Hardening Guide

Editorial and Asset Management Workflows

Microsoft Azure

Revision 3

July 2020

Independent Security Evaluators | www.ise.io

August 3, 2020 | *Authorized Distribution Only*

Executive Summary

Microsoft engaged Independent Security Evaluators (ISE) to evaluate multiple editorial and asset management workflows using the Microsoft Azure cloud computing environment. The primary goal of this evaluation was to establish an Azure-specific hardening guide for the media and entertainment (M&E) industry. This guidance is specific to MAM/DAM systems, cloud-based media editorial applications, and editorial data storage workflows.

The evaluated workflows included hybrid cloud editorial deployments, near-line editing storage systems, archive and long-term production content storage systems, and cloud-based editing systems with on-premises storage.

The M&E industry wishes to use Azure in its internal and vendors' software systems to increase the throughput, security, scalability, and cost-efficiency of its film production activities while improving the deployment's security posture. This document is meant for those who may be deploying editorial or asset management workflows using Microsoft Azure. **While Azure provides a variety of benefits, it is the responsibility of the deploying party to configure Azure properly to ensure that the security posture of the deployment is sufficient for asset protection.**

We performed hands-on evaluations of relevant Azure services and publicly available documentation to develop the security controls described in this document. This guide is current as of May 2020. Changes to Azure after this date may invalidate specific recommendations or introduce new concerns. Furthermore, users are responsible for understanding their cloud deployments and associated security risks.

Overall, we recommend that studios consider the security controls described in this document and perform independent deployment assessments of individual asset management systems in the future.

Revision	Date	Description
1	June 2018	Initial hardening guide.
2	May 2019	Update to the hardening guide with latest security concerns and recommendations.
3	July 2020	Added recommendations for additional Azure services.

Table of Contents

Securing Editorial and Asset Management Workflows

This section describes a data flow for asset management systems and editorial workflows as a unified data management workflow to illustrate how assets are ingested, processed, and stored. ISE provides a security stack and a reference architecture that administrators can utilize to deploy secure editorial and asset management systems using Azure services.

MAMS/DAMS, Editorial Workflows

Post-production teams are generally composed of the video storage, editorial, and art teams. These teams are responsible for performing various tasks that are required to produce creative content.

The following is a summary of the post-production teams' duties and functions.

VIDEO STORAGE TEAM

The video storage team is responsible for the following duties and functions:

- Ingest and management of original camera media, access metadata
- Continuous quality, data checking, and securing of content
- Color management and Lookup Table (LUT) creation
- Dailies creation and transcoding
- Linear Tape Open (LTO) archiving and redundancy checks

The video lab engineer is responsible for handling original camera media ingestion. A copy of the original camera media is stored on near-line direct storage servers. The video lab also processes the original media by performing color management using LUT files provided by a digital imaging technician (DIT) and original camera media in raw format.

EDITORIAL TEAM

The editorial team ensures consistent visual storytelling throughout the motion picture. The editorial team utilizes on-premises and/or cloud editing tools to create short clips for editorial and director reviews. The shorts are shared via cloud media sharing services and/or local asset management systems.

ART TEAM

The art team is responsible for the following duties and functions:

- Creation and management of previsualization art
- Working with VFX editors for consistency in VFX

The art team supports collaborative pre-visualization and post-visualization activities. This team develops and manages the concept artwork that drives the pre-visualization process. All creative artwork is stored in on-premises or cloud MAM/DAM data storage services. A digital asset manager is responsible for ingesting artwork from external sources and curating it on the server.

Below is an Azure cloud-based reference editorial workflow. In the depicted workflow, digital data assets are ingested into the cloud system on to specialized near-line storage. Editors remotely connect to editorial

clients. Edited media is then linked to MAM/DAM systems, and all raw and processed content is synced using long-term archival or project parking storage systems.

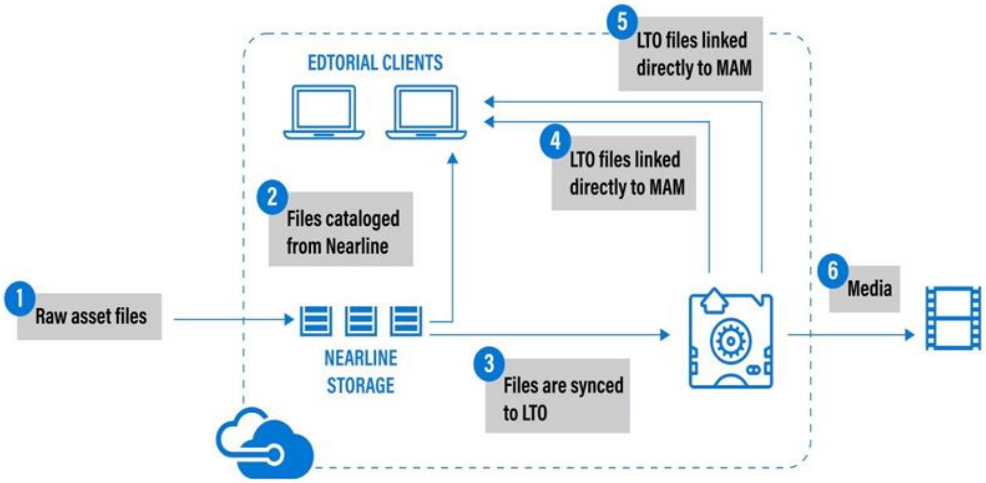

Figure 1. Azure-based editorial workflow.

Azure Reference Architecture

The Azure reference architecture includes provisions to extend an on-premises network and Active Directory (AD) environment to Azure. It consists of a perimeter network to enable secure connectivity between the on-premises network and an Azure virtual network (VNet). The architecture requires a connection to the on-premises datacenter, which can either be a VPN gateway or an ExpressRoute connection.

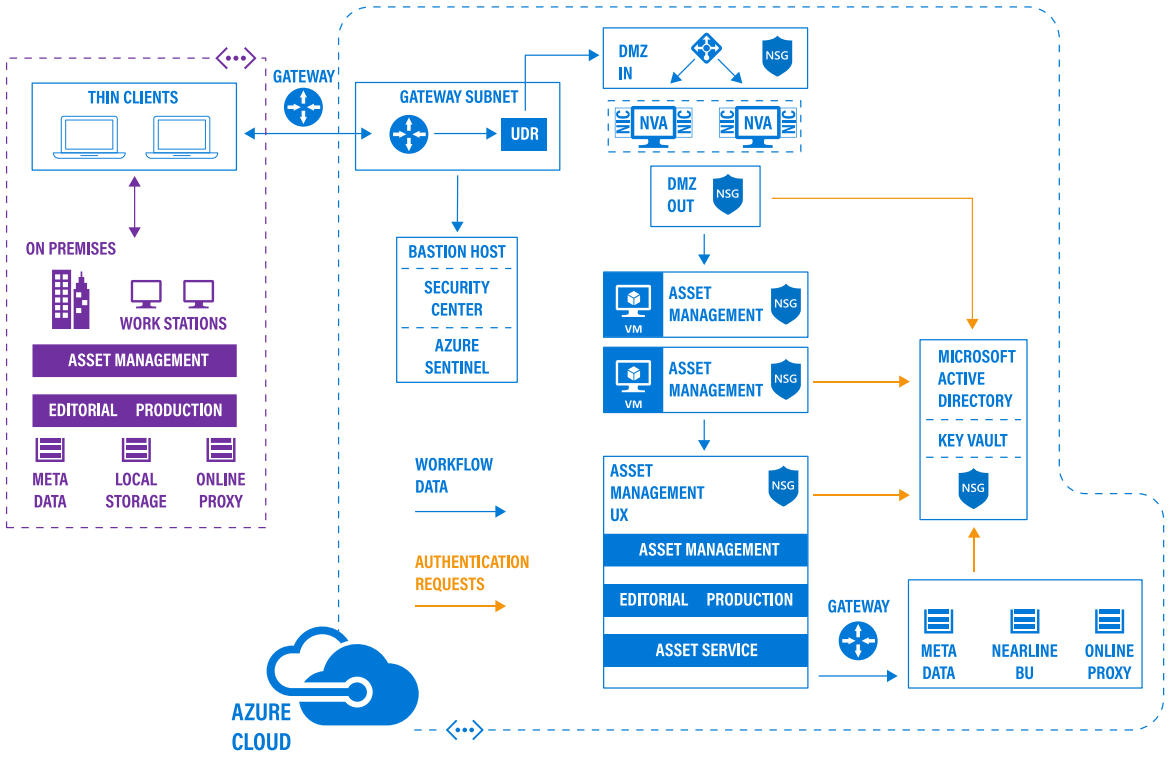

Figure 2. Azure-based MAM/DAM architecture.

KEY ELEMENTS

Below is a list of the key elements of the Azure-based MAM/DAM architecture depicted in Figure 2:

- **On-premises network**. A private local area networked implemented in an organization.
- **Azure Virtual Network (VNet)**. The VNet hosts the application and other resources running in Azure.
- **Gateway.** The gateway provides connectivity between the router in the on-premises network and virtual networking. User-defined routes handle routing for on-premises traffic that passes to Azure.
- **User-defined routes.** These rules define the flow of network traffic within VNet subnets.
- **Management subnet.** This subnet contains virtual machines that are used for managing and monitoring components running in the VNet.
- **Management bastion host.** A virtual machine on the network that administrators can use to connect to other virtual machines. The bastion host has a network security group (NSG) that allows remote SSH access from whitelisted IP addresses.

