

Secure Cloud for Azure (IaaS)

Design Guide

June 2021

Contents

Abstract	4
Scope	4
SAFE Architecture - Introduction	5
Cloud Business Flows.....	6
Public Cloud Attack Surface.....	8
Solution Overview.....	9
What is our security approach?	9
Secure Cloud Business Flows	10
Cisco's Secure Architecture for Azure.....	11
Secure Cloud architecture	11
Business flows in Cisco's Reference Architecture	12
Security Integrations	15
Cisco Tetration	15
Cisco Advanced Malware Protection for Endpoints	17
Cisco Stealthwatch Cloud	19
Cisco Umbrella	20
Cisco Firepower Next-Generation Firewall and Adaptive Security Appliance	22
Web Application Firewall and DDoS Prevention	23
Cisco Duo	25
Cisco SecureX threat response	27
Design Implementation	29
Deployment Overview:	29
Set up the Azure Virtual Network	29
Integrating Stealthwatch Cloud	32
Setting up Cisco Umbrella	32
Setting up the Azure Database for MySQL servers	35
Setting up the Virtual Machine Scale Sets	37
Setting up the App and Web load balancers	43
Setting up the Firepower Next-Generation Firewalls	45
Enabling WAF and DDoS protection	56
Integrating with Cisco SecureX threat response	60
Validation Testing	62
Tetration	62
Test case 1: Creating an application workspace for Azure cloud application	62
Test case 2: Using ADM to discover the policies for Azure workloads and setting up an app view	64

Test case 3: Enforcing the policies on workloads.	66
Test case 4: Discovering the vulnerable packages on the Azure workloads.	68
Advanced Malware Protection for Endpoints	69
Test Case: Quarantine a suspicious file	69
Stealthwatch Cloud	73
Test Case: Monitor suspicious activity	73
Cisco Umbrella	74
Test Case: DNS security	74
Cisco Firepower NGFWv and CDO.....	75
Test Case: Enforce an access policy using CDO	75
Web Application Firewalls and DDoS Protection.....	77
Azure WAF and DDoS	77
Test Case: Monitor Web and DDoS activity on Azure cloud	77
Radware Cloud WAF and DDoS	80
Test Case: Monitor Web and DDoS activity on Radware cloud	80
Duo Beyond	83
Test Case 1: Set up the cloud application for Two-Factor Authentication (2FA)	83
Test Case 2: Monitor 2FA activity from Duo admin portal	84
Cisco SecureX Threat Response	85
Test Case: Track malicious Activity on threat response	85
Appendix.....	87
Appendix A- Acronyms Defined	87
Appendix B- Software Versions	88
Appendix C- References	88

Abstract

This design guide aligns with the [Cisco® Secure Cloud Architecture guide](#). The Secure Cloud Architecture guide explains the secure architecture for cloud applications, critical business flows; attack surfaces and corresponding security controls required for the cloud environment. This guide proposes a Cisco Validated Design (CVD) for security in a tiered application architecture. The solution proposed in this guide leverages Cisco security controls along with Cloud-Native security controls to achieve the desired security posture for applications in the Azure cloud.

Scope

This document illustrates the design and security aspects of an application hosted in Azure Cloud. Along with the design and security specifications, this document also delves into the details of implementation and validation steps for the proposed architecture.

This guide covers the following security controls.

- Cisco Tetration
- Cisco Advanced Malware Protection for Endpoints (AMP4E)
- Cisco Stealthwatch Cloud (SWC)
- Cisco Umbrella
- Azure Network Security Groups (NSG)
- Cisco Firepower Next-Generation Firewalls (NGFW)
- Cisco Adaptive Security Appliance (ASA)
- Cisco Defense Orchestrator (CDO)
- Azure Web Application Firewall (WAF) and DDOS prevention
- Radware Cloud Web Application Firewall (WAF) and DDOS prevention
- Cisco Duo Beyond
- Cisco SecureX threat response

For setting up the web application, we used the following Azure cloud components and services.

- Azure Virtual Network (VNET) and Subnets
- Azure Route Tables
- Azure Database for MySQL
- Azure Virtual Machine Scale Sets (VSS)
- Azure Virtual Machines
- Azure Internal Load balancer (Standard)
- Azure External Load balancer (Standard)
- Azure Storage Containers
- Azure Resource Manager (ARM) Templates
- Azure Private Links

- Azure Front Doors (AFD)

SAFE Architecture - Introduction

As your data flows from an increasing number of devices to your data center or private/public cloud, you must understand your data flow, to be able to protect it. Cisco SAFE is an architectural approach that helps you visualize this transit of the data in terms of business flows, understand the attack surface associated with these flows and hence, devise appropriate capabilities to secure them. This framework provides complete guidance from the initial identification of business flows in a given architecture to securing it and then deploying and validating the solution.

These validated designs provide guidance that is complete with configuration steps that ensure secure deployments for your organization. Cisco Validated Designs (CVDs) for various SAFE PINs can be found at [SAFE home page](#).

Cisco SAFE simplifies network security by providing solution guidance using the concept of ‘Places in the Network’ (PINs). This design guide is a recommended threat defense architecture for the Cloud PIN (see figure 1). Within the Cloud PIN, this design guide specifically covers the Azure public cloud.

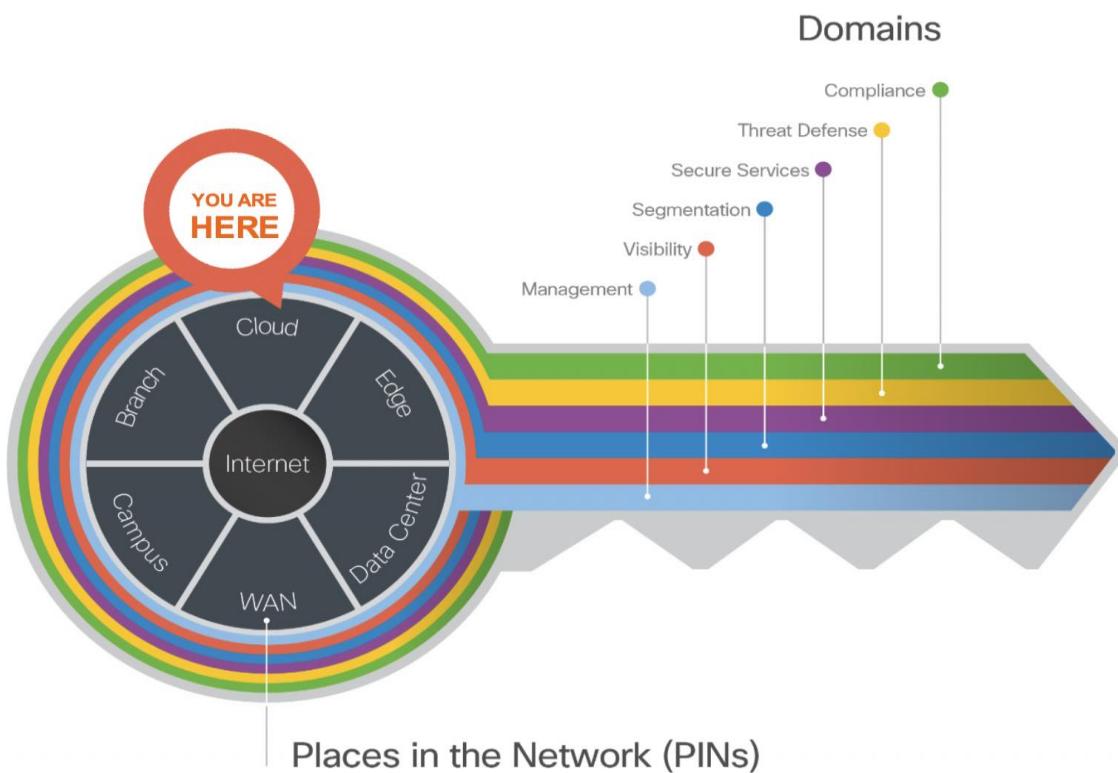


Figure 1.
Key to SAFE framework

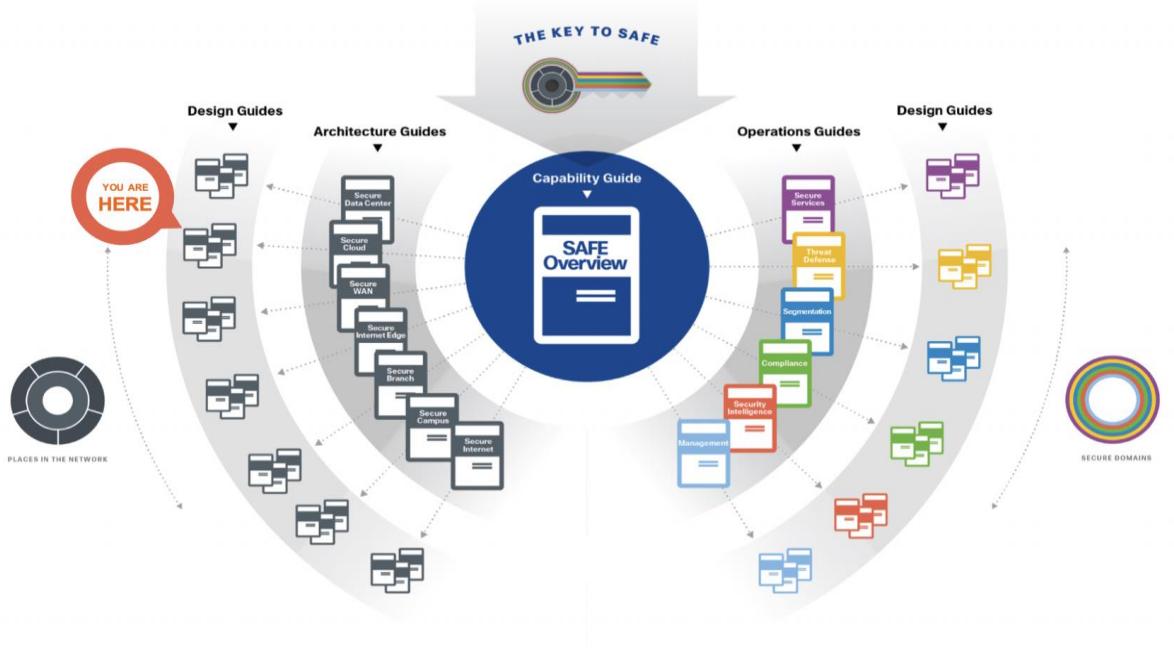


Figure 2.
SAFE Guidance Hierarchy

For more information on SAFE framework and architecture/design guides, check out the [SAFE documentation](#) (select architecture/design tab).

Cloud Business Flows

SAFE uses the concept of business flows to simplify the identification of threats. This enables the selection of very specific capabilities necessary to secure them.

This solution addresses the following business flows for a typical tiered web application hosted in the Azure cloud:

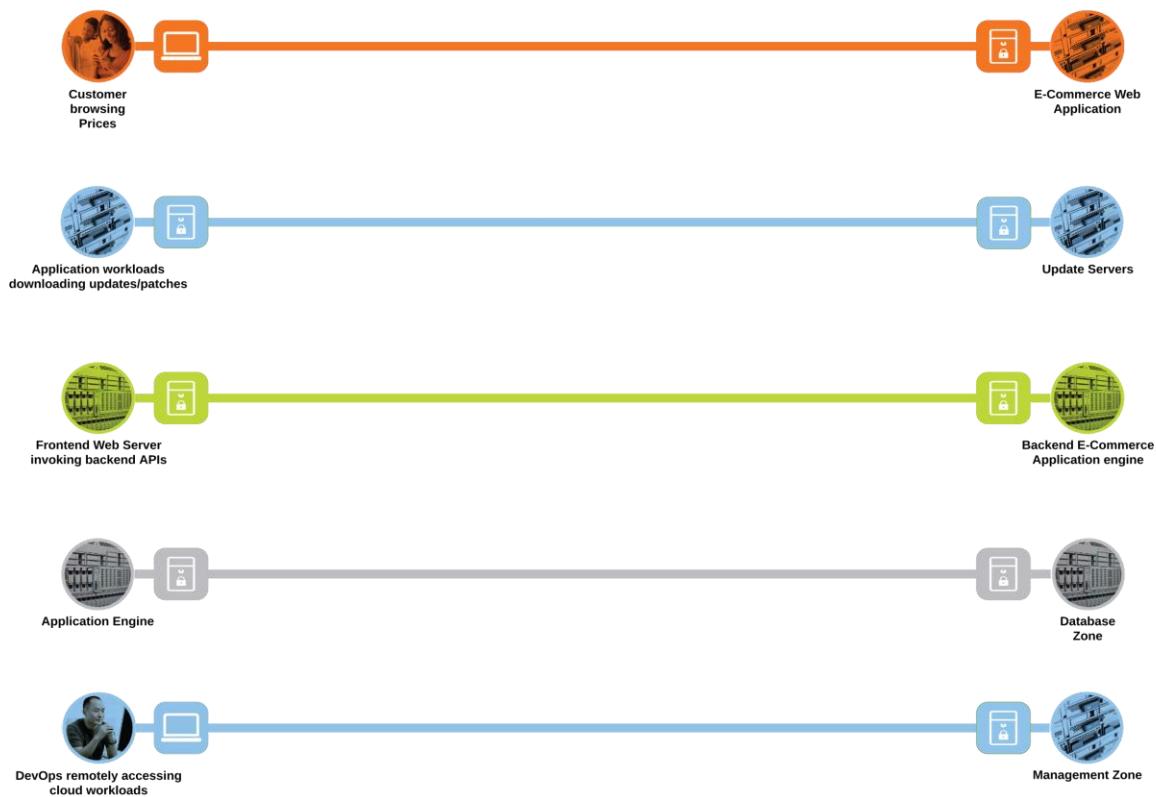


Figure 3.
Cloud business flows

- Customer browsing an e-commerce web application. The customer, sitting somewhere out on the Internet, browses the e-commerce web application hosted in the Azure cloud
- Application workloads downloading updates/patches from update servers outside the cloud (Internet). Application workloads sitting in the cloud need to reach out to various update servers to fetch the updates and patches at regular intervals
- Systems communicating east/west within the Azure cloud. For example- the frontend web servers will make HTTP requests to backend application engine or the application engine workloads will make API calls among themselves
- Application workloads transacting data with database server within the cloud
- DevOps remotely accessing the management zone for workload management/update/patching purposes

Public Cloud Attack Surface

The secure cloud design protects systems by applying security controls to the attack surface found in the public cloud. The attack surface in public cloud spans the business flows used by humans, devices, and the network.

