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	.NET Framework 3
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Fortify</a>
<b>Steps</b>	Using a UsernameToken with a plaintext password over an unencrypted channel exposes the password to attackers who can sniff the SOAP messages. Service Providers that use the UsernameToken might accept passwords sent in plaintext. Sending plaintext passwords over an unencrypted channel can expose the credential to attackers who can sniff the SOAP message.

### Example

The following WCF service provider configuration uses the UsernameToken:

```
<security mode="Message">
<message clientCredentialType="UserName" />
```

Set clientCredentialType to Certificate or Windows.

## WCF-Security Mode is not enabled

TITLE	DETAILS
<b>Component</b>	WCF
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic, .NET Framework 3
<b>Attributes</b>	Security Mode - Transport, Security Mode - Message
<b>References</b>	<a href="#">MSDN</a> , <a href="#">Fortify Kingdom</a> , <a href="#">Fundamentals of WCF Security CoDe Magazine</a>
<b>Steps</b>	No transport or message security has been defined. Applications that transmit messages without transport or message security cannot guarantee the integrity or confidentiality of the messages. When a WCF security binding is set to None, both transport and message security are disabled.

### Example

The following configuration sets the security mode to None.

```
<system.serviceModel>
 <bindings>
 <wsHttpBinding>
 <binding name=""MyBinding"">
 <security mode=""None""/>
 </binding>
 </bindings>
 </system.serviceModel>
```

## Example

Security Mode Across all service bindings there are five possible security modes:

- None. Turns security off.
- Transport. Uses transport security for mutual authentication and message protection.
- Message. Uses message security for mutual authentication and message protection.
- Both. Allows you to supply settings for transport and message-level security (only MSMQ supports this).
- TransportWithMessageCredential. Credentials are passed with the message and message protection and server authentication are provided by the transport layer.
- TransportCredentialOnly. Client credentials are passed with the transport layer and no message protection is applied. Use transport and message security to protect the integrity and confidentiality of messages. The configuration below tells the service to use transport security with message credentials.

```
<system.serviceModel> <bindings> <wsHttpBinding> <binding name=""MyBinding""> <security
mode=""TransportWithMessageCredential""/> <message clientCredentialType=""Windows""/> </binding> </bindings>
</system.serviceModel>
```

# Security Frame: Session Management | Articles

8/22/2017 • 14 min to read • [Edit Online](#)

PRODUCT/SERVICE	ARTICLE
Azure AD	<ul style="list-style-type: none"><li>• Implement proper logout using ADAL methods when using Azure AD</li></ul>
IoT Device	<ul style="list-style-type: none"><li>• Use finite lifetimes for generated SaaS tokens</li></ul>
Azure Document DB	<ul style="list-style-type: none"><li>• Use minimum token lifetimes for generated Resource tokens</li></ul>
ADFS	<ul style="list-style-type: none"><li>• Implement proper logout using WsFederation methods when using ADFS</li></ul>
Identity Server	<ul style="list-style-type: none"><li>• Implement proper logout when using Identity Server</li></ul>
Web Application	<ul style="list-style-type: none"><li>• Applications available over HTTPS must use secure cookies</li><li>• All http based application should specify http only for cookie definition</li><li>• Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET web pages</li><li>• Set up session for inactivity lifetime</li><li>• Implement proper logout from the application</li></ul>
Web API	<ul style="list-style-type: none"><li>• Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET Web APIs</li></ul>

## Implement proper logout using ADAL methods when using Azure AD

TITLE	DETAILS
Component	Azure AD
SDL Phase	Build
Applicable Technologies	Generic
Attributes	N/A
References	N/A

TITLE	DETAILS
<b>Steps</b>	If the application relies on access token issued by Azure AD, the logout event handler should call

## Example

```
HttpContext.GetOwinContext().Authentication.SignOut(OpenIdConnectAuthenticationDefaults.AuthenticationType,
CookieAuthenticationDefaults.AuthenticationType)
```