- **Active Directory (AD) servers.** Azure AD supports multiple authentication methods for hybrid identity solutions. A federated authentication method relies on an external trusted system, such as an on-premises Active Directory Federation Services (AD FS), to authenticate users. Federation with Azure AD enables users to authenticate using on-premises credentials and access all resources in cloud. Federated authentication is required when organizations have requirements that Azure AD does not support natively.
- **Active Directory subnet.** The AD Domain Services (DS) servers are hosted in a separate subnet. NSG rules protect the AD DS servers and provide a firewall against traffic from unexpected sources.
- **Web tier.** This section of the architecture is the web front end. Web pages are served from here.
- **Data Processing tier.** The business tier communicates with the web tier on one end and the data tier on the other. Business logic and processing takes place at this level.
- **Data Storage tier.** The data tier houses persistent data in a database within an asset management system.

Azure-based Security Stack

This section describes a security stack that provides guidance on how Azure services can be used to secure editorial and asset management workflows in Azure.

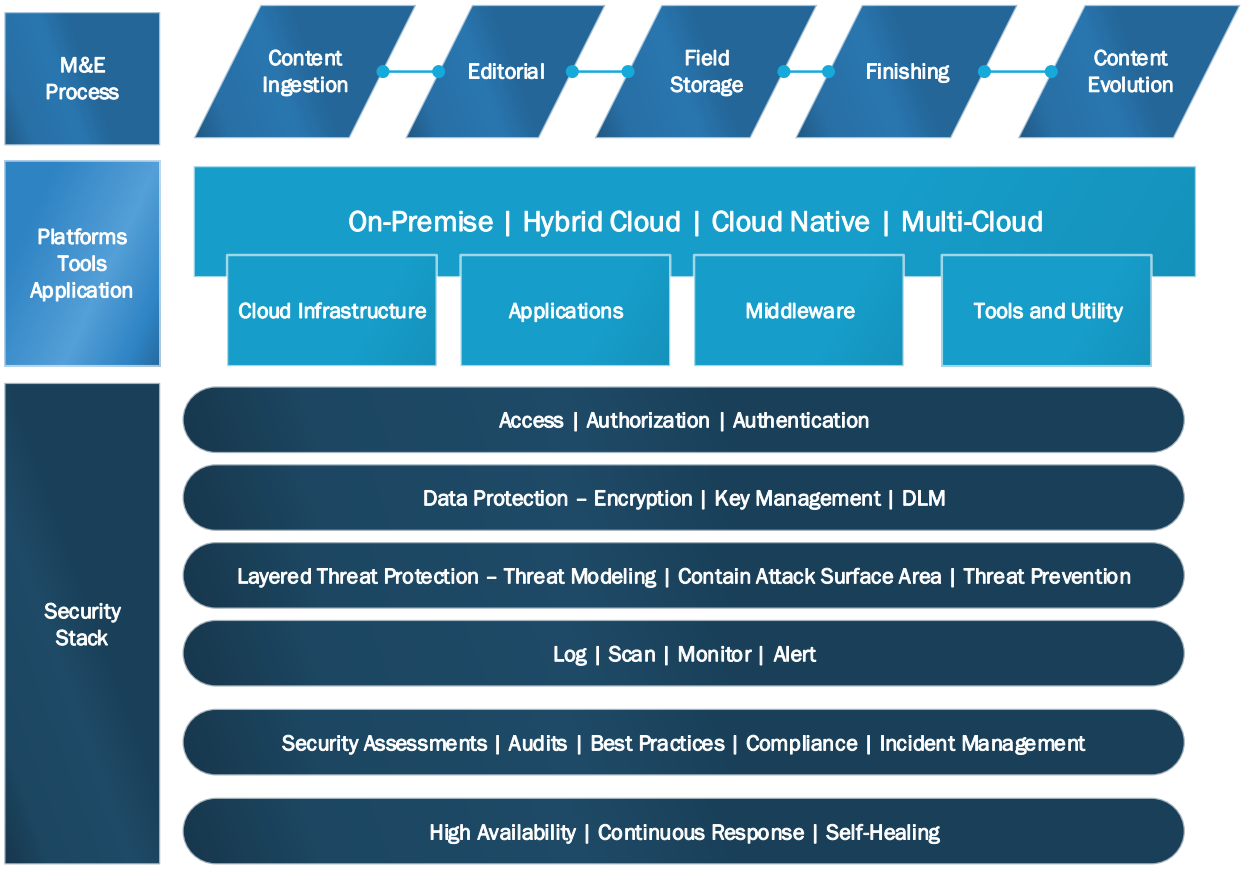

Figure 3. Azure security stack.

Authentication, Authorization, And Access control

Azure provides several services and resources that studios can use to implement authentication and authorization in Azure-based editorial and asset management workflows.

AZURE RBAC

Azure role-based access control (RBAC) can be used to manage access to Azure resources. Using RBAC, Azure administrators can segregate duties within various teams (e.g., video lab, editorial, and art teams) and grant only the minimal amount of access that they need to perform their job functions, upholding the principle of least privilege. Azure provides built-in roles; however, ISE recommends creating custom roles with more granularity to meet specific organizational requirements.

SSO/MFA/PASSWORD POLICIES

Use single sign-on (SSO) across multiple Azure accounts, enable multi-factor authentication (MFA), and define password policies and restrictions that prevent the usage of insecure passwords.

SUBSCRIPTION

Studios can use subscriptions to organize and manage assets in Azure-based editorial and asset management systems. It is crucial to design a cloud environment that reflects an organization's hierarchy and anticipated needs. For instance, each subscription could be treated as a container for a workflow, client, or production.

Data Protection

Azure offers various security features that studios can use to store data securely and process data for various editorial and asset management workflows.

ENCRYPTION

Data encryption at-rest and in-transit are essential parts of a data protection strategy. Azure provides multiples features to help safeguard customers' data. Azure Virtual Machines support encryption at-rest via the Azure Disk Encryption feature. Azure Storage and Azure SQL database encrypt data-at-rest by default, and many services offer encryption as an option.

Protecting data in transit is important because data can traverse through various locations on the public internet. Azure ExpressRoute and Azure VPN gateway provides dedicated private communication channels between on-premises datacenters to Azure for sensitive data such as pre-release content.

DATA CLASSIFICATION

Data classification allows an organization to categorize data by sensitivity and business impact, which is vital in identifying risks. Studios can use Azure Information Protection to classify, label, and protect their organization's data. Similarly, the Data Discovery & Classification for Azure SQL Database provides advanced capabilities for discovering, classifying, labeling, and reporting sensitive data in databases.

AUTOMATED KEY MANAGEMENT

Use Azure Key Vault to securely store and manage cryptographic keys, secrets, and certificates.

DATA LIFECYCLE MANAGEMENT (DLM)

Setup data storage accounts with automatic backups and snapshots.

Layered Threat Protection

Microsoft provides various Azure services and resources that studios can use to implement a multi-layered approach to security.

NETWORK SECURITY

Network Security Groups (NSGs) secure both inbound and outbound access to Azure virtual machines (VMs) and Azure virtual networks (VNets), like a traditional firewall. Studios should use NSGs to restrict access to assets to the highest degree possible.

While NSGs and user-defined routes provide security at the network and transport layers of the OSI model, organizations may want or need to enable security at higher levels of the stack. In such situations, ISE recommends deploying virtual network security appliances (e.g., firewall, WAN accelerator) provided by Azure partners. Network security capabilities of virtual network security appliances include firewalling, intrusion detection and prevention, and vulnerability management.

Azure supports multiple methods for securely connecting on-premises networks to the cloud for hybrid network architectures. Studios could use services like Azure VPN Gateway, ExpressRoute, and Bastion to create secure connections from on-premises systems to Azure resources.

CONTENT DELIVERY NETWORK (CDN) SECURITY

A CDN is a distributed network of servers that provides efficient delivery of web content to users. CDNs store cached content on the edge server in the point-of-presence (POP) location that is close to end-users to minimize latency. Azure provides multiple CDN features that would help secure customers' data.

The Azure Web Application Firewall (WAF) on Azure CDN provides global and centralized protection for web content by defending web services against common exploits and vulnerabilities. Azure CDN also supports the use of geo-filtering and the Standard rules engine to restrict access to content.

THREAT PREVENTION

Azure Advanced Threat Protection leverages on-premises Active Directory signals to identify, detect, and investigate advanced threats, compromised identities, and malicious insider actions directed at an organization.

Azure Sentinel is a security information event management (SIEM) and security orchestration automated response (SOAR) solution that studios can use to alert detection, threat visibility, proactive hunting, and threat response.

Azure DDoS protection provides defense against distributed denial of service (DDoS) attacks with always-on monitoring and automatic network attack mitigation. Azure provides two DDoS service offerings – basic and standard. It uses dedicated traffic monitoring and machine learning algorithms to develop protection policies that are tuned specifically to Azure Virtual Network resources.

Log, Scan, Monitor, and Alert

Azure provides a wide range of configurable auditing and logging options to help customers identify and troubleshoot security and operational issues within various Azure services and resources.

TAGGING

Studios should apply tags to various Azure resources to logically organize resources based on their associated workflow, client, or production.

SCANNING

Microsoft Antimalware for Azure is real-time protection that provides a wide range of features that helps with identifying and removing malicious software. It generates alerts when known malicious or unwanted software attempts to install itself or run on an Azure-based system. Azure provides customers the ability to use the default anti-malware configuration settings or use custom configuration settings for workload-specific requirements.