Threats include; rogue identity, DDoS, web vulnerabilities, infections, and advanced persistent threats allowing hackers the ability to take control of your devices and networks.

Considering the business flows elaborated in the last section (Figure 3), a deep dive into the attack surface for each of those business flows can be showcased as below.

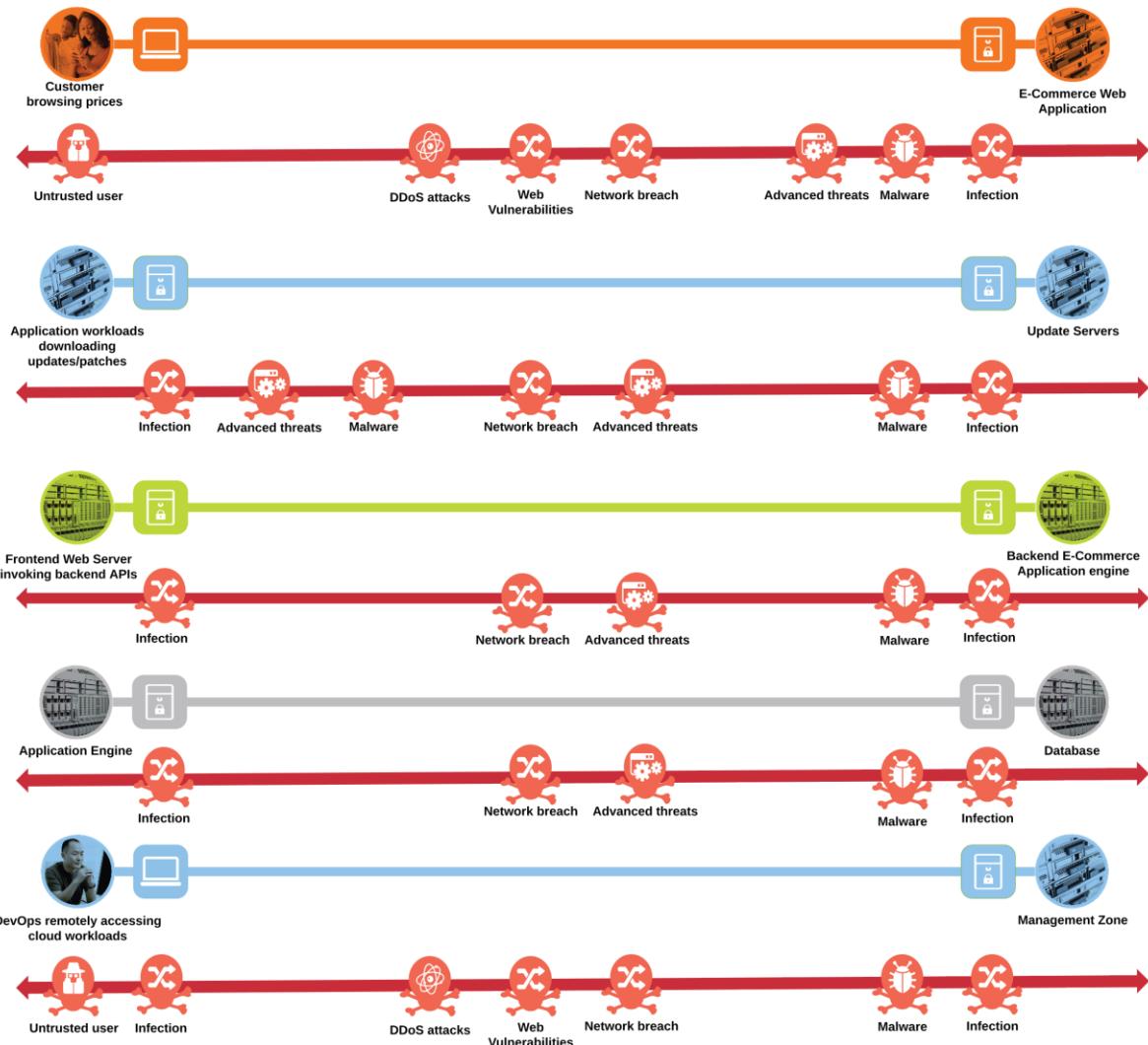


Figure 4.
Public cloud attack surface

- An untrusted/compromised user, out on the Internet, may try to exploit the cloud application or flood it with fake traffic to render it incapable of serving the genuine users
- The workloads need to communicate with update servers out on the untrusted public network. An attacker might compromise workloads to download malware to the application environment or upload crucial data to malicious servers
- Systems communicating east/west within the Azure cloud may spread the infection from one workload to another within the cloud, eventually compromising the whole application
- An attacker may compromise the application workloads to steal or corrupt data stored on the database servers
- A malicious user may try to gain the same privileged access as DevOps to compromise the complete application environment in Azure

Solution Overview

Cisco's security approach for the modern cloud applications allows companies to achieve:

- Improved resiliency to enable cloud availability and secure services
- Operational efficiency from automated provisioning and flexible, integrated security
- Advanced threat protection from [Cisco TALOS](#) – industry-leading threat intelligence to stay up to date, informed, and secure

What is our security approach?

Specific capabilities are necessary to protect the public cloud and build the appropriate layers of defense. These capabilities work together to create several layers of defense protecting the cloud applications. The top priorities or the three pillars that we keep in mind while designing the secure public cloud solutions are:

- **Visibility** - Complete visibility of users, devices, networks, applications, workloads, and processes
- **Segmentation** - Reduce the attack surface by preventing attackers from moving laterally, with consistent security policy enforcement, application access control and micro-segmentation
- **Threat Protection** - Stop the breach by deploying multi-layered threat sensors strategically in the public cloud to quickly detect, block, and dynamically respond to threats



Visibility

"See Everything"

Complete visibility of users, devices, networks, applications, workloads & processes



Segmentation

"Reduce the attack surface"

Prevent attackers from laterally (east-west) with application access control & micro-segmentation



Threat protection

"Stop the breach"

Quickly detect, block and respond to attacks before hackers can steal data or disrupt operations

Secure Cloud Business Flows

Developing a defense-in-depth architecture requires identifying existing threats and applying appropriate security capabilities to thwart them. Business flows and the corresponding attack surface and threat patterns that we defined earlier (Figures 3 and 4) are mapped to their corresponding security controls as below.

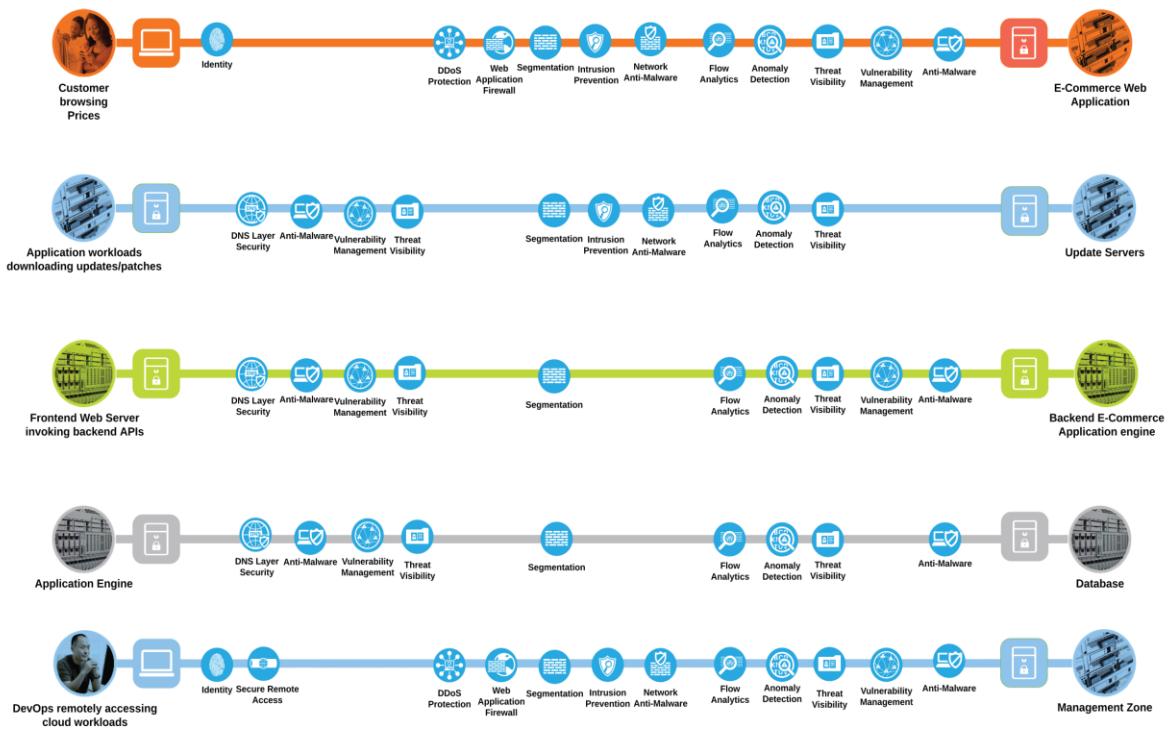


Figure 5.
Secure business flows

Cisco's Secure Architecture for Azure

The tiered architecture has been a popular underlying principle for web application deployment for over a decade now and it remains equally relevant to date.

The multi-tier architecture provides a general framework to ensure decoupled and independently scalable application components. Each tier is separately developed, scaled, maintained and secured.

In the simplest tiered architecture form, the web applications would have the following layers:

Web tier: The end-user directly interacts with this layer. This tier has all the static web content.

Application tier: This tier is responsible for translating the user actions to application functionality. This tier carries the core application code components. For example, application code performing the read/write database operations.

Database tier: Storage tier or the database tier holds the data relevant to the application.

In this document, we are securing a tiered web application in the Azure cloud. We add various security capabilities and controls, that we established in the previous sections, to a tiered web application model to make it much more robust, secure and transparent in its security posture.

Secure Cloud architecture

The Cisco Secure Cloud reference architecture solution includes all the security capabilities that we illustrated in previous sections.

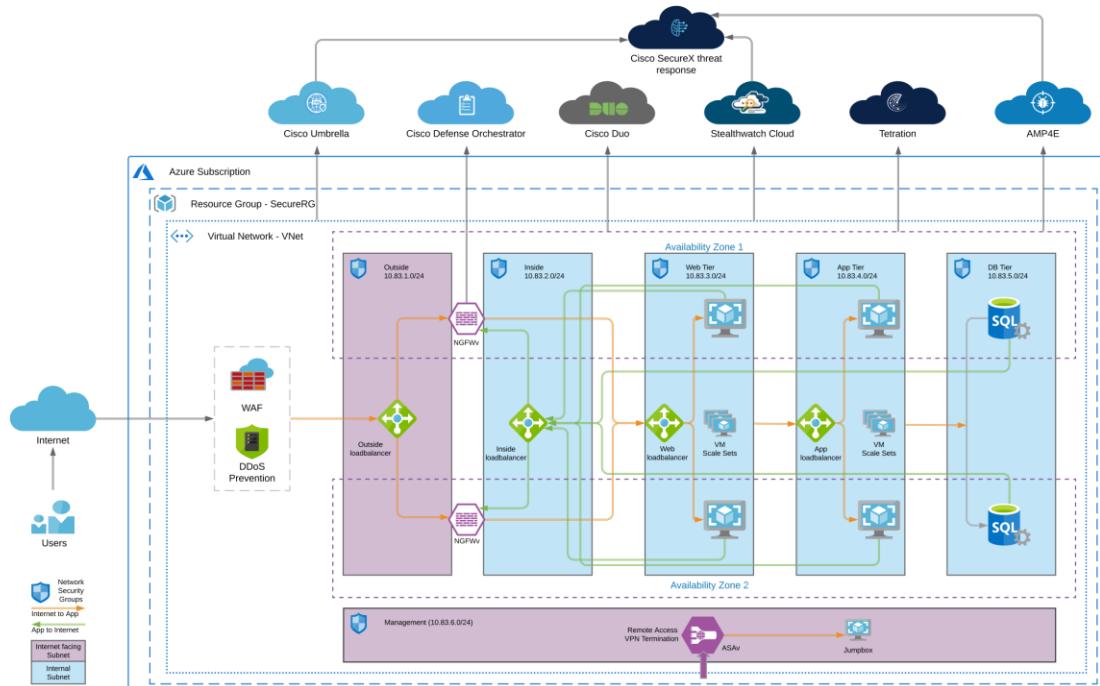


Figure 6.
Cisco Secure Cloud Reference Architecture

Business flows in Cisco's Reference Architecture

Considering the design above, all the threats, corresponding security capabilities and solutions required to attain those capabilities can be mapped as below.