## Example

It should also destroy user's session by calling Session.Abandon() method. Following method shows secure implementation of user logout:

```
[HttpPost]
[ValidateAntiForgeryToken]
public void LogOff()
{
 string userObjectID =
ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/objectidentifier").Value;
 AuthenticationContext authContext = new AuthenticationContext(Authority + TenantId, new
NaiveSessionCache(userObjectID));
 authContext.TokenCache.Clear();
 Session.Clear();
 Session.Abandon();
 Response.SetCookie(new HttpCookie("ASP.NET_SessionId", string.Empty));
 HttpContext.GetOwinContext().Authentication.SignOut(
 OpenIdConnectAuthenticationDefaults.AuthenticationType,
 CookieAuthenticationDefaults.AuthenticationType);
}
```

## Use finite lifetimes for generated SaS tokens

TITLE	DETAILS
<b>Component</b>	IoT Device
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	SaS tokens generated for authenticating to Azure IoT Hub should have a finite expiry period. Keep the SaS token lifetimes to a minimum to limit the amount of time they can be replayed in case the tokens are compromised.

## Use minimum token lifetimes for generated Resource tokens

TITLE	DETAILS
<b>Component</b>	Azure Document DB
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Reduce the timespan of resource token to a minimum value required. Resource tokens have a default valid timespan of 1 hour.

## Implement proper logout using WsFederation methods when using ADFS

TITLE	DETAILS
<b>Component</b>	ADFS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	If the application relies on STS token issued by ADFS, the logout event handler should call <code>WSFederationAuthenticationModule.FederatedSignOut()</code> method to log out the user. Also the current session should be destroyed, and the session token value should be reset and nullified.

### Example

```

[HttpPost, ValidateAntiForgeryToken]
[Authorization]
public ActionResult SignOut(string redirectUrl)
{
 if (!this.User.Identity.IsAuthenticated)
 {
 return this.View("LogOff", null);
 }

 // Removes the user profile.
 this.Session.Clear();
 this.Session.Abandon();
 HttpContext.Current.Response.Cookies.Add(new System.Web.HttpCookie("ASP.NET_SessionId",
string.Empty)
 {
 Expires = DateTime.Now.AddDays(-1D),
 Secure = true,
 HttpOnly = true
 });

 // Signs out at the specified security token service (STS) by using the WS-Federation protocol.
 Uri signOutUrl = new Uri(FederatedAuthentication.WSFederationAuthenticationModule.Issuer);
 Uri replyUrl = new Uri(FederatedAuthentication.WSFederationAuthenticationModule.Realm);
 if (!string.IsNullOrEmpty(redirectUrl))
 {
 replyUrl = new Uri(FederatedAuthentication.WSFederationAuthenticationModule.Realm +
redirectUrl);
 }
 // Signs out of the current session and raises the appropriate events.
 var authModule = FederatedAuthentication.WSFederationAuthenticationModule;
 authModule.SignOut(false);
 // Signs out at the specified security token service (STS) by using the WS-Federation
 // protocol.
 WSFederationAuthenticationModule.FederatedSignOut(signOutUrl, replyUrl);
 return new RedirectResult(redirectUrl);
}

```

## Implement proper logout when using Identity Server

TITLE	DETAILS
<b>Component</b>	Identity Server
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">IdentityServer3-Federated sign out</a>
<b>Steps</b>	IdentityServer supports the ability to federate with external identity providers. When a user signs out of an upstream identity provider, depending upon the protocol used, it might be possible to receive a notification when the user signs out. It allows IdentityServer to notify its clients so they can also sign the user out. Check the documentation in the references section for the implementation details.

## Applications available over HTTPS must use secure cookies

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	EnvironmentType - OnPrem
<b>References</b>	<a href="#">httpCookies Element (ASP.NET Settings Schema)</a> , <a href="#">HttpCookie.Secure Property</a>
<b>Steps</b>	Cookies are normally only accessible to the domain for which they were scoped. Unfortunately, the definition of "domain" does not include the protocol so cookies that are created over HTTPS are accessible over HTTP. The "secure" attribute indicates to the browser that the cookie should only be made available over HTTPS. Ensure that all cookies set over HTTPS use the <b>secure</b> attribute. The requirement can be enforced in the web.config file by setting the requireSSL attribute to true. It is the preferred approach because it will enforce the <b>secure</b> attribute for all current and future cookies without the need to make any additional code changes.