INLINE DATABASE MONITORING

Studios should use a database monitoring solution to detect operational and security issues within databases. Streaming metrics and resource logs shipped to Log Analytics workspace in Azure Monitor provide system administrator a data-driven operational view of the cloud resources. Streamed data can then be analyzed with other monitoring data and enables customers to leverage other Azure Monitor features such as alerts and visualizations.

LOG ANALYTICS

Azure Monitor is a comprehensive solution for collecting, analyzing, and acting on telemetry from the cloud and on-premises environments. Log Analytics can be used to write, execute, and manage Azure Monitor log queries in the Azure Portal. Log Analytics queries capabilities such as the search for terms and identification of patterns can lead to useful insights from the collected data.

SECURE BASELINE

A secure baseline ensures that technical requirements and security constraints are consistently applied to a cloud environment, as requirements mature. Azure provides documentation[1] on tools that can be used to develop and maintain a secure baseline.

Security Assessments, Audits, Best Practices, Compliance, Incident Management

There are various Azure services and resources that studios can utilize to improve the overall security posture of Azure-based editorial and asset management systems.

VULNERABILITY ANALYSIS

Azure Security Center is an infrastructure security management system that collects and processes security-related data to help customers prevent, detect, and respond to threats. Security Center aggregates the collected data and generates a secure score that reflects the specified environment's security situation. The Security Center identifies potential security vulnerabilities; it creates recommendations to remediate the discovered vulnerabilities.

COMPLIANCE

Enforcing governance is a common challenge faced by organizations within their IT environment. Governance provides mechanisms and processes that are used to ensure than an environment conforms with specific design, regulatory, and security requirements. As organizations create policies and plan governance strategies, it is important that they have the appropriate tools to evaluate and enforce configuration in their IT environments. Azure provides tools and services like Azure Policy and Azure Blueprints to help organizations develop and enforce governance processes.

System administrators can use Azure Policy to enforce organizational standards and assess compliance. Azure Policy determines if a resource is compliant or not by comparing its properties to business rules. Business rules are outlined in policy definitions and initiatives, which are then assigned to any scope of resources that Azure supports (e.g., management groups, subscriptions). This ensures that all Azure resources comply with specific organizational standards and requirements.

Azure Policy can also be used in conjunction with Azure Blueprints. Azure Blueprints is a service designed to help with environment setup. With Blueprints, cloud architects and system administrators have the ability to create a blueprint or a package of repeatable set of Azure resources that adheres to specific organizational requirements.

A blueprint is composed of artifacts which may include resource groups, ARM templates, policy assignments, and role assignments. Including a policy in a blueprint makes it possible to create the correct design during blueprint assignment. Moreover, it ensures that only approved changes can be made to the environment to which the blueprint was assigned, which ensures ongoing compliance with the blueprint.

Although Azure Policy can be used in conjunction with Azure Blueprints, it is important to note the main differences between the two services. Azure Policy focuses on the evaluation of resource properties during deployment and for existing resources. It supports governance by validating that resources within an

[1] https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/govern/security-baseline/toolchain

environment adhere to specific standards and requirements. On the other hand, Azure Blueprints is a service that allows cloud architects to create a repeatable sets of Azure resources that comply with specific requirements, which ultimately speeds up the development and delivery of Azure-based solutions while maintaining consistency and compliance.

INCIDENT MANAGEMENT

Developing and implementing an incident response plan is an essential component of protecting an organization's information. Azure Security Center can be used for incident scoring and prioritization procedures. Specifically, Security Center assigns a severity to each security alert, which helps system administrators prioritize which alerts should be investigated first. Additionally, using tags and creating a naming system to identify and categorize Azure resources will help in prioritizing the remediation of alerts that could be associated with business-critical functions.

High Availability, Continuous Response, Self-healing

Azure provides a variety of tools and mechanisms that help customers build highly available, fault-tolerant systems.

REDUNDANCY

An Azure region is a set of data centers deployed within a latency-defined perimeter and connected through a dedicated regional low-latency network. Availability Zones are unique physical locations within an Azure region. Customers can synchronously replicate applications and data using Availability Zones within an Azure area to ensure high availability. Replication of data across Azure regions is vital for disaster recovery protection.

AUTOSCALING

Autoscaling is the process of dynamically allocating resources to match performance demand. Azure provides built-in autoscaling mechanisms for some services (e.g., virtual machine scale sets). A custom autoscaling implementation is recommended if a specific service does not have a built-in autoscaling functionality, or if the customer requires specific autoscaling requirements.

DISASTER RECOVERY

It is critical for organizations to adopt a business continuity and disaster recovery (BCDR) strategy to ensure data is protected and services continue to run during planned and unplanned outages. The Azure Site Recovery service provides various mechanisms that can be used for a BCDR strategy, including VM and workload replication. The Azure Backup service offers simple and secure solutions for data backup and recovery.

Recommendation-TPN-Service Mappings

The following table maps ISE's recommendations to a draft version (current as of June 2020) of the Trusted Partner Network's (TPN) control categories and a suggested Azure service to fulfil that recommendation.

ISE Recommendation	TPN Category	Azure Control
Extend Federated Identity Management for Access Control	Identity & Authentication	Azure Active Directory
Use Azure Role-based Access Control	Access Control	Azure Active Directory
Use Conditional Access Policies	Access Control	Azure Active Directory
Utilize the Custom Banned Password List	Access Control	Azure Active Directory
Enable MFA	Access Control	Azure Active Directory
Restrict Access to Bastion Hosts	Access Control	Azure Bastion
Utilize Logging features	Auditing & Logging	Azure Bastion
Ensure Updated Node Agents are Used	System Integrity	Azure Batch
Isolate Jobs in Separate Batch Pools	Production Specific Controls	Azure Batch
Deploy Preconfigured Environments	System Integrity	Azure Blueprints
Avoid Caching Session Information	Access Control	Azure Command Line Interface
Restrict Access to Azure Container Registries	Access Control	Azure Container Registry
Ensure Updated Container Images are Used	System Integrity	Azure Container Registry
Enable Logging for Azure Container Registries	Auditing & Logging	Azure Container Registry
Create Separate Container Registries for Productions	Production Specific Controls	Azure Container Registry
Avoid Transferring Sensitive Data over the Public Internet	Data Protection	Azure Express Route
Use Separate Azure Key Vaults for Productions	Access Control	Azure Key Vault
Restrict Access to Azure Key Vaults	Access Control	Azure Key Vault
Utilize Logging Feature	Auditing & Logging	Azure Key Vault
Periodically Rotate Keys	Access Control	Azure Key Vault
Use Separate Azure Storage Accounts for Media Service Accounts	Access Control	Azure Media Services
Configure Live Media Archiving Policy	Production Specific Controls	Azure Media Services
Limit and Control Data Sources	Data Protection	Azure Monitor
Restrict Access to Log Analytics Workspaces	Access Control	Azure Monitor
Create Separate Log Analytics Workspaces for Productions	Production Specific Controls	Azure Monitor
Define Policies that Ensure Compliance	Network Security	Azure Policy
Use Enterprise Accounts for Administration	Identity & Authentication	Azure Portal
Use Separate Azure Subscriptions to Segregate Work	Identity & Authentication	Azure Portal
Use Azure Resource Groups	Security Planning	Azure Portal

Disable Non-SSL Connections	Cryptographic Controls	Azure Redis Cache
Monitor Redis Cache Performance	System Integrity	Azure Redis Cache
Define Custom Security Policy	Security Planning	Azure Security Center
Define a Security Response Plan	Security Awareness	Azure Security Center
Limit Security Center Data Collection	Data Protection	Azure Security Center
Use Separate Azure SQL Database Instances	Production Specific Controls	Azure SQL Database
Enable Advanced Threat Protection	System Integrity	Azure Storage
Enable Encryption at Rest	Cryptographic Controls	Azure Storage
Periodically Rotate Access Keys	Access Control	Azure Storage
Protect Assets with Digital Rights Management (DRM)	Data Protection	Azure Storage
Require Secure Transfers	Data Protection	Azure Storage
Use Shared Access Signatures to Access Storage Account Resources	Access Control	Azure Storage
Enable Soft Delete	Data Protection	Azure Storage
Configure Firewall Rules	Network Security	Azure Storage
Use Private Endpoints	Network Security	Azure Storage
Enable Storage Logging	Auditing & Logging	Azure Storage
Ensure Disk Storage is Encrypted at Rest	Cryptographic Controls	Azure Virtual Machines
Ensure Virtual Machines are Patched Regularly	System Integrity	Azure Virtual Machines
Use Public Key Authentication for Virtual Machines	Identity & Authentication	Azure Virtual Machines
Use Hardened OS Images for VM Instantiation	Change & Config Management	Azure Virtual Machines
Apply a Multi-tiered Architecture for VNets	Network Security	Azure Virtual Networking
Use Service Endpoints	Network Security	Azure Virtual Networking
Avoid Deprecated Cryptography for IPSec VPNs	Cryptographic Controls	Azure Virtual Networking
Configure Network Watcher-Configure Connection Monitor	System Integrity	Azure Virtual Networking
Configure Network Watcher-Configure Flow logs	Auditing & Logging	Azure Virtual Networking
Create Separate Virtual Networks for Productions	Production Specific Controls	Azure Virtual Networking
Enable DDos Protection	System Integrity	Azure Virtual Networking
Isolate Virtual Appliances Using Individual Subnets	Network Security	Azure Virtual Networking
Secure Connections to Azure Networks	Network Security	Azure Virtual Networking
Use Azure Application Gateway	Network Security	Azure Virtual Networking
Use Network Security Groups (NSG)	Network Security	Azure Virtual Networking

Azure Security Controls

Security controls, as outlined in the executive summary, are presented here for Azure-based editorial and asset management systems. They are grouped according to each specific Azure component with a loose relationship back to industry best practice guidelines as provided by the Motion Picture Association (MPA).