Threat	Security Capability	Security Solutions	
	Attackers or malicious users accessing restricted resources and information.	Identity based access	Cisco Duo – 2FA
	Massively scaled attacks that overwhelm services.	DDoS prevention	Azure DDoS prevention Radware DDoS prevention
	Attacks against poorly developed applications and web vulnerabilities.	Web Application Firewalls	Azure Web Application Firewall Radware Web Application Firewall
	Network breach causing unauthorized access and malformed packets between and within application in the cloud.	Segmentation	Cisco Firepower NGFWv Azure Network Security Groups Cisco Tetration
	Zero-day malware attacks and other forms of covert threats.	Threat visibility	Cisco Stealthwatch Cisco Tetration Cisco AMP4E Cisco Firepower NGFWv
	Attacks using worms, viruses, or other techniques.	Intrusion Prevention	Cisco Firepower NGFWv Cisco AMP4E
	Infections, attackers using a compromised workload to spread the damage.	Micro-segmentation	Cisco Tetration
	Malware distribution among workloads or between servers.	Anti-malware	Cisco Firepower NGFWv Cisco AMP4E
	Traffic, telemetry, and data exfiltration from successful attacks. Covert threats.	Flow Analytics	Cisco Stealthwatch Cisco Tetration Cisco Umbrella
	Exploiting privileged access to run shell code	Process Anomaly Detection & Forensics	Cisco Stealthwatch Cloud Cisco Tetration Cisco AMP4E
	Malware distribution across networks.	Network Anti-Malware	Cisco Firepower NGFWv

Threat		Security Capability	Security Solutions
	Exploiting unpatched or outdated applications.		Vulnerability Assessment and Workload Inventory
	Redirection of session to malicious domains.		DNS Layer Security
	Exposed services and data theft.		VPN Gateway or Concentrator
			Cisco ASA Cisco Firepower NGFWv

We have established the attack surface and, the capabilities and security solutions that we needed to secure the business flows mentioned previously.

- Customer browsing an e-commerce web application
 - Access to the web application is secured using Duo – Multi-Factor Authentication (MFA)
 - WAF and DDoS Services protect against web vulnerabilities and denial of service attacks. In this document, we will demonstrate two options for WAF and DDoS protection- Azure cloud native service and Radware cloud service. Based on your preference, you could choose either
 - Network segmentation is done using next-generation firewalls (NGFW) to protect against any network or perimeter level breaches. NGFWs also provide next-generation IPS and AMP capabilities along with stateful firewall, AVC (Application Visibility and Control) and URL filtering.
 - At the macro level, workloads are segmented into various tiers using Azure Network Security Groups. Micro-segmentation of workloads is done using the Tetration policy enforcement agents. This would prevent any malware or malicious movement within the pool of workloads in a specific tier
 - Stealthwatch Cloud provides enhanced threat visibility into workload activity and the Azure cloud. It looks for any anomalous activity within the application environment. It also facilitates the flow analytics
 - Tetration agents allow us to gain a deep visibility into vulnerable packages and processes on the workloads that an attacker may leverage. It also provides a very robust network flow analytics for workload communications
 - AMP4E detects and quarantines any malware that may infect the workloads
- Application workloads downloading updates/patches from update servers
 - Application workloads are segmented into App and Web tier using Azure Network Security Groups. No direct inbound public access is allowed to the App and Web servers, management access is allowed only from the management tier (also controlled via Azure Network Security Group)
 - DNS layer security is achieved using Cisco Umbrella. This prevents any accidental or deliberate exposure to a malicious domain
 - Stealthwatch Cloud and Tetration provide enhanced threat visibility and flow analytics
 - AMP4E detects and quarantines any malware that may get downloaded to application workloads

- Network segmentation is achieved using Cisco Firepower next-generation firewalls to protect against any network or perimeter level breaches. NGFWs also provide network IPS and AMP capabilities, that would prevent any malicious file download right at the network perimeter
- Systems communicating east/west within the Azure cloud
 - Workloads are segmented into tiers at the macrolevel using Azure Network Security Groups. Web, App, Database and Inside tier has no direct inbound public access/addresses. Only Management and the Outside tier is allowed Public IP addressing, hence exposing them to untrusted public network/internet
 - Micro-segmentation within Web and App tier is done using the Tetration enforcement agents. This restricts any internal movement within an application tier
 - DNS layer security using Umbrella provides visibility into workload activity
 - Stealthwatch Cloud and Tetration provide enhanced threat visibility and flow analytics for this flow. They also look for any anomalous movement within the application environment or among the workloads within a tier. Tetration agents provide deep visibility into the workloads
 - AMP4E protects against malware spread
- Application engine transacting data with database server within the cloud
 - Azure Network Security Groups restrict access to the database. Only App tier is allowed to communicate with database tier
 - DNS layer security using Umbrella
 - Stealthwatch Cloud and Tetration provide enhanced threat visibility and flow analytics. They also look for any anomalous movement within the application environment or among the workloads within a tier. Tetration agents provide deep visibility into the workloads
 - AMP4E protects the application workloads against any malware infection
- DevOps remotely accessing the management zone for workload management/update/patching purposes
 - Anyconnect VPN mobility client is used to provide Secure Remote Access to the management tier. An ASA or NGFWv can be used for VPN termination. We tested a standalone ASAv for this design. Refer to the [Secure Remote Worker](#) design guide for detailed information on secure remote access designs and deployments
 - Management zone is segmented using Azure Network Security Groups. This provides the control knob for restricting access to workloads or the various tiers
 - Stealthwatch Cloud and Tetration provide enhanced threat visibility and flow analytics. They also look for any anomalous movement or activity within the application environment or from the management tier. Tetration agents provide deep visibility into the workloads
 - AMP4E protects the jump servers and workloads against any malware infection

Security Integrations

Let's look at each of the security integrations in this secure design in more depth, we will start from the security controls on the workload itself and go all the way to the edge of our public cloud web application.



We start by looking at workload security using Tetration and Advanced malware protection, followed by an agentless deployment of Stealthwatch for greater visibility into the Azure cloud and workload activity. Then, we will look into Umbrella DNS layer security at the Azure VNET level.

Afterwards, we move to perimeter protection using Cisco Firepower NGFW (policy orchestrated by Cisco Defense Orchestrator), WAF and DDoS protection. We will explore two different options for WAF and DDoS protection – Azure Cloud Native Service and Radware Cloud.

Lastly, we will delve into securing authentication to our application using Duo MFA.

To connect all these security pieces together, we will look at Cisco SecureX threat response integrations to get a unified view of the Azure cloud security.

Cisco Tetration

Tetration has a SaaS offering that provides the capability to do micro-segmentation in a highly flexible manner along with an in-depth visibility into the workloads.

Tetration offers visibility and enforcement agents that are installed on the workloads. Enforcement agents provide an additional capability to enforce policies.

Tetration can dynamically learn various ongoing changes in the cloud workload environment and enforce an adaptive micro-segmentation. The Tetration portal allows us to create workspaces and graphical views for applications and enforce security from the web application point of view unlike the traditional network perspective.

The Tetration platform supports multi-cloud and hybrid environments and hence, make the whole process of security operations seamless across the board.

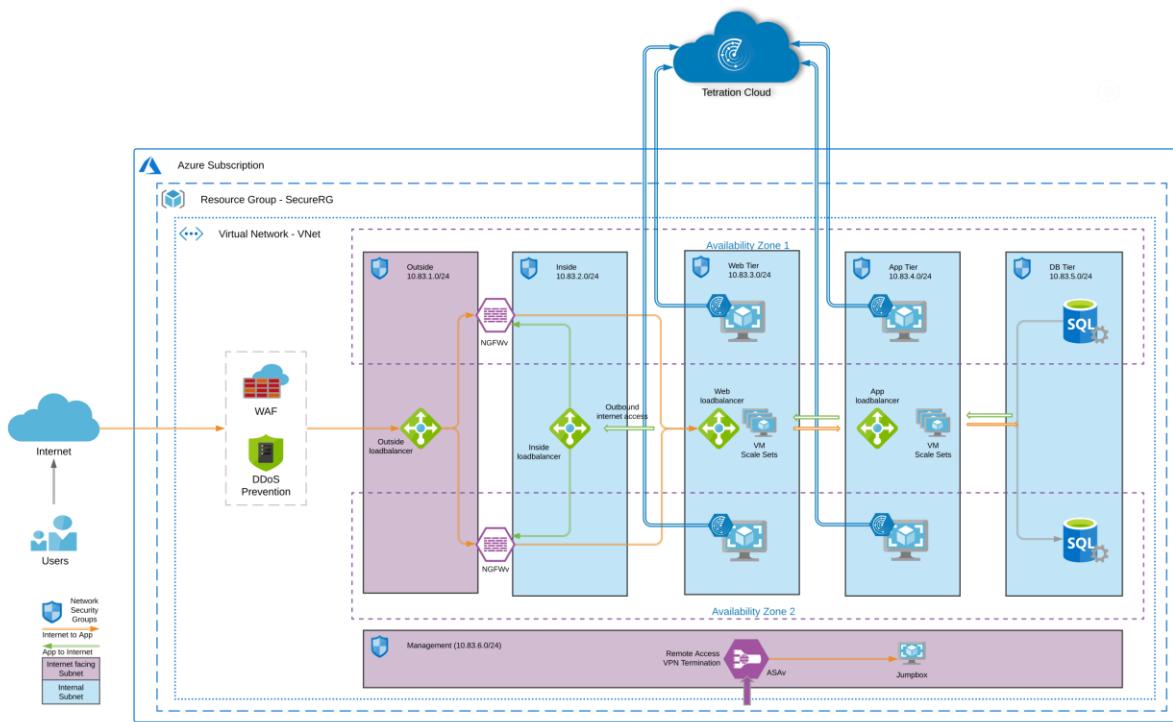


Figure 7.
Cisco Tetration

In this specific architecture, Web and Application tier has workloads in [Virtual Machine Scale Sets](#) (VSS). To enable the auto-provisioning of Tetration agents, we used the custom scripts provided under [Azure VM extensions](#). When the VSS deploys a new workload, the custom script will install the Tetration agent on it as part of the initialization process. Refer to the implementation section of this guide for more details.

Once the Tetration agent on the new workload is registered with the Tetration cloud (SaaS), it starts exporting the network flow and process information to the Tetration cloud engine for analysis. Tetration ensures Cisco's Zero Trust model by offering key features like:

- Policy enforcement (Micro-segmentation)
- Visibility into workload process activity
- Network flow visibility
- Software vulnerability reports
- Forensic analysis
- Behavior deviations

Based on all these features and more, the Tetration dashboard provides us with a very convenient and flexible scoring mechanism to monitor the security compliance of cloud applications. Tetration considers six parameters to calculate this score (Figure 8), and these parameters can be adjusted based on one's preference or requirements.

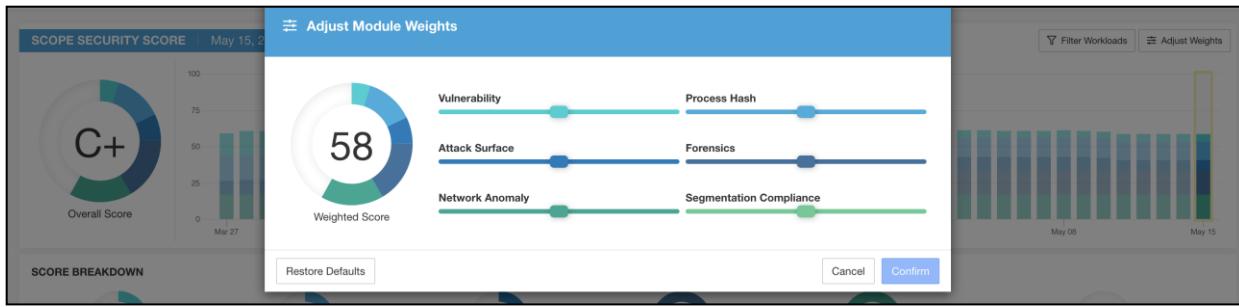


Figure 8.
Tetration Dashboard - Weighted Score

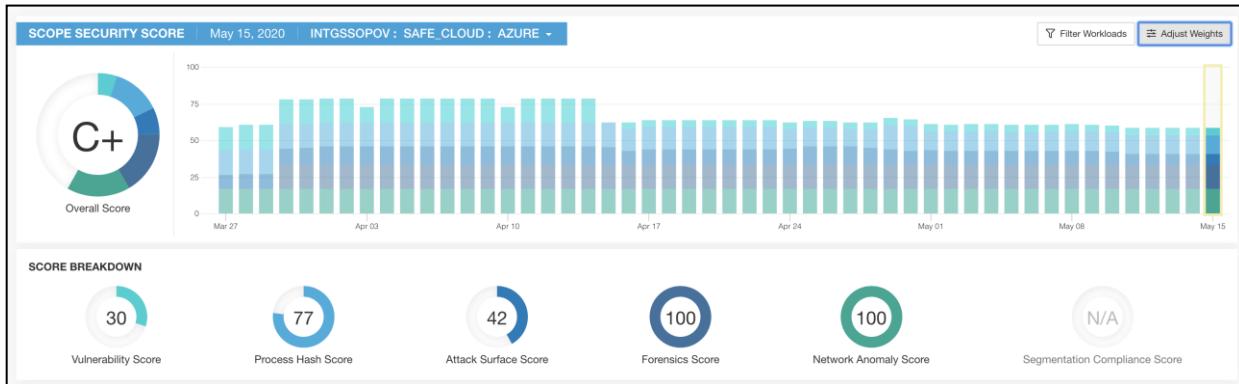


Figure 9.
Tetration Dashboard - Compliance Score Board

Refer to the [Tetration documentation](#) for more detailed information on cloud workload protection.