### Example

```
<configuration>
 <system.web>
 <httpCookies requireSSL="true"/>
 </system.web>
</configuration>
```

The setting is enforced even if HTTP is used to access the application. If HTTP is used to access the application, the setting breaks the application because the cookies are set with the secure attribute and the browser will not send them back to the application.

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Web Forms, MVC5
<b>Attributes</b>	EnvironmentType - OnPrem
<b>References</b>	N/A
<b>Steps</b>	When the web application is the Relying Party, and the IdP is ADFS server, the FedAuth token's secure attribute can be configured by setting requireSSL to True in <code>system.identityModel.services</code> section of web.config:

## Example

```
<system.identityModel.services>
 <federationConfiguration>
 <!-- Set requireSsl=true; domain=application domain name used by FedAuth cookies (Ex: .gdinfra.com); -->
 <cookieHandler requireSsl="true" persistentSessionLifetime="0.0:20:0" />
 ...
 </federationConfiguration>
</system.identityModel.services>
```

All http based application should specify http only for cookie definition

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Secure Cookie Attribute</a>
<b>Steps</b>	To mitigate the risk of information disclosure with a cross-site scripting (XSS) attack, a new attribute - httpOnly - was introduced to cookies and is supported by all major browsers. The attribute specifies that a cookie is not accessible through script. By using HttpOnly cookies, a web application reduces the possibility that sensitive information contained in the cookie can be stolen via script and sent to an attacker's website.

## Example

All HTTP-based applications that use cookies should specify HttpOnly in the cookie definition, by implementing following configuration in web.config:

```
<system.web>
 .
 .
 <httpCookies requireSSL="false" httpOnlyCookies="true"/>
 .
</system.web>
```

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Web Forms
<b>Attributes</b>	N/A

TITLE	DETAILS
<b>References</b>	<a href="#">FormsAuthentication.RequireSSL Property</a>
<b>Steps</b>	The RequireSSL property value is set in the configuration file for an ASP.NET application by using the requireSSL attribute of the configuration element. You can specify in the Web.config file for your ASP.NET application whether SSL (Secure Sockets Layer) is required to return the forms-authentication cookie to the server by setting the requireSSL attribute.

## Example

The following code example sets the requireSSL attribute in the Web.config file.

```
<authentication mode="Forms">
 <forms loginUrl="member_login.aspx" cookieless="UseCookies" requireSSL="true"/>
</authentication>
```

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC5
<b>Attributes</b>	EnvironmentType - OnPrem
<b>References</b>	<a href="#">Windows Identity Foundation (WIF) Configuration – Part II</a>
<b>Steps</b>	To set httpOnly attribute for FedAuth cookies, hideFromCscript attribute value should be set to True.

## Example

Following configuration shows the correct configuration:

```
<federatedAuthentication>
<cookieHandler mode="Custom"
 hideFromScript="true"
 name="FedAuth"
 path="/"
 requireSsl="true"
 persistentSessionLifetime="25">
</cookieHandler>
</federatedAuthentication>
```

## Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET web pages

TITLE	DETAILS
<b>Component</b>	Web Application

TITLE	DETAILS
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic
<b>Attributes</b>	N/A
<b>References</b>	N/A
<b>Steps</b>	Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker can carry out actions in the security context of a different user's established session on a web site. The goal is to modify or delete content, if the targeted web site relies exclusively on session cookies to authenticate received request. An attacker could exploit this vulnerability by getting a different user's browser to load a URL with a command from a vulnerable site on which the user is already logged in. There are many ways for an attacker to do that, such as by hosting a different web site that loads a resource from the vulnerable server, or getting the user to click a link. The attack can be prevented if the server sends an additional token to the client, requires the client to include that token in all future requests, and verifies that all future requests include a token that pertains to the current session, such as by using the ASP.NET AntiForgeryToken or ViewState.