These Azure-specific security controls consider the high-level architecture of the system within its operational context and offer recommendations for the hardening of that system. A detailed description of each security control, recommendation, and where to find further documentation for each follows in this section.

Azure Active Directory

Extend On-premises Identity Management for Access Control

Managing identity is as vital in Azure as it is on-premises. Studios use on-premises Active Directory (AD) systems to store directory data and manage communication between users and resources, including user logon processes, authentication, and directory searches. When scaling out editorial and asset management workflows to Azure in a hybrid deployment, the cloud resources are being used as an extension of the on-premises datacenter—in this scenario, and there are applications that require a domain controller to handle authentication and authorization.

Recommendation: Use the Azure Active Directory Service

Use an Active Directory service in the cloud. The Windows Server AD is running in VMs created using Azure Virtual Machines, and the AD VMs should be grouped into a virtual network connected to an on-premises datacenter using the Azure Virtual Network.

The virtual network carves out a group of cloud virtual machines that interact with the on-premises network via a virtual private network (VPN) connection, which allows the AD Azure virtual machines to look like just another subnet to the on-premises datacenter.

Documentation: Information about Azure Active Directory is available at: https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-whatis.

Use Azure Role-based Access Control

In many cases, studios have multiple parallel workflows for various productions and vendors involving a myriad of artists and resources. Role-based access control (RBAC) can be used to manage access to Azure resources. Using RBAC, Azure administrators can segregate duties within various teams (e.g., video lab, editorial, and art teams) and grant only the minimal amount of access that they need to perform their job functions, upholding the principle of least privilege.

Recommendation: Use Custom Roles

Administrators should use Azure RBAC to manage access to Azure resources. Azure provides built-in roles; however, ISE recommends creating custom roles to meet specific organizational needs. Custom roles can be assigned to users, groups, and service principals at the management, subscriptions, and resource group levels. Note that custom roles can be shared between subscriptions that trusts the same Azure AD directory.

Documentation: Information about custom roles is available at: https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles

Use Custom Banned Password List

Compromised user passwords are a common cause of breaches and security incidents. Often, the source can be traced to an account using a weak password. To help mitigate this risk, administrators can set password policies and restrictions that prevent the usage of insecure passwords. Azure supports two forms of banned password lists that can aid this goal: a global banned password list that uses a Microsoft-curated compilation of exposed passwords, and a custom banned password feature. The custom list allows administrators to ban passwords containing words and phrases that may be easy to guess, such as the company's name.

Recommendation: Use the Custom Banned Password list

Azure administrators should configure a custom banned password list that blacklists easily guessable words or phrases. Consider the following suggestions:

- Company name and/or initials
- City, state/province, the country where the company is located
- Product and project names
- Names of software used in Azure environment

Documentation: Information about banned password lists is available at: https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-password-ban-bad.

Use Conditional Access Policies

Azure AD security defaults contain preconfigured security settings to help protect Azure resources from frequent attacks. While security defaults provide an easy way to secure an Azure environment, administrators may need to configure policies with more granularity to meet organizational requirements. For instance, administrators may need to exclude specific accounts (e.g., emergency access account) from policies requiring multi-factor authentication (MFA). Common policies deployed by organizations, including blocking legacy authentication and requiring MFA for administrators and Azure management tasks.

Recommendation: Implement Conditional Access Policies

Azure administrators should implement Conditional Access policies to meet specific organizational requirements. For instance, studios may require users to use Hybrid Azure AD joined devices. These devices are joined to an on-premises AD and registered with Azure AD. Conditional access policies ensure that users are accessing Azure resources from devices that meet security and compliance standards.

Documentation: Information about Conditional Access is available at: https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/overview.

Enable MFA

MFA is a process where a user is required to provide an additional form of identification during a sign-in event. MFA improves account security because attackers cannot quickly obtain the additional factor of authentication. Security defaults in Azure AD enable the use of the Microsoft Authenticator app[2] for all users.

[2] https://docs.microsoft.com/en-us/azure/active-directory/user-help/user-help-auth-app-overview

It is recommended to enable and use Azure MFA with Conditional Access policies. These policies can be used to define specific events or applications that require MFA for more granular controls.

Recommendation: Enable MFA

Azure administrators should enable and configure MFA using Conditional Access policies. We recommend enabling MFA for all users, especially for administrative accounts that generally have unrestricted access to various Azure services.

Documentation: Information about enabling and configuring MFA is available at: https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-mfa-howitworks.

Azure Bastion

Restrict Access to Bastion Hosts

Bastion hosts provide an entry point into a network and are typically used to connect to resources that have restricted access from public networks. As these machines have a critical role in network security, access to bastion hosts must be secured.

Recommendation: Limit Access to Bastion Hosts

Access to Azure Bastion hosts should be restricted to only users who require this access. Furthermore, firewall restrictions can be used to limit access to certain known IP addresses, such as the addresses for employee offices.

Utilize Logging Features

Proper logging can enable administrators to track who accessed what systems and is a vital piece of any audit strategy. Azure Bastion supports a feature called Diagnostic Logging that stores information about the users who accessed the service.

Recommendation: Enable Diagnostic Logging

Diagnostic logging should be enabled, and suitable retention policy for the logs should be configured. Diagnostic logging may allow administrators to investigate a security incident better if the attacker uses the Azure Bastion service during the attack.

Documentation: Information about Azure Bastion's logging capabilities can be found at: https://docs.microsoft.com/en-us/azure/bastion/diagnostic-logs.

Azure Batch

Isolate Jobs in Separate Batch Pools

Azure Batch jobs within different pools are unable to view or communicate with one another. This may be useful if processing must occur in a manner that requires isolation.

Recommendation: Place Jobs Requiring Isolation in Separate Pools

Jobs performing processing that could benefit from isolation should be placed in separate batch jobs. For example, processing on media assets can be separated by production.

Ensure Updated Node Agents are Used

The Batch node agent is a software installed on virtual machines used in Batch jobs that provides an interface for controlling the virtual machine. Periodically, new releases of the node agent are released that contain new features and bugfixes.

Recommendation: Update Node Agents

Node agent updates can be applied by periodically reducing the size of pools to zero compute units. Administrators should back up any log files before performing this operation. Companies should document and implement a policy to perform this action on a regular cadence.

Documentation: Microsoft provides documentation on this procedure and others in their best practices guide for Azure Batch: https://docs.microsoft.com/en-us/azure/batch/best-practices.

Azure Blueprints

Deploy Preconfigured Environments

Azure Blueprints is a service that offers several pre-defined sample configurations that adhere to various compliance and regulatory standards. These sample configurations consist of multiple artifacts that may include a set of resource groups, policies, role assignments, and ARM template deployments. Free-form deployment of resources can often lead to configuration errors and may result in security vulnerabilities. As such, cloud architects should consider using premade blueprints to develop configuration baselines and best practices for securely deploying their environment.

Recommendation: Utilize Blueprint Samples

Organizations often use resources that are repeatedly deployed in similar configurations to meet organizational requirements. The sample configurations provided by Azure Blueprints offer prescriptive guidance for establishing a secure baseline system configuration. Cloud architects can use Blueprints to quickly build and deploy environments with resources that adhere to various compliance and regulatory standards.

For instance, Azure provides a Blueprint designed for Motion Pictures Association (MPA) compliance called "MPAA Audit." The MPAA Audit Blueprint has several artifacts that require security-enhancing features of other Azure services to be enabled. Some artifacts are focused on Windows virtual machines, and SQL Server databases and customers using other platforms will need to add additional artifacts that correspond to their technology stack. MPAA Audit also features several parameterized artifacts that require system administrators to configure secure values manually.

Note that system administrators should manually review Blueprint templates before they are used to ensure that the artifacts being applied are appropriate for their environment.

Documentation: The MPAA Audit Blueprint, and other premade Blueprints, can be found at https://docs.microsoft.com/en-us/azure/governance/blueprints/samples/.

Azure Command Line Interface

Avoid Caching Session Information

One way that users authenticate to Azure is by using a Microsoft account email address and password. To prevent a user from needing to retype credentials upon every invocation of the command line utility, the utility persistently caches the user's OAuth session information (excluding passwords) on the local system. Caching session information poses a security risk because an attacker who gains access to the cached session information could use it to gain unauthorized access to the user's account.

Recommendation: Delete Credentials at Session Termination

Azure administrators should implement a policy that mandates the deletion of credentials at session termination.

Recommendation: Use RAM Disk for Credential Storage

Using a RAM disk for credential storage ensures that credentials are never written to disk. Specifically, credentials are only stored in volatile memory and deleted upon unmounting the RAM disk or rebooting the machine. To use a RAM disk, administrators should first create a RAM disk then, create a symbolic link from the credentials file to a file located on the RAM disk. This can be achieved using the Linux `tmpfs` file system, or using analogous techniques on Windows or Mac OS X.

Azure Container Registry

Restrict Access to Azure Container Registries

Azure Container Registries (ACR) store container images that can then be deployed using technologies such as Docker. As the registries may house assets containing proprietary or sensitive data, regulating access to ACRs is an essential aspect of securing container-based workflows. ACR supports the use of firewall rules and Azure RBAC to control access to registries.

Recommendation: Use Firewall Rules to Control Access to ACR

Firewall rules should be created to restrict access to Azure Container Registries. When possible, access should be restricted to whitelisted IP addresses or networks.