Cisco Advanced Malware Protection for Endpoints

The AMP4E agents installed on the cloud workloads provide us protection against zero-day attacks. Powered by [Cisco TALOS](#), AMP4E not only relies on antivirus, but also uses machine learning and file reputation to block both file-based and file-less attacks. It also enables you to isolate the infected host before the malware is spread onto the others in the network. Advanced Malware Protection also supports taking forensic snapshots that help immensely with the security investigations.

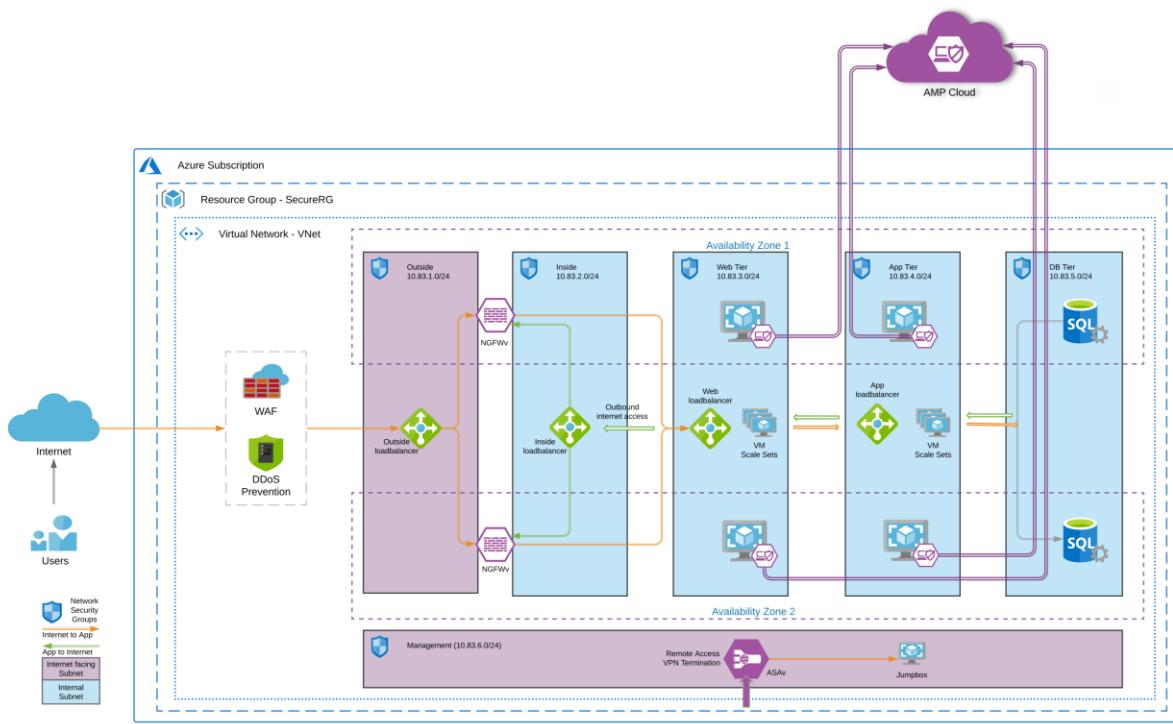


Figure 10.
Cisco Advanced Malware Protection for Endpoints

In this specific architecture, just like the Tetration agent, the web and application workloads in [VSS](#) are auto-provisioned with AMP4E agents using custom scripts option available via [Azure VM extensions](#). When the [VSS](#) deploys a new workload, the custom script will install the AMP4E agent on the workload as part of the initialization process.

As soon as AMP4E agent on the new workload registers with the AMP cloud, the workload is continuously monitored and reported for any malicious activity. AMP's host isolation feature comes in very handy to contain any spread of malware in the cloud workloads.

File Detection	Detection	EICAR.TEST.FILE.FromHash
Connector Info	Fingerprint (SHA-256)	275a021b...f651fd0f
Comments	File Name	elcar.com
	File Path	/home/centos/elcar.com
	File Size	68 B
	Parent Fingerprint (SHA-256)	782bed6a...5f896bd2
	Parent Filename	wget

Figure 11.
AMP Dashboard - Threat monitoring

Cisco Stealthwatch Cloud

Stealthwatch Cloud (SWC) helps overcome the visibility challenge, especially in public cloud environments. It provides an agentless deployment in the Azure cloud.

Stealthwatch Cloud pulls the Network Security Group (NSG) flow logs from the designated Azure storage containers. It learns the Azure environment and baselines the resources. NSG flow logs have the flow information associated with various Azure resources, even for those that are not strictly tied to a static IP address. SWC is capable of correlating the IPs and then tying them back to their origin Azure service. In other words, SWC performs dynamic entity modeling and organizes all the Azure resources based on the functions that they're performing. For example, the entity could be categorized as a firewall, an application server or a load balancer and so on. This type of resource profiling and modeling is extremely important to look for any suspicious activity within the cloud application environments.

In addition to NSG flow logs, Stealthwatch Cloud also consumes other telemetry sources like Azure Activity logs for additional context and alerting.

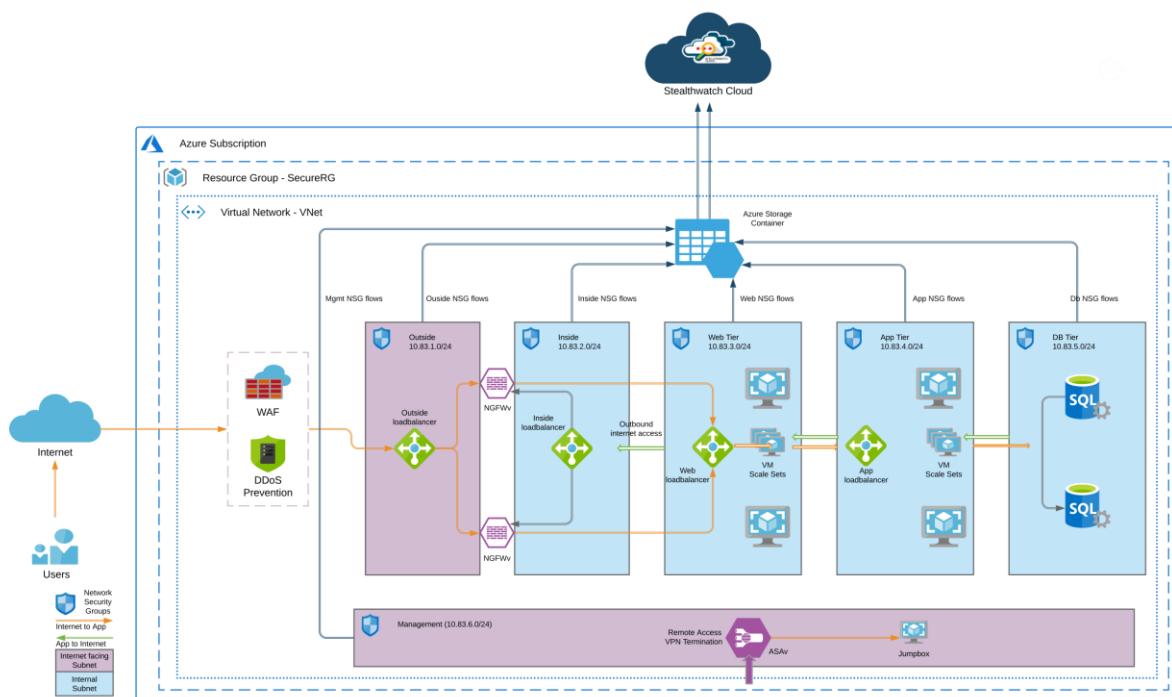


Figure 12.
Cisco Stealthwatch cloud

Once the Stealthwatch Cloud finishes identifying the entities, it baselines their behavior over a fixed period of time. As soon as the baselining is completed, any unexpected behavioral change of the entities and the way different cloud services communicate with each other is alerted on. This helps to maintain deep visibility into the cloud environment and hence, track and prevent any unauthorized transfer of data or resource access.

Some of the common Stealthwatch alerts related to the Azure cloud include:

- Azure Activity Log IP Watchlist Hit – This is triggered by an IP address that matched a user-defined or an integrated watchlist. This alert may indicate that an unauthorized user has gained access to Azure
- Azure Activity Log Watchlist Hit – This alert is triggered when Azure Activity Log reports an event on a user-supplied watchlist
- Azure Permissive Storage Account – If the storage accounts are identified by Azure Security Center as having unrestricted firewall settings then this alert is triggered
- Azure Security Event – Triggers when Azure Security Center reports a medium or high severity event
- Azure Virtual Machine in Unused Location – Triggers when an Azure Virtual Machine has been created in a previously unused location

Alerts			
<input type="text" value="Search..."/> <input type="button" value="🔍"/>		Status	Tags
Status ▾ Tags ▾ Assignee ▾ Sort ▾			
7 open alerts sorted by newest			Page 1 of 1
! Excessive Access Attempts (External)	i-032dc6c1e859be077	3 hours ago	🔗 43
<input type="checkbox"/>	#299		
! Excessive Access Attempts (External)	i-031bb97fc8aa5a9b1	3 hours ago	🔗 42
<input type="checkbox"/>	#298		
! Excessive Access Attempts (External)	i-09e0d2badc2cf3a1c	4 hours ago	🔗 16
<input type="checkbox"/>	#496		
! Excessive Access Attempts (External)	ScaleWebServers i-0fa81682fd2ca2fdb, i-01b15f0e2c9d254f9	14 hours ago	🔗 33
<input type="checkbox"/>	#364		
! Excessive Access Attempts (External)	i-0b071afe7f70b7134	1 day, 16 hours ago	🔗 8
<input type="checkbox"/>	#397		
! Geographically Unusual Remote Access	i-031bb97fc8aa5a9b1	6 days, 10 hours ago	
<input type="checkbox"/>	#530		
! Inbound Port Scanner	Network	1 week, 4 days ago	🔗 8
<input type="checkbox"/>	#331		

Figure 13.
Stealthwatch Cloud – Alerts

Cisco Umbrella

Cisco Umbrella offers flexible cloud-delivered security. It combines multiple security functions into one solution. Cisco Umbrella solutions provide DNS-layer security, secure web gateway, cloud-delivered firewall, cloud access security broker (CASB), and interactive threat intel. This document covers Umbrella DNS-layer protection for the workloads in the Azure Virtual Network (VNET).

The Umbrella DNS policies allow you to dictate block policy for a variety of pre-defined web categories. More details on web categories can be found in [Umbrella documentation](#). It also gives you the flexibility to apply the policies to specific identities. For example, you could have one set of rules for your Azure cloud application and another set for a different site.

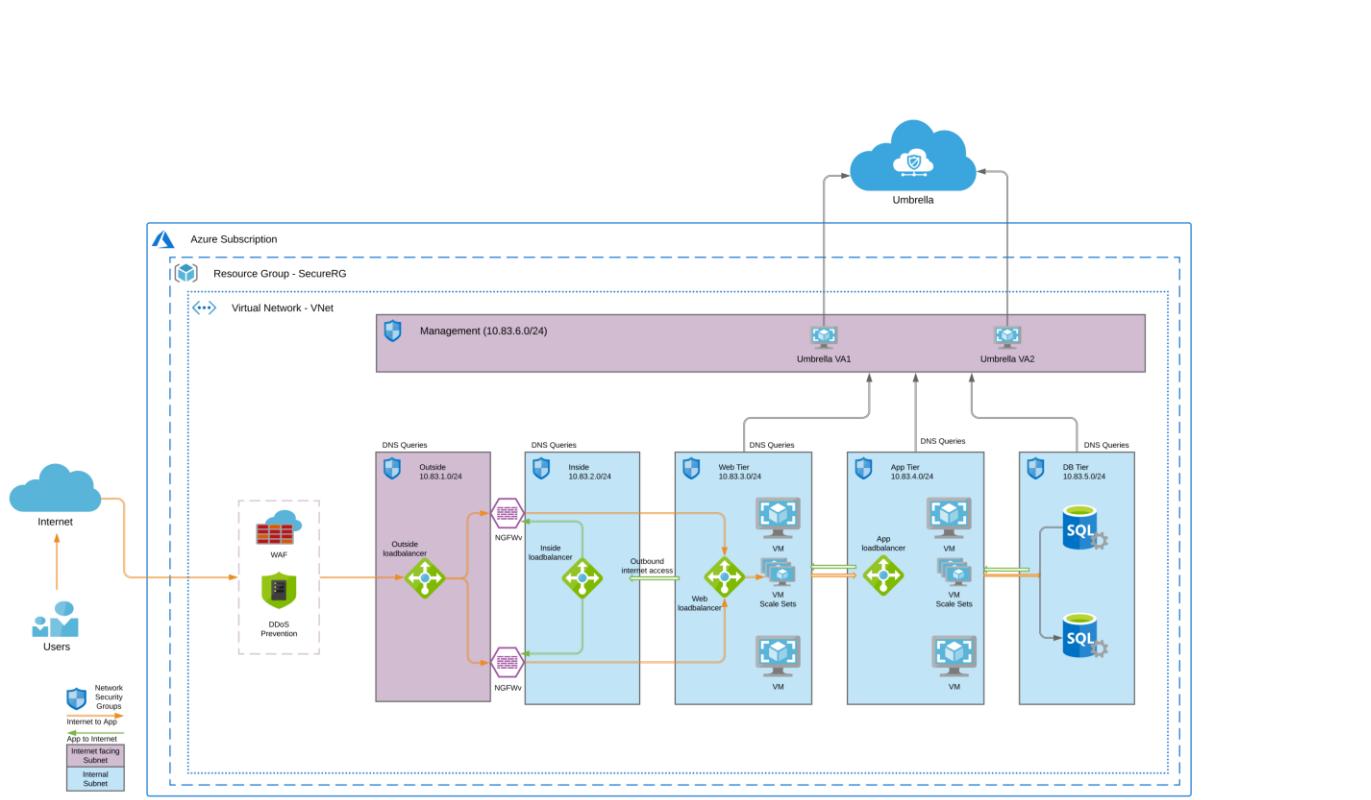


Figure 14.
Cisco Umbrella – DNS layer Security

We deploy Umbrella Virtual Appliances (VA) in the Management tier of the Azure VNET. These VAs act as DNS forwarders to Umbrella. The Azure VNET offers the option to configure custom DNS settings, allowing us to point the cloud resources in a given VNET to Umbrella VAs instead of Azure local DNS. Every resource, that is launched into the VNET, will use these Umbrella DNS forwarders, to provide a control knob for the DNS layer security.