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	MVC5, MVC6
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">XSRF/CSRF Prevention in ASP.NET MVC and Web Pages</a>
<b>Steps</b>	Anti-CSRF and ASP.NET MVC forms - Use the <code>AntiForgeryToken</code> helper method on Views; put an <code>Html.AntiForgeryToken()</code> into the form, for example,

## Example

```
@using (Html.BeginForm("UserProfile", "SubmitUpdate")) {
 @Html.ValidationSummary(true)
 @Html.AntiForgeryToken()
 <fieldset>
```

## Example

```

<form action="/UserProfile/SubmitUpdate" method="post">
 <input name="__RequestVerificationToken" type="hidden"
value="saTFWpkKN0BYazFtN6c4YbZAmEwG0srqlUqqloifVgeV2ciIFVmElvzwRZpArs" />
 <!-- rest of form goes here -->
</form>

```

## Example

At the same time, `Html.AntiForgeryToken()` gives the visitor a cookie called `__RequestVerificationToken`, with the same value as the random hidden value shown above. Next, to validate an incoming form post, add the `[ValidateAntiForgeryToken]` filter to the target action method. For example:

```

[ValidateAntiForgeryToken]
public ViewResult SubmitUpdate()
{
// ... etc.
}

```

Authorization filter that checks that:

- The incoming request has a cookie called `__RequestVerificationToken`
- The incoming request has a `Request.Form` entry called `__RequestVerificationToken`
- These cookie and `Request.Form` values match Assuming all is well, the request goes through as normal. But if not, then an authorization failure with message "A required anti-forgery token was not supplied or was invalid".

## Example

Anti-CSRF and AJAX: The form token can be a problem for AJAX requests, because an AJAX request might send JSON data, not HTML form data. One solution is to send the tokens in a custom HTTP header. The following code uses Razor syntax to generate the tokens, and then adds the tokens to an AJAX request.

```

<script>
@functions{
 public string TokenHeaderValue()
 {
 string cookieToken, formToken;
 AntiForgery.GetTokens(null, out cookieToken, out formToken);
 return cookieToken + ":" + formToken;
 }
}

$.ajax("api/values", {
 type: "post",
 contentType: "application/json",
 data: { }, // JSON data goes here
 dataType: "json",
 headers: {
 'RequestVerificationToken': '@TokenHeaderValue()'
 }
});
</script>

```

## Example

When you process the request, extract the tokens from the request header. Then call the `AntiForgery.Validate` method to validate the tokens. The `Validate` method throws an exception if the tokens are not valid.

```

void ValidateRequestHeader(HttpRequestMessage request)
{
 string cookieToken = "";
 string formToken = "";

 IEnumerable<string> tokenHeaders;
 if (request.Headers.TryGetValues("RequestVerificationToken", out tokenHeaders))
 {
 string[] tokens = tokenHeaders.First().Split(':');
 if (tokens.Length == 2)
 {
 cookieToken = tokens[0].Trim();
 formToken = tokens[1].Trim();
 }
 }
 AntiForgery.Validate(cookieToken, formToken);
}

```

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Web Forms
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">Take Advantage of ASP.NET Built-in Features to Fend Off Web Attacks</a>
<b>Steps</b>	CSRF attacks in WebForm based applications can be mitigated by setting ViewStateUserKey to a random string that varies for each user - user ID or, better yet, session ID. For a number of technical and social reasons, session ID is a much better fit because a session ID is unpredictable, times out, and varies on a per-user basis.

## Example

Here's the code you need to have in all of your pages:

```

void Page_Init (object sender, EventArgs e) {
 ViewStateUserKey = Session.SessionID;
 :
}

```

## Set up session for inactivity lifetime

TITLE	DETAILS
<b>Component</b>	Web Application
<b>SDL Phase</b>	Build
<b>Applicable Technologies</b>	Generic

TITLE	DETAILS
<b>Attributes</b>	N/A
<b>References</b>	<a href="#">HttpSessionState.Timeout Property</a>
<b>Steps</b>	Session timeout represents the event occurring when a user does not perform any action on a web site during a interval (defined by web server). The event, on server side, change the status of the user session to 'invalid' (for example "not used anymore") and instruct the web server to destroy it (deleting all data contained into it). The following code example sets the timeout session attribute to 15 minutes in the Web.config file.