Recommendation: Use Azure RBAC to Control Access to ACR

RBAC provides Azure-built in roles that allow administrators to grant different levels of permissions to Azure resources. ACR supports a set of built-in Azure roles that can be used to assign specific permissions to users, service principals, or other identities that need access to ACRs.

Documentation:

- Microsoft provides documentation on ACR firewalls at: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-firewall-access-rules and https://docs.microsoft.com/en-us/azure/container-registry/container-registry-vnet.
- Microsoft provides documentation on using RBAC for restricting access to ACRs at: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-roles

Enable Logging for Azure Container Registries

Azure administrators can use Azure Monitor[3] to collect resource logs for user-driven events in registries. Information contained in the resource logs can be monitored and reviewed to audit registry authentication events to ensure security and compliance. In addition, resource logs provide an activity trail on registry artifacts (e.g., push and pull events), which is useful for detecting and troubleshooting operational issues within a registry. Note that Azure Monitor's Log Analytics Workspace also allows administrators to enable alerts for anomalous activities related to registries.

Recommendation: Enable Logging for Azure Container Registries

Resource logs should be collected for ACRs to help administrators detect and troubleshoot operational and security issues within registries.

Documentation: Microsoft provides documentation on ACR logs at: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-diagnostics-audit-logs.

Create Separate Container Registries for Productions

Create separate ACRs for each production if container-based editorial and asset management workflows for each production requires isolation.

Recommendation: Create Separate Container Registers for Productions

Azure administrators should create separate ACRs for each production. Resources storing or processing sensitive assets should be sufficiently isolated.

Ensure Updated Container Images are Used

Dockerfiles used to create container images specify a parent image referred to as its base image. A base image generally contains the operating system, on which the rest of the container's layers are applied. An image maintainer is responsible for updating a base image to include new features or security patches. ACR Tasks provides the ability to automatically build images when a container's base image is updated.

Recommendation: Deploy Automated Patch Management Solution

Azure administrators should utilize ACR tasks to automate OS and software patching workflows. ACR tasks can detect when an application image's base image is updated. Individually, ACR tasks dynamically discover base image dependencies when it builds a container image. A preconfigured build task can be used to automatically rebuild every application image that references the updated base image.

Documentation: Microsoft provides documentation on base image updates for ACR tasks at: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-tasks-base-images.

[3] https://docs.microsoft.com/en-us/azure/azure-monitor/overview

Azure ExpressRoute

Avoid Transferring Sensitive Data over the Public Internet

While modern protocols with transport-layer encryption provide robust security that ensures the confidentiality and integrity of data in transit, for an additional level of security, consider transferring sensitive assets solely over ExpressRoute, a dedicated connection to Azure that does not use the public Internet.

Recommendation: Use ExpressRoute to Transfer Sensitive Data

Rather than using connections to Azure resources that travel over the public Internet and may be exposed to adversaries with privileged network access, utilize an ExpressRoute connection between on-premises data centers and Azure to transfer sensitive data.

For connections to Azure services from within an Azure VNet, consider using Azure Private Endpoints[4], which similarly avoid transferring data across the public Internet.

Azure Key Vault

Use Separate Azure Key Vaults for Productions

Though Azure Key Vault is a generic container for keys and secrets, it should not span across multiple disjoint productions. Each production has a defined lifecycle with varying priority, protection, and collaboration requirements. Production-specific key vaults should be used to limit access to protected assets associated with the product lifecycle.

Recommendation: Use Separate Key Vaults for Productions

Using a separate key vault for each production provides simplified administration of production-specific secrets. Each Key Vault will consist of a collection of cryptographic keys and cryptographically protected secrets logically bundled together for a specific production or workflow.

Documentation: Additional information about Azure Key Vault is available at: https://docs.microsoft.com/en-us/azure/key-vault/general/overview.

Restrict Access to Azure Key Vaults

Access to a key vault is controlled via two interfaces, the management plane and the data plane. That management plane is used for creating and deleting key vaults, retrieving key vault properties, and updating access policies. The data plane is used for adding, deleting, and modifying data stored in a key vault. Both planes use Azure AD for authentication. For authorization, the management plane uses Azure RBAC, and the data plane uses key vault access policies. Azure administrators should ensure that roles and access policies uphold the principle of least privilege.

[4] https://docs.microsoft.com/en-us/azure/private-link/private-endpoint-overview

Recommendation: Use Azure RBAC

Azure provides several predefined roles to help administrators control access to key vaults. Custom roles can be created if the predefined roles do not fit specific organizational needs,

Recommendation: Use Key Vault Access Policies

Azure administrators can grant data plane access by implementing access policies for a key vault. Note that the `Contributor` permissions for the management plane for the specified key vault are needed to set access policies.

Documentation: Additional information about securing access to Azure Key Vault is available at: https://docs.microsoft.com/en-us/azure/key-vault/general/secure-your-key-vault.

Utilize Logging Feature

Proper logging can enable administrators to track who accessed what systems and are a vital piece of any audit strategy. Azure Key Vault stores information about the activities performed on key vaults such as the creation or deletion of key vaults, keys, and secrets.

Recommendation: Enable Logging

Logging should be enabled, and a suitable retention policy for the logs should be configured. Logging may allow administrators to investigate a security incident better if the attacker uses key vaults service during the attack.

Documentation: Information about Azure Key Vault logging is available at: https://docs.microsoft.com/en-us/azure/key-vault/general/logging.

Periodically Rotate Keys

Applications can offload the storage of keys and secrets to Azure Key Vault, as needed. This centralized approach allows administrators to update keys and secrets without affecting the behavior or structure of applications. Best practices discourage the extended use of access keys as the long duration of validity increases the time an attacker could use a compromised key.

Recommendation: Implement Key Rotation

There are various options for implementing a key rotation strategy for values stored as key vault secrets. Though secrets can be rotated as part of a manual process, secrets should be rotated programmatically using API calls or automation scripts. Azure administrators are encouraged to consult the guidance in NIST SP 800-57 to create a baseline key management policy.

Documentation:

- Key Rotation using Azure Automation: https://docs.microsoft.com/en-us/azure/key-vault/secrets/key-rotation-log-monitoring.
- NIST recommendations for Key Management: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf.

Azure Media Services

Use Separate Azure Storage Accounts for Media Service Accounts

When a user creates an Azure Media Services (AMS) account, they either use an existing storage account or create a new storage account in the same region.

Recommendation: Use Separate Storage Accounts for Media Accounts

Use separate storage accounts for media accounts to logically separate media assets. Assets in the storage account associated with the Media service will likely be shared with an audience or processed for a specific purpose. The content of the storage account associated with the media should only share a limited set of data that the user has explicitly marked for processing or sharing. The storage should be protected with guidance provided earlier in this document.

Documentation: Azure Media account setup is described here: https://docs.microsoft.com/en-us/azure/media-services/latest/storage-account-concept.

Configure Live Media Archiving Policy

Azure Media Services offer live streaming of events directly from a device. While streaming the event, the service can be configured to archive content in a storage account as it is encoded and streamed live.

Recommendation: Define Content-Specific Archive Policy

Define a content-specific archiving policy for streamed content. In the media production environment, each asset has its own lifecycle, which should be mirrored in its archive policy.

Azure Monitor

Limit and Control Data Sources

The azure monitor is used to collect massive amounts of data from a large set of data sources. Data collected from the data sources may include personal data or data about the production, workflows, and clients. As such, Azure administrators should carefully set up and manage the data that is pushed out to Azure Monitor.

Recommendation: Limit Data Sources

Data from various Azure resources and services should be carefully reviewed before being pushed to Azure Monitor. It is recommended that data from guest operating systems and Azure resources do not include any specific references to clients, workflow, or production. If possible, the data should be anonymized to prevent any accidental data exposure.

Documentation: Information about Azure Monitor data sources is available here: https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-sources.

Restrict Access to Log Analytics Workspaces

Azure Monitor stores log data in a Log Analytics workspace. Each workspace has its own data repository and configuration. Data sources and solutions are configured to store data in a particular workspace. Each

workspace can have multiple user accounts associated with it, and each user account can access multiple workspaces. The `Access control mode` setting in each workspace defines how permissions are determined for the workspace.

Recommendation: Use RBAC to Restrict Access to Workspaces

Administrators should use RBAC to define more granular control to data stored in workspaces. Azure allows administrators to define specific data types that are accessible only to a specific set of users. In addition, table access control can be implemented with Azure custom roles to grant access to specific tables in a workspace. Note that custom roles are applied to workspaces regardless of the user's access mode[5]. Specify only those permissions that are required to perform editorial and asset management workflows to uphold the principle of least privilege.

Documentation: Information about managing access to log data and workspaces is available here: https://docs.microsoft.com/en-us/azure/azure-monitor/platform/manage-access.

Create Separate Log Analytics Workspaces for Productions

Though a single Log Analytics workspace can be defined to collect, aggregate, and analyze log data from the entire subscription – this can lead to possible data comingling, incorrect metrics, and difficulty with managing/analyzing logs.

Recommendation: Create Separate Log Analytics Workspaces for Productions

Define and use separate workspaces based on the editorial and asset management workflows for each production.

Documentation: Information about creating Log Analytics workspaces available here: https://docs.microsoft.com/en-us/azure/azure-monitor/learn/quick-create-workspace.

Azure Policy

Define Policies to Ensure Compliance

Azure Policy allows systems administrators to create compliance checks to ensure resources in their Azure environment follow those rules. In order to maximize benefits from this service, policies should be created to ensure company and industry guidelines and best practices are followed.