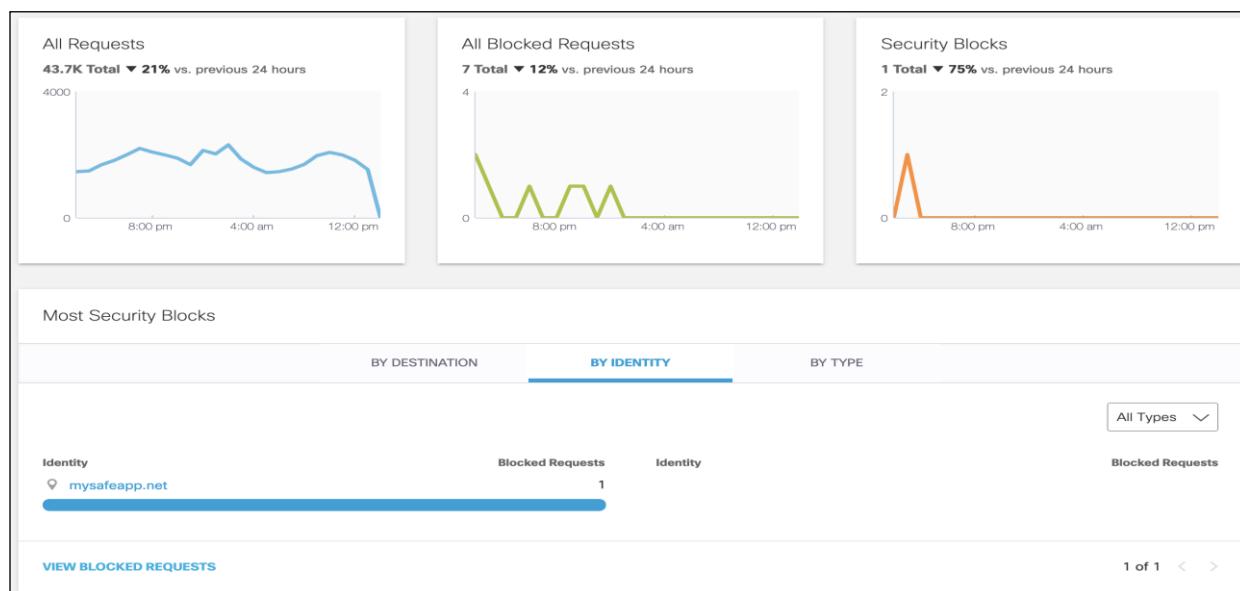


Figure 15.
Umbrella – DNS Traffic Monitoring

Cisco Firepower Next-Generation Firewall and Adaptive Security Appliance

Cisco® Firepower Next-Generation Firewall Virtual (NGFWv) appliance combines Cisco's network firewall with advanced next-gen IPS, URL filtering, AVC and malware detection (AMP). In this design, we use NGFWv to secure the network perimeter from all sorts of threats from public Internet. This ensures that we have security controls like filtering, intrusion prevention and malware detection right at the gateway to the cloud application.

To provide secure remote access to the Management tier, we use Cisco ASA v as a VPN headend. Cisco ASA v offers secure remote access capabilities using Anyconnect VPN mobility client. You could also use NGFWv for this purpose. For detailed information on secure remote access deployments, refer to the [Secure Remote Worker](#) SAFE design guide.

Cisco Defense Orchestrator (CDO) is used for management and policy orchestration. CDO provides one security policy, faster deployment, and smart configuration management. It eliminates the time-consuming complexity of managing policies across multiple FTDs and ASAs. Cisco Defense Orchestrator helps to correct issues such as unused, duplicate, and inconsistent objects hence ensuring consistent policies for firewalls.

We use CDO to manage both NGFWv and ASA v, providing centralized management.

Note: The terms Next-Generation Firewall (NGFW) and Firepower Threat Defense (FTD) are used interchangeably throughout this guide. Both these terms refer to Cisco Firepower Next-Generation Firewalls in the context of this document. Azure marketplace offering is available under the name ‘Cisco Firepower NGFW Virtual (NGFWv)’.

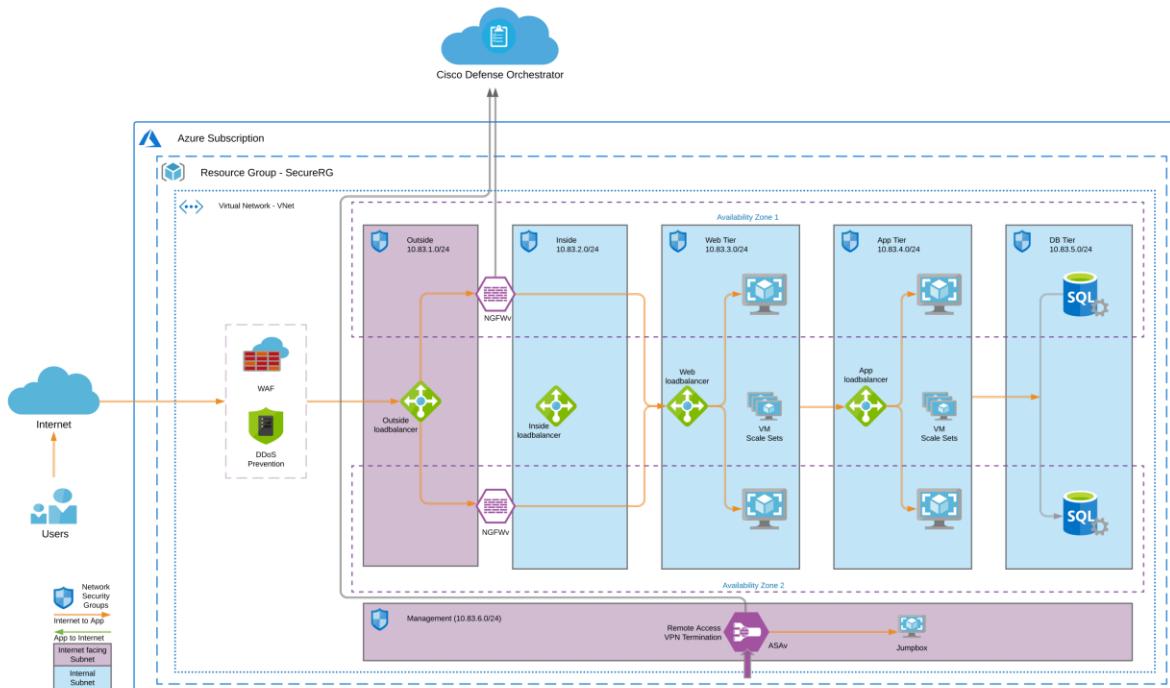


Figure 16.
NGFWv - Traffic flow from Internet User to application

User to application traffic flow

When the user out on the Internet tries to browse the cloud-hosted web application, it lands on ‘Outside’ load balancer after being scanned by WAF and DDoS protection system for any malicious activity. The destination IP at this point is the public IP of the ‘Outside’ load balancer. ‘Outside’ load balancer sits in the Outside tier (segmented using Azure Network Security Group) and load balances traffic onto the pool of outside interfaces of next-generation firewalls. The NGFW receiving the request then forwards this traffic to ‘Web’ load balancer, to be load-balanced on to web servers bundled in Web Virtual Machine Scale Set (VSS).

Before the traffic leaves the inside interface of the firewall (in the Inside tier), the source of this packet is translated (Network Address Translation) to inside interface IP of the firewall and the destination is changed to ‘Web’ load balancer IP. The source IP is translated here to ensure traffic symmetry.

Web server receiving this incoming request, after being load-balanced by the ‘Web’ load balancer, fetches dynamic content from app workloads and returns the final response directly to the firewall which forwarded the initial request. At this point, firewall routes this response back to the end-user via the outside interface.

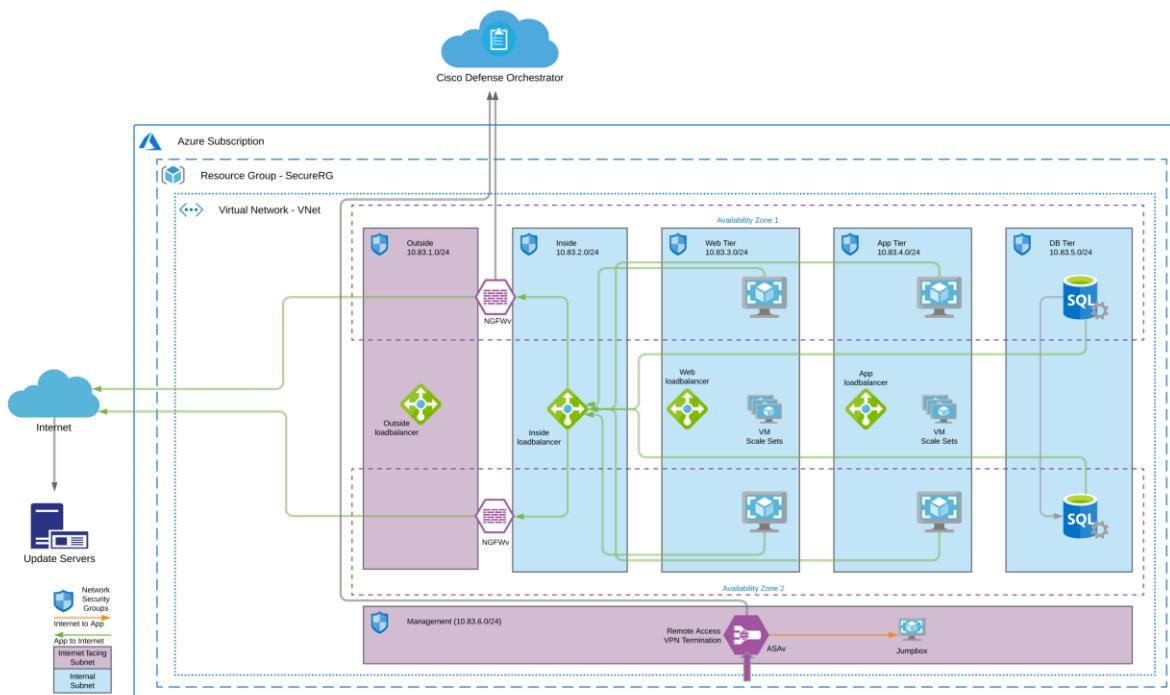


Figure 17.
NGFWv - Traffic flow from workloads out to the Internet

Application to update servers

For the traffic or requests initiated from the Web or Application tier, we use the Azure Route Tables to point the default route to ‘Inside’ load balancer sitting in the Inside tier. The ‘Inside’ load balancer load-balances all the outgoing requests to inside interfaces of the firewalls. The firewalls then route the traffic out to the Internet destination/update server after port address translation (PAT) to the outside interfaces. The outside firewall interfaces are assigned public IPs.

Web Application Firewall and DDoS Prevention

Public cloud has become a common place to host critical applications and make these applications available to end-users (internal or external). As a result, it is essential to ensure these applications receive the same level

of protection from distributed denial of service (DDoS) and advanced web attacks that on-premises applications do. For this design, we validated two different solutions i.e. Azure WAF & DDOS service and Radware Cloud WAF & DDoS service.

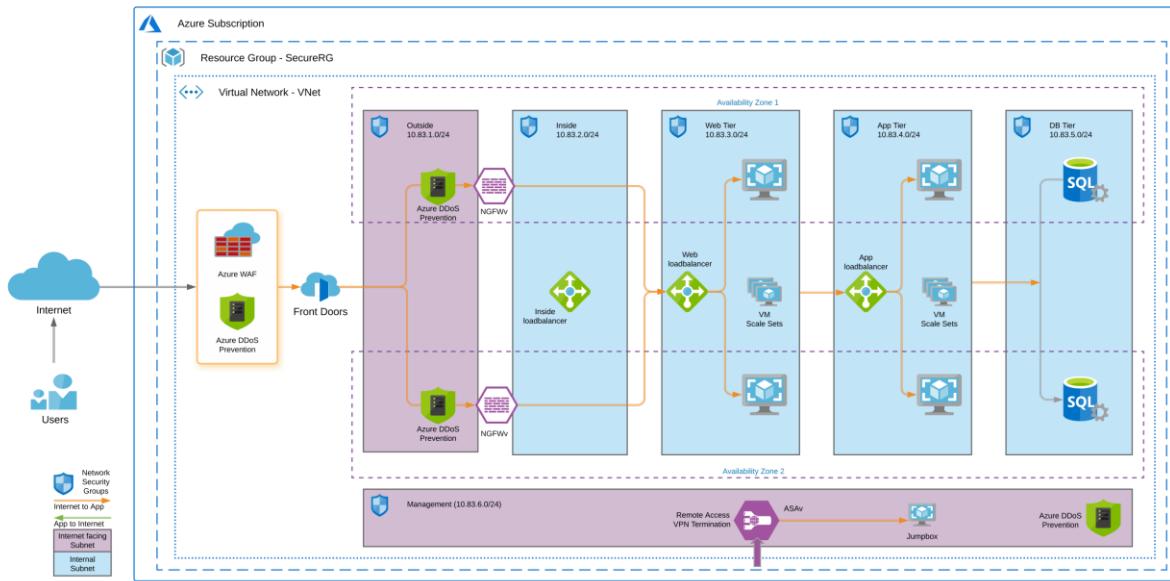


Figure 18.
Azure WAF and DDoS protection

Azure offers WAF and DDoS as native services that controls access to the application by allowing or blocking web requests. This protects web applications from common web exploits. In addition to web traffic protection, DDoS component provide network flow monitoring to protect against DDoS attacks.