## Example

```XML code

```
## <a id="threat-detection"></a>Enable Threat detection on Azure SQL
```

| TITLE | DETAILS |
|--------------------------------|--|
| Component | Web Application |
| SDL Phase | Build |
| Applicable Technologies | Web Forms |
| Attributes | N/A |
| References | forms Element for authentication (ASP.NET Settings Schema) |
| Steps | Set the Forms Authentication Ticket cookie timeout to 15 minutes |

Example

```XML code

| Title                       | Details                                                                                                                                                                               |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| **Component**               | Web Application                                                                                                                                                                       |
| **SDL Phase**               | Build                                                                                                                                                                                 |
| **Applicable Technologies** | Web Forms, MVC5                                                                                                                                                                       |
| **Attributes**              | EnvironmentType - OnPrem                                                                                                                                                              |
| **References**              | [asdeqa]( <a href="https://skf.azurewebsites.net/Mitigations/Details/wefn">https://skf.azurewebsites.net/Mitigations/Details/wefn</a> )                                               |
| **Steps**                   | When the web application is Relying Party and ADFS is the STS, the lifetime of the authentication cookies - FedAuth tokens - can be set by the following configuration in web.config: |

### ### Example

```XML

```

<system.identityModel.services>
  <federationConfiguration>
    <!-- Set requireSsl=true; domain=application domain name used by FedAuth cookies (Ex: .gdinfra.com); -->
    <cookieHandler requireSsl="true" persistentSessionLifetime="0.0:15:0" />
    <!-- Set requireHttps=true; -->
    <wsFederation passiveRedirectEnabled="true" issuer="http://localhost:39529/" realm="https://localhost:44302/" reply="https://localhost:44302/" requireHttps="true"/>
    <!--
      Use the code below to enable encryption-decryption of claims received from ADFS. Thumbprint value varies
      based on the certificate being used.
    <serviceCertificate>
      <certificateReference findValue="4FBBA33A1D11A9022A5BF3492FF83320007686A" storeLocation="LocalMachine"
      storeName="My" x509FindType="FindByThumbprint" />
    </serviceCertificate>
    -->
  </federationConfiguration>
</system.identityModel.services>

```

Example

Also the ADFS issued SAML claim token's lifetime should be set to 15 minutes, by executing the following powershell command on the ADFS server:

```
Set-ADFSRelyingPartyTrust -TargetName "<RelyingPartyWebApp>" -ClaimsProviderName @("Active Directory") -
TokenLifetime 15 -AlwaysRequireAuthentication $true
```

Implement proper logout from the application

| TITLE | DETAILS |
|--------------------------------|-----------------|
| Component | Web Application |
| SDL Phase | Build |
| Applicable Technologies | Generic |
| Attributes | N/A |
| References | N/A |

| TITLE | DETAILS |
|--------------|---|
| Steps | Perform proper Sign Out from the application, when user presses log out button. Upon logout, application should destroy user's session, and also reset and nullify session cookie value, along with resetting and nullifying authentication cookie value. Also, when multiple sessions are tied to a single user identity, they must be collectively terminated on the server side at timeout or logout. Lastly, ensure that Logout functionality is available on every page. |

Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET Web APIs

| TITLE | DETAILS |
|--------------------------------|---|
| Component | Web API |
| SDL Phase | Build |
| Applicable Technologies | Generic |
| Attributes | N/A |
| References | N/A |
| Steps | <p>Cross-site request forgery (CSRF or XSRF) is a type of attack in which an attacker can carry out actions in the security context of a different user's established session on a web site. The goal is to modify or delete content, if the targeted web site relies exclusively on session cookies to authenticate received request. An attacker could exploit this vulnerability by getting a different user's browser to load a URL with a command from a vulnerable site on which the user is already logged in. There are many ways for an attacker to do that, such as by hosting a different web site that loads a resource from the vulnerable server, or getting the user to click a link. The attack can be prevented if the server sends an additional token to the client, requires the client to include that token in all future requests, and verifies that all future requests include a token that pertains to the current session, such as by using the ASP.NET AntiForgeryToken or ViewState.</p> |

| TITLE | DETAILS |
|--------------------------------|---|
| Component | Web API |
| SDL Phase | Build |
| Applicable Technologies | MVC5, MVC6 |
| Attributes | N/A |
| References | Preventing Cross-Site Request Forgery (CSRF) Attacks in ASP.NET Web API |

| TITLE | DETAILS |
|--------------|--|
| Steps | Anti-CSRF and AJAX: The form token can be a problem for AJAX requests, because an AJAX request might send JSON data, not HTML form data. One solution is to send the tokens in a custom HTTP header. The following code uses Razor syntax to generate the tokens, and then adds the tokens to an AJAX request. |

Example