Recommendation: Create Policies and Audit Compliance

Administrators should configure compliance policies that are tailored to their organization's editorial and asset management architecture and workflows. In addition, policies should be reviewed regularly to ensure that systems are in compliance with all policies.

Systems operators should also consider using Microsoft Compliance Manager[6] to facilitate implementing policies from a large number of governance and policy-making organizations.

[5] https://docs.microsoft.com/en-us/azure/azure-monitor/platform/design-logs-deployment#access-mode
[6] https://docs.microsoft.com/en-us/microsoft-365/compliance/compliance-manager-overview?view=o365-worldwide

Azure Portal

Use Enterprise Accounts for Administration

Access to Azure is possible through two types of Microsoft account (formerly known as Microsoft Live ID) and a work or school account, which is an account stored in either an on-premises Active Directory installation or Azure Active Directory (AD). Any Azure account can be used to set up and administer the Azure portal. If an employee uses a personal account to set up and manage an Azure account, then the employer may have difficulty gaining control of the account if the employee later resigns or is terminated. Further, employers are unable to maintain security policies on personal accounts or otherwise audit the accounts' security.

Recommendation: Use Enterprise Accounts for Administration

Azure administrators should use an enterprise account to create and manage Azure accounts. Using enterprise accounts is recommended because they can be centrally managed by the organization that issued them; they have more features than Microsoft accounts and are directly authenticated by the Azure AD service.

Use Separate Azure Subscriptions to Segregate Work

Though Azure subscriptions are a container for billing, they can also be used as a security boundary. Subscriptions can be used to limit who can access Azure services associated with that subscription. Each Azure subscription is associated with a single Azure Active Directory. The subscription itself governs access to and use of the Azure services that are subscribed to. The subscription administrator manages the subscription service through the Azure Portal.

Recommendation: Use Separate Azure Subscriptions to Segregate Work

It is recommended to use separate Azure subscriptions for each different production, client, and media creation workflow to limit access. Segregating workflows, in this way, will provide administrative infrastructure and individual resource monitoring.

Documentation: Information on Azure subscriptions is available at: https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/decision-guides/subscriptions/.

Use Azure Resource Groups

In Azure, most things can be considered a resource (e.g., virtual machines, virtual networks). Azure administrators can utilize resource groups to logically group related resources (e.g., storage accounts, virtual machines) that share a common lifecycle. For example, a virtual machine that depends on a specific storage account should be in the same Batch resource group. The resource group can include all the resources for the editorial and asset management workflows, or only those resources that you want to manage as a single entity.

Recommendation: Use Azure Resource Groups

Azure administrators should create and use resource groups for each editorial and asset management workflow to make it easier to manage them as a single unit. A resource group can contain resources from multiple regions if the resources belong to the same subscription. Resource groups can be managed via the Azure Portal.

Documentation: Information on Azure resource groups is available at: https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/manage-resource-groups-portal.

Azure Redis Cache

Disable Non-SSL Connections

Redis is designed to be accessed by trusted clients inside trusted environments, and natively does not support encryption for data-in-transit. The Azure-based Redis database cache service provides access via an SSL proxy for an additional layer of protection, while customers can configure the service to use the optional non-SSL based connection.

Recommendation: Disable Non-SSL Connections

Disable the non-SSL port to the Redis cache server to force users to use the SSL-based connection only.

Documentation: Best practices for Azure Cache for Redis is available at: https://docs.microsoft.com/en-us/azure/azure-cache-for-redis/cache-best-practices.

Monitor Redis Cache Performance

Redis is susceptible to attacks triggered by carefully selected inputs from external clients. For instance, an attacker could supply a set of strings, via a web form, that is known to hash to the same bucket into a hash table. Doing this will turn the O(1) expected time (average time) to the O(N) worst case, consuming more CPU than expected, and ultimately causing a denial-of-service. Both use cases are rare, an advanced attacked or a poorly developed application can trigger this type of attack.

Recommendation: Monitor Redis cache Instances

Azure administrators should monitor Redis cache instances using the Azure Monitor service. Azure Monitor provides cache metrics including cache hits and misses, the number of connected clients, and used memory and CPU. In addition, administrators can use Azure Monitor to set alerts when certain conditions are met.

Documentation: Information on using Azure Monitor for Azure Cache for Redis is available at: https://docs.microsoft.com/en-us/azure/azure-cache-for-redis/cache-how-to-monitor.

Azure Security Center

Define Custom Security Policy

Using Built-in policies, the security center can continuously assess the configuration of the resources to identify security issues and vulnerabilities. The built-in policies cover basic security concerns around resource health, malware, access, and availability – in order to provide targeted coverage of concerns, custom policies should be defined. These custom security policies should focus on specific editorial and asset management systems designs.

Recommendation: Implement Custom Security Policy

Use workflow-specific security policy to measure security compliance. General security compliance is a good start in gauging security posture, but it can lead to a false sense of security if not augmented with complete workflow-specific security policies.

Documentation: Information on creating custom Azure security policies is available at: https://docs.microsoft.com/en-us/azure/security-center/custom-security-policies.

Administrators may also benefit from using Azure Blueprints to create environments using secure configurations.

Limit Security Center Data Collection

Security Center collects and processes security-related data, including configuration information, metadata, event logs, crash dump files, and more to prevent, detect, and respond to threats. Azure Security Center sources data from Azure services, network traffic, partner solutions, and virtual machines/servers. Though the data is kept logically separate from data access controls, there is a large amount of contextual data that can provide sensitive information about the workflow or production or even possibly the content being produced. The user should understand and limit the data being provided security center, continuously remove the data stored by the security center, and validate the security center data (e.g., crash dumps, alert data) in log analytic location.

Recommendation: Define Data Location

Use separate log analytics workspaces for securely storing security telemetry data from various subscriptions, workloads and applications.

Documentation: Information on Azure Security Data Collection workspaces can be found here: https://docs.microsoft.com/en-us/azure/security-center/security-center-enable-data-collection

Recommendation: Validate the Diagnostic Data Purge

According to Azure documentation, the Azure Security Center collects ephemeral copies of your crash dump files and analyzes them for evidence of exploit attempts and successful compromises. Administrators should validate this data is purged periodically and removed.

Documentation: Information on Azure Security Policy can be found at: https://docs.microsoft.com/en-us/azure/security-center/security-center-info-protection-policy.

Define a Security Response Plan

Security Center provides great insight on possible vulnerabilities and threats; however, the data must be reviewed and acted upon proactively. Many organizations define an incident plan after an actual attack has occurred, reducing their preparedness for the initial incident.

Recommendation: Create an Incident Response Plan

Define a custom incident response plan based on the threat model, deployment architecture, and workflow type. The plan should include evidence collection, data process, analysis, assessment, updates, and conclusions.

Documentation: Microsoft has created guidance on creating an incident response plan here: https://info.microsoft.com/rs/157-GQE-382/images/EN-US-CNTNT-emergency-doc-digital.pdf.

Azure SQL Database

Use Separate Azure SQL Database Instances

An Azure subscription can be used to create multiple database instances. Each database instance should be assigned to a high-level business unit such as a production, client, or workflow project.

Recommendation: Use Separate Database Instances

ISE recommends instantiating databases in a separate subscription for each production, client, or editorial and asset management workflow. Logically grouping databases through subscriptions will provide administrative autonomy and individualized resource monitoring.

Documentation: An overview of Azure SQL database security capabilities is available at: https://docs.microsoft.com/en-us/azure/sql-database/sql-database-security-overview.

Azure Storage

This section pertains to all storage services that Azure offers: blobs, tables, queues, disks, and files. Storage Service Encryption (SSE) is turned on by default for all storage services. Data is encrypted before being written to storage and decrypted after the data is read. Microsoft manages the keys by default, but a user can provide their own keys as an option. The Azure Key Vault can be used to manage the keys, or the Azure Key Vault APIs can be used by applications to support user-supplied keys. This is documented at: https://docs.microsoft.com/en-us/azure/storage/common/storage-service-encryption-customer-managed-keys?toc=%2fazure%2fstorage%2fblobs%2ftoc.json.

Use Shared Access Signatures to Access Storage Account Resources

An Azure Storage Account is a logical container used to store and access Azure Storage data objects. Each storage account has two Azure generated 512-bit Storage Access Keys (SAK), which are used for authentication when the storage account is accessed. SAKs are similar to a root password in that users with the key have unfettered access to all the storage account's services. To authenticate access to an Azure Storage Account from a client application, an account access key is required. However, most client applications should not require access to the entire storage environment, which includes all storage services, including Tables, Queues, Files, Blobs and Azure virtual machine disks.

Recommendation: Use Shared Access Signatures

A shared access signature (SAS) provides granular access to services within a storage account. The goal is to avoid distributing the SAK to other users or applications, hardcoding it, or saving it anywhere in plaintext that is accessible to others. Editorial and asset management workflows should only require access to a subset of services within a storage account, and access via SAS should be used. When creating a SAS, specify only those permissions that are required to perform editorial and management workflows to uphold the principle of least privilege. Additionally, ISE recommends using a policy to explicitly define controlled SAS expiration times, tokens, and source IP address ranges.

Documentation: Azure SAS has multiple use-cases and deployment models. Our recommendation focuses on the use of Shared Access Signature to control access to services within a storage account. Azure SAS is defined here: https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview.

Periodically Rotate Access Keys

To authenticate to an Azure Storage Account from a client application, an account access key is required. Regenerating SAKs can affect associated Azure services (e.g., Batch) that are dependent on the storage account.