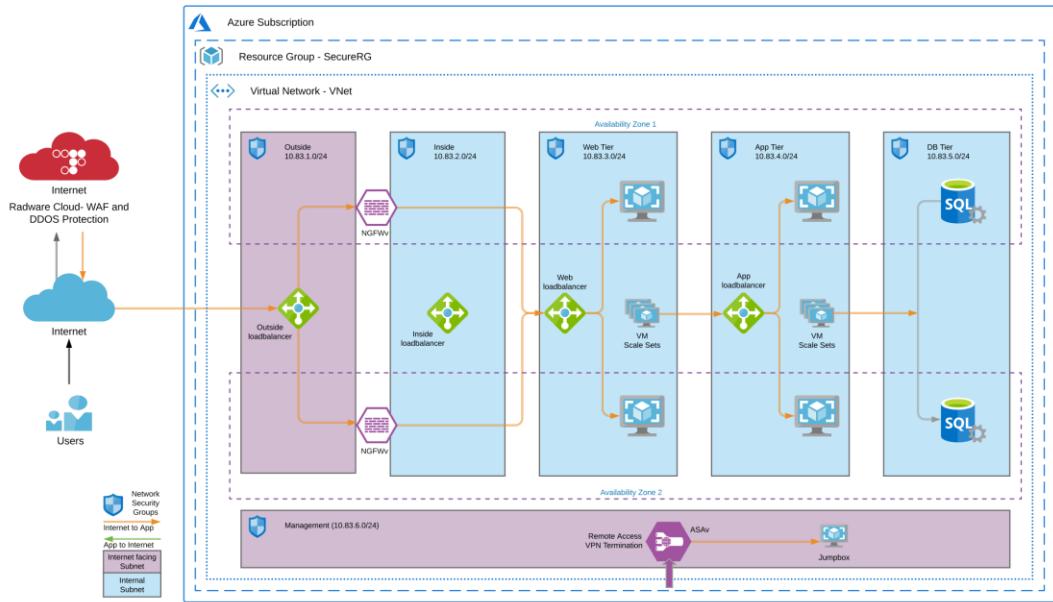


Figure 19.
Radware Cloud - WAF and DDoS Prevention

Radware's Cloud Security Services offer easy-to-deploy cloud-based security that can be integrated with your cloud environments to provide proactive, automated protection from advanced threats. Radware provides applications hosted in the cloud with the protection from the full breadth of DDoS attacks with real-time mitigation and no added latency in peacetime.

Radware acts as a man in the middle. Application's domain name points to the Radware Cloud service. Traffic is first routed to the Radware Cloud and scanned for any malicious activity. Post-inspection, the traffic is forwarded to the origin servers in the Azure cloud. Refer to the implementation section of this guide for more deployment level details.

Cisco Duo

Cisco Duo provides secure access to applications and data, no matter where the users are, on any device, and from anywhere. Cisco Duo's secure access solution creates trust in users, devices, and the applications they access. Cisco Duo provides the following functions:

- Multi-Factor Authentication: Verify the identity of all users with Duo's strong multi-factor authentication
- Single Sign-on: Seamless, single dashboard access to all applications
- Remote Access: Secure access to cloud and on-premises applications and servers, with or without VPN
- Device Trust: Check that user devices meet security standards before granting them access
- Adaptive Access Policies: Set policies to allow or block access attempts by a user or a device, based on contextual factors

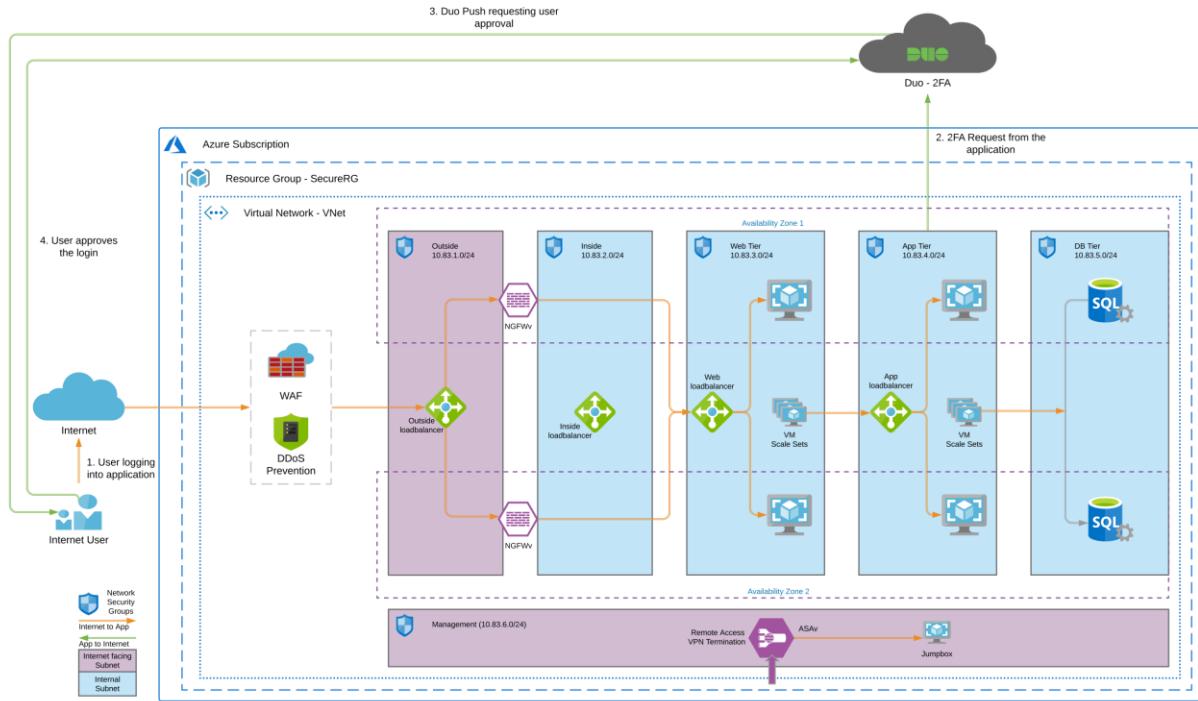


Figure 20.
Duo MFA Push

In this design, we used Duo's Multi-Factor Authentication (MFA) for our Azure cloud application. Multi-factor authentication from Duo protects the cloud applications by using a second source of validation, like a phone or token, to verify user identity before granting access. MFA not just allows you to build a zero-trust framework but is also essential for compliance purposes. Duo provides native integration for any application. Refer to the implementation section of this guide for more details.

Admins have several options when it comes to enrolling new users in Duo, such as self-enrollment, Azure sync, Active Directory sync, and OpenLDAP sync. Duo admin portal allows a highly convenient way to track any user activity.

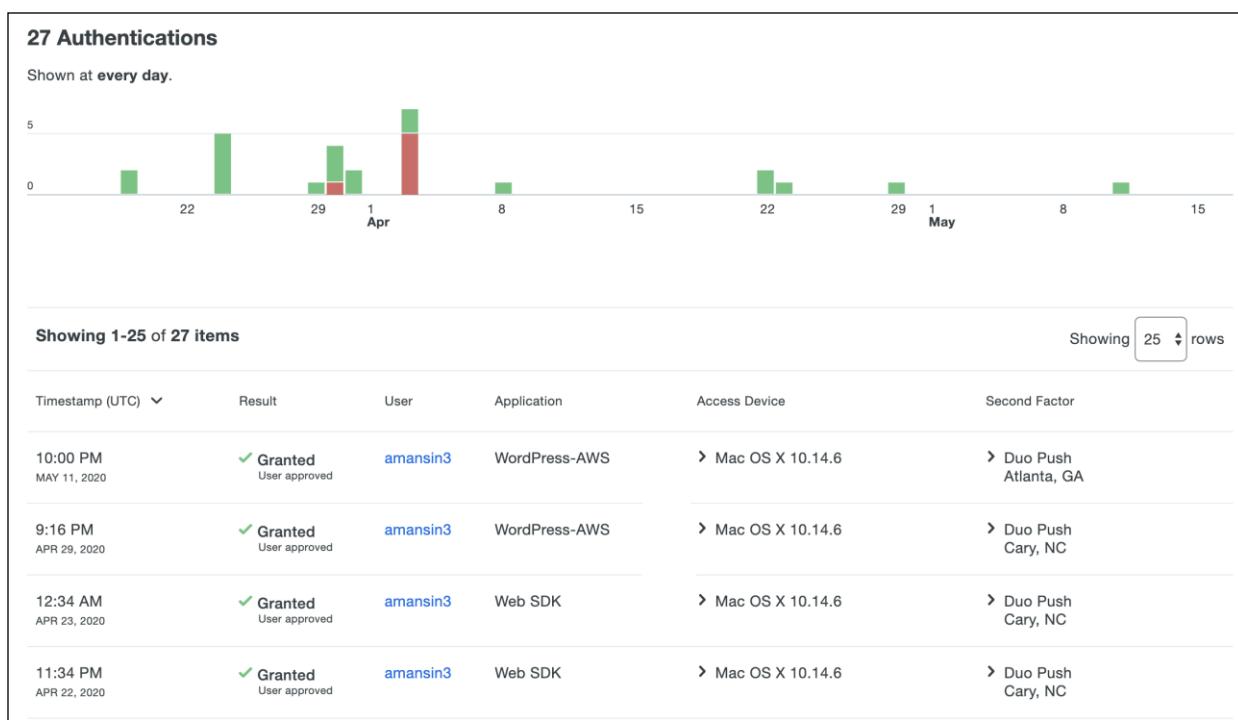


Figure 21.
Duo – User Activity

Cisco SecureX threat response

Cisco SecureX threat response leverages the integrated security architecture to accelerate investigations by automating and aggregating threat intelligence and data across your security infrastructure in one unified view. Some of the key features are:

- Aggregated threat intelligence: Integrates threat intelligence from Cisco TALOS and third-party sources to automatically research indicators of compromise (IOCs) and confirms threats quickly
- Automated enrichment: Automatically adds context from integrated Cisco Security products, so that you instantly know which of your systems was targeted and how
- Incident tracking: Provides the capability you need to collect and store key investigation information, and to manage and document your progress and findings
- Interactive visualizations – Shows your results on intuitive, configurable graphs for better situational awareness and quick conclusions
- Seamless drill down – Makes deeper investigations easy using integrated Cisco Security products. A single click takes you inside Cisco AMP for Endpoints
- Direct remediation – Lets you take corrective action directly from its interface. Block suspicious files, domains, and more without having to log in to another product

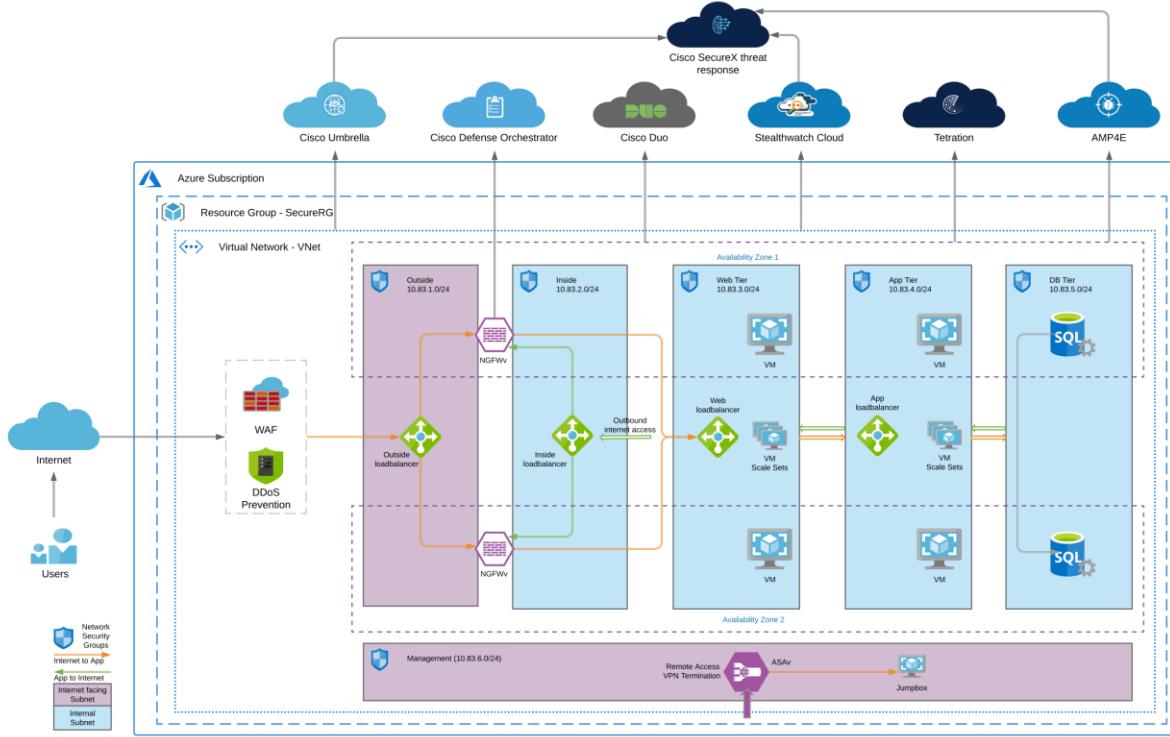


Figure 22.
Cisco SecureX threat response

In this architecture, we are receiving information from Umbrella and AMP Cloud to provide threat intelligence, contextual approach, and threat hunting capabilities. Integrations for Stealthwatch Cloud, Radware Cloud WAF and DDoS service, and Cisco Firepower NGFW are also available. Refer to the [Cisco SecureX threat response](#) documentation for more details on available Cisco and third-party integrations.