```
<script>
@functions{
    public string TokenHeaderValue()
    {
        string cookieToken, formToken;
        AntiForgeryToken.GetTokens(null, out cookieToken, out formToken);
        return cookieToken + ":" + formToken;
    }
}
$.ajax("api/values", {
    type: "post",
    contentType: "application/json",
    data: { }, // JSON data goes here
    dataType: "json",
    headers: {
        'RequestVerificationToken': '@TokenHeaderValue()'
    }
});
</script>
```

Example

When you process the request, extract the tokens from the request header. Then call the `AntiForgery.Validate` method to validate the tokens. The `Validate` method throws an exception if the tokens are not valid.

```
void ValidateRequestHeader(HttpRequestMessage request)
{
    string cookieToken = "";
    string formToken = "";

    IEnumerable<string> tokenHeaders;
    if (request.Headers.TryGetValues("RequestVerificationToken", out tokenHeaders))
    {
        string[] tokens = tokenHeaders.First().Split(':');
        if (tokens.Length == 2)
        {
            cookieToken = tokens[0].Trim();
            formToken = tokens[1].Trim();
        }
    }
    AntiForgery.Validate(cookieToken, formToken);
}
```

Example

Anti-CSRF and ASP.NET MVC forms - Use the `AntiForgeryToken` helper method on Views; put an `Html.AntiForgeryToken()` into the form, for example,

```

@using (Html.BeginForm("UserProfile", "SubmitUpdate")) {
    @Html.ValidationSummary(true)
    @Html.AntiForgeryToken()
    <fieldset>
}

```

Example

The example above will output something like the following:

```

<form action="/UserProfile/SubmitUpdate" method="post">
    <input name="__RequestVerificationToken" type="hidden"
    value="saTFWpkKN0BYazFtN6c4YbZAmEwG0srqlUqqloifVgeV2ciIFVmElvzwRZpAxs" />
    <!-- rest of form goes here -->
</form>

```

Example

At the same time, `Html.AntiForgeryToken()` gives the visitor a cookie called `__RequestVerificationToken`, with the same value as the random hidden value shown above. Next, to validate an incoming form post, add the `[ValidateAntiForgeryToken]` filter to the target action method. For example:

```

[ValidateAntiForgeryToken]
public ViewResult SubmitUpdate()
{
    // ... etc.
}

```

Authorization filter that checks that:

- The incoming request has a cookie called `__RequestVerificationToken`
- The incoming request has a `Request.Form` entry called `__RequestVerificationToken`
- These cookie and `Request.Form` values match Assuming all is well, the request goes through as normal. But if not, then an authorization failure with message "A required anti-forgery token was not supplied or was invalid".

| TITLE | DETAILS |
|--------------------------------|--|
| Component | Web API |
| SDL Phase | Build |
| Applicable Technologies | MVC5, MVC6 |
| Attributes | Identity Provider - ADFS, Identity Provider - Azure AD |
| References | Secure a Web API with Individual Accounts and Local Login in ASP.NET Web API 2.2 |

| TITLE | DETAILS |
|--------------|--|
| Steps | <p>If the Web API is secured using OAuth 2.0, then it expects a bearer token in Authorization request header and grants access to the request only if the token is valid. Unlike cookie based authentication, browsers do not attach the bearer tokens to requests. The requesting client needs to explicitly attach the bearer token in the request header. Therefore, for ASP.NET Web APIs protected using OAuth 2.0, bearer tokens are considered as a defense against CSRF attacks. Please note that if the MVC portion of the application uses forms authentication (i.e., uses cookies), anti-forgery tokens have to be used by the MVC web app.</p> |

Example

The Web API has to be informed to rely ONLY on bearer tokens and not on cookies. It can be done by the following configuration in `WebApiConfig.Register` method:

```
C-Sharp code config.SuppressDefaultHostAuthentication();
config.Filters.Add(new HostAuthenticationFilter(OAuthDefaults.AuthenticationType));
```

The `SuppressDefaultHostAuthentication` method tells Web API to ignore any authentication that happens before the request reaches the Web API pipeline, either by IIS or by OWIN middleware. That way, we can restrict Web API to authenticate only using bearer tokens.