Recommendation: Regenerate Storage Access Keys Regularly

Administrators should regenerate storage account access keys periodically according to company or regulatory policies. All storage account client services that use the access keys to access the storage account must be updated to use the regenerated key.

Documentation: Azure compute virtual machines, and Batch computes processes rely on storage accounts. Each storage account has a set of access 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 in storage. These storage account keys should be rotated after a predefined period (for example, every 90 days) or after completion of production.

Azure storage security processes are defined here: https://docs.microsoft.com/en-us/azure/storage/storage-securityguide.

Enable Encryption at Rest

All Virtual machines have an operating system disk and possibly multiple attached data disk devices. These devices should be encrypted using a key managed by the Key Vault. Encryption at rest protects data if attackers gain access to the physical storage devices used by an account.

Recommendation: Encrypt Disks

Administrators should enable encryption when Virtual machines or data disks are instantiated.

Documentation: Azure compute virtual machines rely on on-disk storage for operations system and user data. In the context of editorial and asset management workflows, these storage account disks should be protected at rest using disk encryption since these disks may hold sensitive content, metadata, or personally identifiable information.

Azure Disk encryption is described here: https://docs.microsoft.com/en-us/azure/security/azure-security-disk-encryption-faq.

Require Secure Transfers

Azure storage accounts can support both HTTP and HTTPS protocols for transfers. The former is a plaintext connection that can expose assets to adversaries in privileged network locations, while the latter uses robust encryption to ensure the confidentiality and integrity of network traffic. Depending on the method used to create a storage account, the secure transfer required property may not be enabled.

Recommendation: Enable Secure Transfer Required Property

Administrators should ensure that the secure transfer required property is enabled on any applicable storage accounts.

Documentation: Information on secure transfer is available at: https://docs.microsoft.com/en-us/azure/storage/common/storage-require-secure-transfer.

Enable Advanced Threat Protection

Advanced Threat Protection attempts to detect anomalous or potentially malicious interactions with a storage account. This automated protection can help detect attacks on storage without human intervention.

Recommendation: Enable Advanced Threat Protection

Administrators should ensure that advanced threat protection is enabled on any applicable storage accounts. Alerts should be sent to an email address that can be reviewed in a timely manner.

Documentation: Information on advanced threat protection is available at: https://docs.microsoft.com/en-us/azure/storage/common/storage-advanced-threat-protection.

Protect Assets with Digital Rights Management (DRM)

Media production workflows rely heavily on ad-hoc and near real-time asset sharing and collaboration. Azure Media service video delivery can be used to deliver video assets within a content production team across the globe. The video assets should be protected when shared in this manner.

Recommendation: Use DRM on Media Assets

Media assets should be delivered using a robust DRM solution designed to prevent unauthorized copying or sharing. Azure Media Services supports popular DRM solutions, including Microsoft PlayReady, Apple FairPlay, and Google Widevine.

Documentation: A guide describing how to enable DRM is located here: https://docs.microsoft.com/en-us/azure/media-services/latest/protect-with-drm.

Enable Soft Delete

A soft delete allows for the recovery of blob data when it is erroneously modified or deleted by an application or a user. Deleted data transitions to a soft deleted state instead of being permanently erased when soft delete is enabled. Similarly, a soft deleted snapshot is generated when data is overwritten. Also, a user-defined retention period must be configured to specify the amount of time that soft deleted data is stored and available for recovery.

Recommendation: Enable Soft Delete

Administrators should ensure that soft delete is enabled for any applicable storage accounts (e.g., storage accounts that contain sensitive media assets).

Documentation: Information on soft delete is available at: https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-soft-delete?tabs=azure-portal.

Configure Firewall Rules

Azure storage accounts have a public endpoint that is accessible through the internet. The Azure storage firewall provides a way to control and secure access for public endpoints by allowing administrators to configure rules that only allow traffic originating from specific IP addresses, IP ranges, or from a list of subnets in an Azure virtual network (VNet).

Recommendation: Configure Firewall Rules

Azure storage firewall rules can grant access to traffic from specific public Internet IP address ranges, allowing connections from specific Internet or on-premises clients. Rules can also be configured to grant access to traffic from specific VNets. These rules can be combined on the same storage account, which is useful when editorial and asset management workflows require implementing a secure hybrid network that extends the on-premises network to Azure.

Documentation: Information on Azure storage firewalls is available at: https://docs.microsoft.com/en-us/azure/storage/common/storage-network-security.

Use Private Endpoints

Private endpoints for Azure Storage accounts can be used to allow clients on a virtual network (VNet) to securely access data over a Private Link[7]. The private endpoint is assigned an IP address from the VNet address space for your storage account service. This means that network traffic between the VNet and the storage account traverses over the Microsoft backbone network, eliminating exposure from the public Internet.

Recommendation: Use Private Endpoints

Azure administrators can use private endpoints for controlling access to storage accounts. First, private endpoints can be used only to accept connections from specific VNets. This can be achieved by configuring the storage firewall to deny or control access through its public endpoint. Note that it is not necessary to add a firewall rule to allow traffic from a VNet since the storage firewall only controls access through a public endpoint. In addition, private endpoints can be used to securely connect to storage accounts from on-premises networks that connect to the VNet using VPN or ExpressRoute with private peering, which is useful when editorial and asset management workflows require implementing a secure hybrid network that extends the on-premises network to Azure.

Documentation: Information on using private endpoints for Azure storage is available at: https://docs.microsoft.com/en-us/azure/storage/common/storage-private-endpoints.

Enable Storage Logging

Storage Analytics logs various information about successful and failed requests to a storage service. Logs can be used to detect and troubleshoot operational and security issues with a storage service. For instance, administrators can track how each request made against Azure storage was authorized because logs indicate whether a request was made by using an OAuth 2.0 token, Shared Key, or a Shared Access Signature (SAS).

Recommendation: Enable Storage Logging

We recommend enabling logging on storage services, especially on services that house high-value and sensitive assets used for editorial and asset management workflows.

Documentation: Information on Azure storage logging is available at: https://docs.microsoft.com/en-us/azure/storage/common/storage-analytics-logging.

[7] https://docs.microsoft.com/en-us/azure/private-link/private-link-overview

Azure Virtual Machines

Use Public Key Authentication for Virtual Machines

Azure Virtual machines allow authentication using a username and password or using public keys. Using public key authentication is a good practice since it eliminates weak passwords on the system or hardcoded default passwords. Brute-force attacks are infeasible for those wanting to break into the system. Disabling login via passwords and requiring login keys is a configuration option when instantiating a VM or VM scale set in Azure.

Recommendation: Use SSH Public Key Authentication

Administrators should configure Azure VMs to use SSH keys for authentication, eliminating the need for a password to log in.

Documentation:

- Information on creating and using SSH key pairs for Linux VMs is available at : https://docs.microsoft.com/en-us/azure/virtual-machines/linux/mac-create-ssh-keys.
- Information on using SSH keys for Windows VMs is available at: https://docs.microsoft.com/en-us/azure/virtual-machines/linux/ssh-from-windows.

Use Hardened OS Images for VM Instantiation

Azure provides a number of pre-built operating system images available to users for rapid deployment. In the current state, OS images are supplied with default configurations. Some default configurations of OS images leave VM instances open to vulnerabilities, which could be exploited by publicly known, readily available proof of concepts and exploits. For example, the images may have their SSL/TLS configurations set up for maximum compatibility, rather than security.

Recommendation: Use Hardened OS Images

Azure administrators should harden all OS images using current security best practices. General hardening steps, including closing unused ports, remove unnecessary services, and updating software.

Documentation: Additional information about Azure virtual machines is available at: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/overview.

Ensure Virtual Machines are Patched Regularly

The applications and operating systems in virtual machines hosted in Azure must be kept up to date to ensure they are protected against known security issues. To better accomplish this, organizations should develop patch management strategies that ensure all software hosted in Azure is updated frequently.

Recommendation: Implement a Patch Management Policy

Administrators should develop and implement a patch management process that regularly updates operating systems and applications hosted in Azure Virtual Machines. A common update cadence is 30 days; however, updates may need to be applied more frequently depending on the software used and the sensitivity of the machines' workflows. Administrators should also develop a strategy to deploy updates that patch critical vulnerabilities in a timely manner, which may need to occur between regular update cadences.

Ensure Disk Storage is Encrypted at Rest

Encryption at rest protects data stored on the physical medium from unauthorized access if an attacker is able to interact with the storage device. Azure Virtual Machines support encryption at rest via the Azure Disk Encryption feature, which uses BitLocker and dm-crypt on Windows and Linux systems, respectively, to encrypt data.

Recommendation: Enable Disk Encryption Features

Azure Disk Encryption should be enabled on all disks used by virtual machines, especially if they store sensitive assets. The additional protection offered by this feature provides a layer of defense-in-depth if attackers are able to compromise the physical storage devices used by the virtual disk.

Documentation: Azure provides additional information about disk encryption here: https://docs.microsoft.com/en-us/azure/security/fundamentals/azure-disk-encryption-vms-vmss.

Azure Virtual Networking

Use Network Security Groups (NSG)

While Azure completely restricts incoming traffic from the Internet, it is more permissive about internal traffic—essentially allowing open communication between all VM instances within the virtual network (VNet) similar to a physical LAN network. While the default endpoint security features are a useful mechanism for securing Azure VMs, they have limited functionality. Network Security Groups (NSG) secure both inbound and outbound access to both Azure VMs and Azure VNets, similar to a traditional firewall. NSG rules are defined with a standard five-tuple definition (source network, source port, the destination network, destination port, protocol) as well as a name, type, priority, protocol, and access (allow or deny).