Figure 23.
Cisco SecureX threat response - Threat Hunting

Design Implementation

Now that we have established the design specifics of our tiered application in the Azure cloud, we will begin implementing and setting up the lab environment.

We will start by setting up the Azure VNET (Virtual Network) as per the tiered architecture specifications. Once our VNET is ready, we will integrate the Stealthwatch Cloud and set up the Umbrella VAs in the management tier and update the DNS server settings for the VNET.

We will then set up the Azure MySQL database and bring up VSS's (Virtual Machine Scale Sets) for the App and Web workloads followed by setting up the load balancers for each VSS.

Lastly, we will configure the firewalls, enable WAF and DDoS protection and then conclude our set up with Cisco SecureX threat response integration.

Note: Cisco Tetration, AMP, Cisco SecureX threat response, Stealthwatch Cloud, Umbrella, Duo and CDO offer EU based locations for customers having to follow EU rules.

Deployment Overview:

- Set up the Azure Virtual Network
- Integrate Stealthwatch Cloud to Azure environment
- Set up the Umbrella Virtual Appliances
- Set up Azure Database for MySQL servers
- Set up the App and Web Virtual machine Scale Sets (Includes Tetration, AMP4E and Duo Integration)
- Set up the App and Web load balancers
- Set up Cisco Firepower Next-Generation Firewalls and Cisco Defense Orchestrator
- Enable Web Application Firewall and DDoS prevention (Azure and Radware)
- Set up Cisco SecureX threat response

Set up the Azure Virtual Network

In this section, we will create a new Azure VNET and configure all the associated components that we need for our deployment.

Implementation procedure:

Step 1. Create the VNet

Step 2. Create the Network Security Groups

Step 3. Set up the Routing Tables

Step 4. Define the Subnets

Step 1. Create the VNet - Log on to the Azure portal and add a new VNet in the 'Resource Group' and region of your choice. We had created a new 'Resource Group' for this specific implementation, but you can use an existing one. We chose the IPV4 CIDR block as 10.83.0.0/16.

The screenshot shows the Microsoft Azure portal interface for 'Virtual networks'. The top navigation bar includes 'Microsoft Azure', a search bar, and various icons. Below the navigation is a breadcrumb trail: 'Home > Virtual networks'. The main content area is titled 'Virtual networks' with a subtitle 'Cisco-SAFE-Architecture'. It features a table with one record:

Name	Resource group	Location	Subscription
VNet1	SecureRG	East US	Microsoft Azure Enterprise

Filters at the bottom of the table include 'Name' (SecureVNet1), 'Resource group' (SecureRG), 'Location' (all), and 'Subscription' (all). Buttons for 'Add', 'Manage view', 'Refresh', 'Export to CSV', 'Assign tags', and 'Feedback' are also present.

Follow the [Azure documentation](#) for more details on Azure 'Virtual networks'.

Step 2. Create the Network Security Groups - Go to 'Network Security Groups' on the Azure portal, click on 'add' to set up NSGs corresponding to each tier in the design. Set up the access rules as per your application requirements.

Based on our lab environment, we allowed TCP port 80 inbound to 'Outside' tier, to allow internet users to browse the application. Also, we opened TCP port 22 for specific source IPs to allow inbound access to management zone. For all other NSGs, we used the default settings which allow communication within the VNET, inbound load balancer probes, and outbound internet access for any resource deployed in the NSGs.

The screenshot shows the Microsoft Azure portal interface for 'Network security groups'. The top navigation bar includes 'Microsoft Azure', a search bar, and various icons. Below the navigation is a breadcrumb trail: 'Home > Network security groups'. The main content area is titled 'Network security groups' with a subtitle 'Cisco-SAFE-Architecture'. It features a table with 7 items:

Name	Resource group	Location	Subscription
AppNSG-UE	SecureRG	East US	Microsoft Azure Enterprise
DbNSG-UE	SecureRG	East US	Microsoft Azure Enterprise
DiagNSG-UE	SecureRG	East US	Microsoft Azure Enterprise
InNSG-UE	SecureRG	East US	Microsoft Azure Enterprise
MgmtNSG-UE	SecureRG	East US	Microsoft Azure Enterprise
OutNSG-UE	SecureRG	East US	Microsoft Azure Enterprise
WebNSG-UE	SecureRG	East US	Microsoft Azure Enterprise

Filters at the bottom of the table include 'Name' (UE), 'Resource group' (All resource groups), 'Location' (All locations), and 'Subscription' (All tags). Buttons for 'Add', 'Edit columns', 'Refresh', 'Try preview', and 'Assign tags' are also present.

Follow the [Azure documentation](#) for more details on Azure 'Network Security Groups'.

Step 3. Set up the Route tables - Go to 'Route tables' on the Azure portal and create the routing tables for each tier. There is one routing table for each tier. The default routes allow all access among different tiers/subnets and to the outbound internet.

For now, we will leave the routes as default but when we get to firewall deployment, we will update some of these routing tables to force the traffic through our firewall for scanning and inspection.

The screenshot shows the Microsoft Azure portal interface for managing Route tables. The top navigation bar includes the Microsoft Azure logo, a search bar, and various navigation icons. Below the header, the 'Route tables' section is displayed under the 'Cisco-SAFE-Architecture' resource group. A red box highlights the first six items in the list:

Name	Resource group	Location	Subscription
AppRT-UE	SecureRG	East US	Microsoft Azure Enterprise
DbRT-UE	SecureRG	East US	Microsoft Azure Enterprise
InRT-UE	SecureRG	East US	Microsoft Azure Enterprise
MgmtRT-UE	SecureRG	East US	Microsoft Azure Enterprise
OutRT-UE	SecureRG	East US	Microsoft Azure Enterprise
WebRT-UE	SecureRG	East US	Microsoft Azure Enterprise

Follow the [Azure documentation](#) here for more detailed information on Azure traffic routing and route tables.

Step 4. Define the Subnets - At this point we have all the building blocks to define the subnets in our newly created VNET.

Based on the tiered architecture, we defined one subnet for each tier- web, application, database, management, inside and outside.

IPV4 CIDR Block	Azure Region	Tier
10.83.1.0/24	US-East	Database
10.83.2.0/24	US-East	Application
10.83.3.0/24	US-East	Web
10.83.4.0/24	US-East	Management
10.83.5.0/24	US-East	Inside
10.83.6.0/24	US-East	Outside
10.83.7.0/24	US-East	Diagnostic (for NGFWv deployment)

Go to ‘Virtual Networks’ and select the VNET created in Step 1. From the panel on the left-hand side, select ‘Subnets’ and then click on ‘+ Subnet’ to add all the subnets above. Ensure that the subnet created is tied to the appropriate NSG.

The screenshot shows the Microsoft Azure portal interface for managing subnets in a virtual network. The left sidebar has a red box around the 'Subnets' link under the 'Settings' section. The main content area displays a table of subnets with columns for Name, Address range, IPv4 available addresses, Delegated to, and Security group. Each row in the table has a red box around it, and the 'Security group' column values are listed in a separate column to the right.

Name	Address range	IPv4 available addresses	Delegated to	Security group
WebSubnet-UE	10.83.1.0/24	248	-	WebNSG-UE
AppSubnet-UE	10.83.2.0/24	248	-	AppNSG-UE
MgmtSubnet-UE	10.83.4.0/24	245	-	MgmtNSG-UE
DbSubnet-UE	10.83.3.0/24	250	-	DbNSG-UE
OutSubnet-UE	10.83.6.0/24	249	-	OutNSG-UE
InSubnet-UE	10.83.5.0/24	248	-	InNSG-UE
DiagSubnet-UE	10.83.7.0/24	249	-	DiagNSG-UE

Integrating Stealthwatch Cloud

Implementation procedure:

Step 1. Set up the NSG flow logs and integrate Stealthwatch Cloud.

Step 1. Set up the NSG flow logs and integrate Stealthwatch Cloud - Follow the steps illustrated in Cisco Stealthwatch [Azure Quick Start Guide](#) to create the NSG flow logs and push them to the Stealthwatch cloud to be analyzed. Based on this flow log information, the Stealthwatch Cloud would generate security alerts and observations on any activity within the VNET.

After the SWC integration is done, go to ‘Settings > Integrations > Azure > Storage Access’ on SWC portal and you should see green check marks indicating a successful integration.

The screenshot shows the Cisco Stealthwatch Cloud integration configuration page. The left sidebar has a red box around the 'Integrations' link. The main area shows the 'Storage Access' tab selected under the 'Microsoft Azure' section. A red box highlights the 'Storage Access' tab and the 'Flow Logs' and 'Activity Logs' checkboxes, which are both checked and have green checkmarks.

Setting up Cisco Umbrella

Implementation procedure:

Step 1. Set up the Umbrella Virtual Appliance (VA) image

Step 2. Create the Umbrella VA instances

Step 3. Configure the local DNS on Umbrella VA instances

Step 4. Set up the policies to exempt internal domains

Step 5. Update the DNS server settings for VNET

Step 1. Set up the Umbrella Virtual Appliance image - Follow the [Umbrella documentation](#) to deploy Virtual appliances (VA) in the Azure cloud. As per the documentation, create an Azure Image and then use it to launch VA instances.

Step 2. Create the Umbrella VA instances - Create two VA instances using the Azure Image set up in Step1 and place these appliances in the management tier. We assign the static IP addresses 10.83.4.100 and 10.83.4.101 to these Umbrella Virtual Appliances. These VAs will act as DNS forwarders for the resources in our Azure application environment.

Name	Type	Status	Resource group	Location	Source	Maintenance status
UmbrellaVA1	Virtual machine	Running	SecureRG	East US	Image	-
UmbrellaVA2	Virtual machine	Running	SecureRG	East US	Image	-

Once the appliances are fully up in Azure, login to the Umbrella portal and verify the green status under 'Deployments > Configuration > Sites and Active Directory'.

Name	Internal IP	Site	Type	Status	Version
10.83.4.100	10.83.4.100	myazureapp.net	Virtual Appliance	Imported: 22 days ago	2.7.10
10.83.4.101	10.83.4.101	myazureapp.net	Virtual Appliance	Imported: 22 days ago	2.7.10

Optionally, you can create and assign a site name for your Azure VAs. This site name can be used as an identity to configure specific policies for Azure cloud. Click on ‘Settings’ on the same page to add site name and then update the VA entries above.

The screenshot shows the Cisco Umbrella portal interface. On the left, a sidebar lists various management categories like Mobile Devices, Chromebook Users, Network Tunnels, Web Users and Groups, Configuration, Domain Management, and Admin. Under Configuration, 'Sites and Active Directory' is selected and highlighted with a red box. The main content area has a header 'Deployments / Configuration' and 'Sites and Active Directory'. It displays a message to set up Active Directory integration or deploy Virtual Appliances, with a 'Download' button. Below this, there's a 'Sites' tab and an 'Auto-Updates' section. A 'Name' dropdown and a 'Default Site' section are visible. A prominent red box surrounds the 'Site Name' input field, which contains the value 'myazureapp.net'. At the bottom right, there are 'CANCEL' and 'SAVE' buttons, along with pagination controls for 'Page: 1' and 'Results Per Page: 10'.

Step 3. Configure the local DNS on Umbrella Virtual Appliances - Follow the [Umbrella documentation](#) to configure local DNS on each VA. Access to the recursive resolvers in Azure is provided via the virtual IP 168.63.129.16. Set this IP as local DNS on both Umbrella VAs.

Note: We had set up secure remote access to management tier using ASAv, we use the secure VPN connection to SSH into the VAs via a jump server hosted in the management zone.

```
19.83.4.100 $ config va show
Virtual Appliance Configuration
  Name: 19.83.4.100
  Local DNS -
    ip address : 168.63.129.16
  Internal Domains Count: 9
  SSH access : enabled
  HTTPS status : disabled
  SSL Certificate Start Time : NA
  SSL Certificate Expiry Time : NA
  Issuer : NA
  Common Names : NA
  Subject Alternative Name : NA
  Per-IP based Rate Limiting : Disabled
  Primary Adapter : eth0
  MAC Address : 00:0d:3a:9a:2f:8b
  IP Address : 19.83.4.100
  Netmask : 255.255.255.0
  Gateway : 10.83.4.1
```

Step 4. Set up policies to exempt internal domains - Log on to the Umbrella portal, go to ‘Deployments > Configuration > Domain Management’ and add the internal domains that should be routed to the local Azure resolver. Based on your set up, the list of internal domains will vary.

The screenshot shows the Cisco Umbrella web interface. On the left, a sidebar menu includes 'Overview', 'Deployments', 'Core Identities' (with sub-options like 'Networks', 'Network Devices', 'Roaming Computers', 'Mobile Devices', 'Chromebook Users', 'Network Tunnels', 'Web Users and Groups'), 'Configuration' (selected), 'Domain Management' (selected), 'Sites and Active Directory', 'Internal Networks', 'Root Certificate', 'SAML Configuration', 'Service Account Exceptions', 'Policies', and 'Reporting'. The main content area is titled 'Deployments / Configuration' and 'Domain Management'. It displays a message: 'Want to route certain domains to your local resolver? You've come to the right place. Click "Add" above to get started.' Below this, there are two tabs: 'Internal Domains (10)' (selected) and 'External Domains & IPs (0)'. A table lists ten internal domains with their descriptions and applicable sites/devices. At the bottom right are pagination controls ('Page: 1', 'Results Per Page: 10', '1-10 of 10') and navigation arrows.