Recommendation: Secure Traffic Flow with Azure Network Security Groups

Use network security groups to restrict access to assets to the greatest degree possible.

Documentation: Filtering of network traffic is a security and workflow function. Traffic filters can be deployed to direct or restrict access to compute farm subnets based on a production or a source. A network security group is a list of security rules that allow or deny network traffic to resources connected to Azure Virtual Networks (VNet). Azure Network Security Groups are described here[8].

Use Azure Application Gateway

Azure Application Gateway is a load balancer that allows load balancing at the application layer. This type of load balancer allows it to make decisions based on the content of incoming HTTP requests, such as the path of the URL. More importantly, Azure Application Gateway includes a web application firewall (WAF) that can help protect web applications from common types of exploits.

Recommendation: Use the Azure Application Gateway for Web Applications

To benefit from the defense in-depth, Azure Application Gateway provides, configure the load balancer to protect any web applications deployed in Azure.

[8] https://docs.microsoft.com/en-us/azure/virtual-network/security-overview

Documentation: Additional information about Azure Application Gateway can be found at: https://docs.microsoft.com/en-us/azure/application-gateway/overview

Secure Connections to Azure Networks

Azure supports multiple methods for connecting on-premises networks to the cloud. Services like VPN Gateway, ExpressRoute, and Bastion can be used to create secure connections to Azure resources.

Recommendation: Use Dedicated Services to Connect to Azure

Azure's managed services provide hardened and secured methods for joining on-premises networks to Azure. VPN Gateway uses traditional VPN technologies to create private connections, while ExpressRoute uses dedicated physical connections to connect to Azure so that traffic does not travel on the public Internet. Azure Bastion can be used to provide secure access to Azure resources using a managed bastion host.

Documentation:

- VPN Gateway: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways
- ExpressRoute: https://docs.microsoft.com/en-us/azure/expressroute/expressroute-introduction
- Bastion: https://docs.microsoft.com/en-us/azure/bastion/bastion-overview

Isolate Virtual Appliances Using Individual Subnets

When using virtual appliances, such as a firewall, WAN accelerator, Active Directory server, or VPN gateway in Azure, isolate them in their own gateway subnet. Virtual appliances are useful to create routes between Azure resources and on-premises data centers.

Recommendation: Use Gateway Subnet with User-Defined Routing

Use a gateway subnet with a user-defined routing mechanism to isolate networking appliances in their own dedicated private network subnets. Specifically, to secure these services and appliances, prevent direct internet connectivity by placing them in a separate subnet with an NSG acting as a firewall. Additionally, close all ports on the appliance or service servers except those necessary for authentication, authorization, and server synchronization.

Documentation: Editorial and asset management workflows may require implementing a secure hybrid network that extends the on-premises network and datacenter to Azure. The user-defined routing mechanism in the gateway subnet filters or blocks all user requests other than those received from the on-premises network.

The Azure network DMZ architecture is described here: https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/dmz/secure-vnet-dmz.

Apply a Multi-tiered Architecture for Virtual Networks

A three-tiered virtual network has front-, mid- and back-end network segments to create isolation between various types of assets. In an editorial and asset management environment, place compute or data storage resources in the backend while placing authentication and traffic shaping (e.g., load balancing) servers in the front-end.

The front end, which contains web servers in its own subnet, directly faces the internet. The mid-tier, which contains business logic, does not have direct internet access, either inbound or outbound and can only be reached from the front-end subnet. The back-end, in its own isolated subnet, contains persistent data such as a database system or storage and can communicate only with the middle-tier.

Recommendation: Apply a Multi-tiered Architecture for VNets

Place workload compute machines in the back-end while placing authentication and scheduler software servers in the front-end.

Documentation: Azure virtual networks are described here: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-overview. Furthermore, a reference architecture for deployment of N-tier applications is described here: https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/virtual-machines-windows/n-tier.

Create Separate Virtual Networks for Productions

Create separate virtual networks (VNets) for each production to isolate different workloads from one another.

Recommendation: Separate Productions using VNets

Separate productions and projects into multiple VNets to better isolate assets in the event of a network compromise.

Documentation: Azure virtual networks are described here: https://docs.microsoft.com/enus/azure/virtual-network/virtual-networks-overview; network security concepts and guidance can be found here: https://docs.microsoft.com/en-us/azure/virtual-network/security-overview.

Use Service Endpoints

By default, Azure service traffic from a virtual network uses public IP addresses as source IP addresses. With service endpoints, the identity of a virtual network is extended to Azure services over a direct connection. Specifically, the source IP addresses of the virtual machines in a virtual network switch from using public IP addresses to using private IP addresses. This provides improved security by removing public internet access to resources and only allowing traffic from the specified virtual network.

In addition, using service endpoints ensure that service traffic from a virtual network always remains on the Microsoft Azure backbone network. This allows for continuous auditing and monitoring of outbound traffic from virtual networks.

Recommendation: Secure Service Traffic from Virtual Networks

Use service endpoints to secure critical Azure service resources to only the specified virtual networks.

Documentation: Additional information about service endpoints can be found at: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview

Avoid Deprecated Cryptography for IPSec VPNs

In contrast to SSL/TLS, configuring the set of permitted cipher suites at each end of an IPsec connection can be a manual and time-consuming process. The following concerns affect the configuration of the Cloud IPsec VPN:

- *IKEv1 protocol supported*. The Cloud IPsec VPN, for compatibility, supports both IKE version 1 (introduced in 1998) and IKE version 2 (introduced in 2005). One of the goals of IKEv2 was to improve security over IKEv1, including cryptographic weaknesses[9]. Specifically, the IKEv1 supports 3DES and SHA1 (SHA128) as the encryption and hashing algorithms, respectively.
- *HMAC-MD5, supported (IKEv2).* The Cloud IPsec VPN allows *HMAC-MD5* to be used for integrity checking. HMACMD5 is deprecated due to weaknesses in the underlying MD5 algorithm[10].
- SHA1 *supported (IKEv2).* The Cloud IPsec VPN allows SHA1 to be used for integrity checking. SHA-1 has been practically broken and is considered insecure and ineffective[11].
- *DES, 3_DES, supported (IKEv2)*. The Cloud IPsec VPN allows DES, 3_DES to be used for data encryption. DES is inherently insecure, while Triple-DES has much better security characteristics but is still considered cryptographically flawed.

Recommendation: Configure IPsec to Avoid Deprecated Cryptography

Configure the IPsec VPN to avoid the use of deprecated or inherently insecure protocols and modes of operations. Administrators should configure their IPsec clients in accordance with their security policies and avoid using ciphers and security protocols that have been deemed broken or weak by the industry, including but are not limited to DES, 3DES, SHA1, and MD5.

Documentation: More information on cryptographic requirements and Azure VPN gateways can be found here: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-compliance-crypto.

Configure Network Watcher

Azure Network Watcher is a built-in service that can be used to monitor, diagnose, view metrics, and enable or disable logs for resources in an Azure virtual network. The network watcher service can be used to monitor unstable or inactive endpoints, troubleshoot connections, and review other system-level issues. Though Network Watcher is enabled automatically, it is still necessary to enable NSG flow logs or connection monitor to view traffic flows.

Recommendation: Configure Connection Monitor

Configure connection monitor between VMs or resources where appropriate.

Recommendation: Configure Flow Logs

Network Watcher uses flow logs to view information about ingress and egress through an NSG. Targeted and focused NSG flow definition will capture the right traffic patterns and provide insight into potential anomalies.

Documentation: Logging and monitoring of network traffic is a security and workflow function. Azure Network Watcher is described here: https://docs.microsoft.com/en-us/azure/network-watcher.

[9] RFC 4306 Appendix A, https://tools.ietf.org/html/rfc4306#page-96
[10] http://tools.ietf.org/html/rfc6151
[11] https://security.googleblog.com/2017/02/announcing-first-sha1-collision.html

Enable DDoS Protection

Azure provides a basic level of protection against distributed denial of service (DDoS) attacks for resources deployed in Azure. In addition to this basic protection, administrators may enable DDoS Protection Standard, which provides an enhanced level of security to Azure resources.

Recommendation: Enable DDoS Protection Standard

Administrators should enable on DDoS Protection Standard on resources that would benefit from this protection, such as externally facing web applications.

Documentation: Azure DDoS Protection Standard is described here https://docs.microsoft.com/en-us/azure/virtual-network/ddos-protection-overview.

About ISE

ISE is an independent security firm headquartered in Baltimore, Maryland. We are dedicated to providing clients proven scientific strategies for defense. On every assessment, our team of analysts and developers use adversary-centric approaches to protect digital assets, harden existing technologies, secure infrastructures, and work with development teams to improve our clients' overall security.

Assessing the client through the eyes of potential attackers allows us to understand possible threats and how to protect against those attacks. Our security analysts are experienced in information technology and product development, which allows them to understand the security problems the client faces at every level. In addition, we conduct independent security research, which allows us to stay at the forefront of the ever-changing world of information security.

Attacks on information systems cannot be stopped, however with robust security services provided by an experienced team, the effects of these attacks can often be mitigated or even prevented. We appreciate the confidence placed in us as a trusted security advisor. Please do not hesitate to get in touch for additional assistance with your security needs.

Independent Security Evaluators, LLC

4901 Springarden Drive
Suite 200
Baltimore, MD 21209
(443) 270-2296

contact@ise.io