Domain Name	Description	Applies To
RFC-1918	Non-publicly routable address spaces used only for reverse DNS on internal networks	All Sites, All Devices
local	All *.local domains	All Sites, All Devices
amazonaws.com	AWS Internal	All Sites, All Devices
azure.com	Azure Internal	All Sites, All Devices
compute.internal	AWS Internal	All Sites, All Devices
ec2.internal	AWS Internal	All Sites, All Devices
gateway	Azure gateway	All Sites, All Devices
myazureapp.net	Azure three tier app domain	All Sites, All Devices
mysafeapp.net	AWS three tier app domain	All Sites, All Devices
windows.net	Azure Internal	All Sites, All Devices

Step 5. Update the DNS server settings for VNET - Go to the VNET that we created previously and update the ‘DNS server’ settings. We use the custom DNS option to add the Umbrella VA IP addresses. This will ensure that any instance deployed in this VNET is assigned the Umbrella VAs as DNS forwarders.

The screenshot shows the Microsoft Azure portal interface for managing a virtual network. The top navigation bar includes 'Microsoft Azure', a search bar, and various icons. The main navigation bar shows 'Home > Virtual networks > SecureVNet1 | DNS servers'. The left sidebar lists 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Settings' (with sub-options like 'Address space', 'Connected devices', 'Subnets', 'DDoS protection', 'Firewall', 'Security', 'DNS servers' (selected)), and 'Peering'. The main content area shows the 'SecureVNet1 | DNS servers' configuration page. It features a 'DNS servers' section where 'Custom' is selected, and two IP addresses, '10.83.4.100' and '10.83.4.101', are listed. There are 'Save' and 'Discard' buttons at the top right of this section. At the bottom right, there are buttons for 'Add DNS server' and three dots for more options.

Setting up the Azure Database for MySQL servers

Implementation procedure:

Step 1. Set up Azure database for MySQL server

Step 2. Create a private link to the database

Step 1. Set up Azure Database for MySQL server - We set up an instance of MySQL database, this can vary based on application requirements. Follow the [Azure documentation](#) for help with setting up a MySQL server in Azure.

As part of the initial database set up, you will need to specify an admin username and password. Make sure you remember these credentials; we will need them while setting up the application servers for connections with the database.

Resource group (change) : SecureRG
Status : Available
Location : East US
Subscription (change) : Microsoft Azure Enterprise
Subscription ID : [REDACTED]
Tags (change) : Click here to add tags

Resource utilization (securemysqldb)

Time	Utilization (%)
12:45 PM	0.0%
1 PM	0.0%
1:15 PM	0.0%
1:30 PM	0.0%

Step 2. Create a Private Endpoint to the database – Go to the newly created database under ‘Azure Database for MySQL servers’ and select ‘Private endpoints connections’ from the panel on the left-hand side. Click on ‘+ Private endpoint’ to create a private endpoint link to this newly created MySQL database server.

Azure Private Link enables you to privately access Azure PaaS Services. This ensures that the database is not exposed to public access. This link would be set up via the database subnet and hence database NSG rules would apply. As per the default settings, inbound access from all Azure resources and unfiltered outbound access to Internet is allowed. These rules should be fine tuned to make sure that only the application subnet is allowed access to database.

Connection name	State	Private endpoint name	Request/Response Message
mysqlbpvteast-df36eb87-ad9d-46d8-935a-e52ff3745230	Approved	mysqlbpvteast	Auto-approved

A Private link allows DNS integration to assign an FQDN to the database server instance. This FQDN would resolve to a dynamically assigned IP address in database subnet, enabling access to the database server. Make sure you keep a note of this FQDN along with the admin username and password configured in Step 1.

You can go to the ‘Private Link Center’ to get detailed information about the newly created private link.

The screenshot shows the Azure portal interface for managing private endpoints. The main pane displays the details of a private endpoint named 'mysqldbpvtest'. Key information includes:

- Resource group (change):** SecureRG
- Location:** East US
- Subscription (change):** Microsoft Azure Enterprise
- Provisioning state:** Succeeded
- Virtual network/subnet:** SecureVNet1/DbSubnet-UE
- Network interface:** mysqldbpvtest.nic.791b37c5-60a6-4596-ad25-48a3d7dbd44a
- Private link resource:** securemysqldb
- Target sub-resource:** mysqlServer
- Connection status:** Approved
- Request/Response:** Auto-approved

In the 'Custom DNS settings' section, the FQDN is listed as 'securemysqldb.mysql.database.azure.com'.

Check out the [Azure documentation](#) to learn more about private links.

Setting up the Virtual Machine Scale Sets

In this section, we will set up a pool of workloads for the web and application tier. We will use the Azure Virtual Machine Scale Set (VSS) service to achieve this. As part of workload initialization, we install Tetration and AMP4E agents on the web and app workloads along with other application-specific packages, including the Duo plugin.

Implementation procedure:

Step 1. Create VSS for app and web servers

Step 2. Install Azure VM extension for workload initialization

Step 1. Create VSS for app and web servers - Go to ‘Virtual Machine Scale Sets’ in the Azure portal and click on ‘add’ to create a VSS for both web and app workloads in the Resource Group created for application.

- The geographical regions remain the same, i.e. US East
- For availability zone, make sure you check the required number of zones. Since this design is zone redundant with two availability zones, we selected zone 1 and zone 2 in US East region. This ensures that when workloads are launched in the VSS they are distributed across the two zones
- For the base image/operating system we chose CentOS, we later used custom scripts to initialize them with required packages, code and security agents. You could also use a golden image with all the required applications and packages pre-installed to skip the custom scripts. For the purpose of this document, we have validated the custom scripts option
- Under networking options, choose appropriate subnets for app and web VSS (app subnet for app VSS and web subnet for web VSS without any public IPs). Leave the Network Security Group to basic settings since we use the NSGs at subnet level.

Virtual machine scale sets

Subscriptions: Microsoft Azure Enterprise – Don't see a subscription? [Open Directory + Subscription settings](#)

Name	Status	Instances	Azure Sp...	Resource group	Location	Subscription
AppScaleSet-East	All succeeded	2	-	securerg	East US	Microsoft Azure Enterprise
WebScaleSet-East	All succeeded	2	-	SecureRG	East US	Microsoft Azure Enterprise

For detailed information on all other parameters regarding Virtual Machine Scale Set creation, check out the [Azure documentation](#).

In the snapshot below for app and web VSS, you can see the location set to US east regions with redundancy in zone 1 and zone 2.

AppScaleSet-East

Overview

Resource group (change) : **securerg**
Status : 2 out of 2 succeeded
Location : East US (Zones 1, 2)
Subscription (change) : Microsoft Azure Enterprise
Subscription ID : **[REDACTED]**
Fault domains : 5
Tags (change) : Click here to add tags

Public IP address : -
Public IP address (IPv6) : -
Virtual network/subnet : SecureVNet1/AppSubnet-UE
Operating system : Linux
Size : Standard_B1ms (2 instances)
Ephemeral OS disk : Not applicable
Autoscaling : Off
Azure Spot : N/A

WebScaleSet-East

Overview

Resource group (change) : **SecureRG**
Status : 2 out of 2 succeeded
Location : East US (Zones 1, 2)
Subscription (change) : Microsoft Azure Enterprise
Subscription ID : **[REDACTED]**
Fault domains : 5
Tags (change) : Click here to add tags

Public IP address : -
Public IP address (IPv6) : -
Virtual network/subnet : SecureVNet1/WebSubnet-UE
Operating system : Linux
Size : Standard_B1ms (2 instances)
Ephemeral OS disk : Not applicable
Autoscaling : Off
Azure Spot : N/A

At this point, we have two base workloads each, in app and web VSS. We haven't set up the application on these workloads yet, we will do that in our next step.

Microsoft Azure

Home > Virtual machine scale sets > AppScaleSet-East | Instances

AppScaleSet-East | Instances

Name	Status	Health state	Protection policy
AppScaleSet-East_25	Running	Green	
AppScaleSet-East_26	Running	Green	

Microsoft Azure

Home > Virtual machine scale sets > WebScaleSet-East | Instances

WebScaleSet-East | Instances

Name	Status	Health state	Protection policy	Latest model
WebScaleSet-East_0	Running	Green	Yes	
WebScaleSet-East_1	Running	Green	Yes	

Step 2. Install Azure VM extension for workload initialization – We set up a ‘Storage Account’ in US East region, same region as our application location. We upload all the files that we need for our application and web workloads into this storage account. If you choose to host config files in this manner, ensure that you set appropriate access privileges for these files.

These files include:

- Shell scripts (appservers and webservers) for initializing the application and web workloads
- Config files for our application and web workloads
 - App workloads – App configuration file has information about database connection (we recorded the details for this while setting up database previously).
 - Web workloads – Along with various other details, the web config file has the IP address of app VSS load balancer. We set apart the IP- 10.83.2.100 for the app load balancer. We haven’t created the load balancers yet; we will do that in the next step and assign them the IPs
- AMP4E agent installer (AMP rpm and GPG). These were obtained from AMP portal
- Tetration agent installer (Enforcer and visibility). These are downloaded from Tetration portal

The initialization shell scripts perform the following tasks:

- Install packages (php, wget, unzip, lsof, httpd, ipset, nginx) on the workloads. Some of these are prerequisites for AMP4E and Tetration agents. Refer to the corresponding product documentation to understand these requirements.

- Download configuration files/code for the respective workloads.
 - For the application server, we used a publicly available ‘[WordPress](#)’ blog code. We download the zipped code from the WordPress site and then update the database config file from Azure Storage Account we set up above.
 - For web server, download the modified web configuration file from Azure Storage Account we set up above.
- Download and unzip the Duo plugin (available for ‘[WordPress](#)’ blogs) to the application workloads.
- Download and install the Tetration enforcement agent from Azure Storage Account we set up above.
- Download and install the AMP4E agent from Azure Storage Account we set up above.

Note: If you choose to include the Duo integration in your native application, follow the [DUO Web SDK](#) documentation.

Below are sample initialization scripts that we used.

Web server initialization script:

```
#!/bin/bash
sudo yum install -y wget
sudo yum install -y unzip
sudo yum install -y lsof      // lsof utility is required for enforcing tetration policies
sudo yum install -y ipset     // ipset utility is required for enforcing tetration policies
sudo yum install -y nginx    // Installing nginx

#Setting up the web server and updating it with hosted configuration file.
sudo mv nginx.conf nginx.conf.backup
sudo wget https://safelabfiles.blob.core.windows.net/config/nginx.conf
sudo systemctl restart nginx
sudo systemctl enable nginx

#Downloading the Tetration enforcement agent from Azure Storage account and installing it.
sudo wget https://safelabfiles.blob.core.windows.net/config/tetration_installer_intgssopov_enforcer_linux.sh
sudo chmod 755 tetration_installer_intgssopov_enforcer_linux.sh
sudo ./tetration_installer_intgssopov_enforcer_linux.sh --skip-pre-check

#Downloading the AMP4E agent hosted in an Azure Storage account and installing it.
sudo wget https://safelabfiles.blob.core.windows.net/config/cisco.gpg
sudo rpm --import ./cisco.gpg
sudo wget https://safelabfiles.blob.core.windows.net/config/AWS_rhel-centos-7fireAMPLinux_connector.rpm
sudo yum install -y AWS_rhel-centos-7fireAMPLinux_connector.rpm
```

Application server initialization script:

```

#!/bin/bash

sudo yum install -y wget
sudo yum install -y unzip
sudo yum install -y lsof      // lsof utility is required for enforcing tetration policies
sudo yum install -y ipset     // ipset utility is required for enforcing tetration policies
sudo yum install -y httpd    // Installing httpd

#Setting up the HTTPD server and downloading the application code and Duo plugin hosted in
#Azure Storage Account.

sudo systemctl start httpd
sudo systemctl enable httpd
sudo setsebool -P httpd_can_network_connect 1      //Allow outbound connections from HTTPD
daemon

sudo wget https://safelabfiles.s3.us-east-2.amazonaws.com/wordpresscodefile.zip -P
/var/www/html/
sudo unzip /var/www/html/wordpresscodefile.zip
sudo wget https://downloads.wordpress.org/plugin/duo-wordpress.2.5.4.zip -P
/var/www/html/wp-content/plugins
sudo unzip /var/www/html/wp-content/plugins/duo-wordpress.2.5.4.zip
cd /home/centos/
sudo systemctl restart httpd

#Downloading the Tetration enforcement agent from Azure Storage Account and installing it.

sudo wget
https://safelabfiles.blob.core.windows.net/config/tetration_installer_intgssopov_enforcer_li
nux.sh
sudo chmod 755 tetration_installer_intgssopov_enforcer_linux.sh
sudo ./tetration_installer_intgssopov_enforcer_linux.sh --skip-pre-check

#Downloading the AMP4E agent hosted in an Azure Storage Account and installing it.

sudo wget https://safelabfiles.blob.core.windows.net/config/cisco.gpg
sudo rpm --import ./cisco.gpg
sudo wget https://safelabfiles.blob.core.windows.net/config/AWS_rhel-centos-
7fireamplinux_connector.rpm
sudo yum install -y AWS_rhel-centos-7fireamplinux_connector.rpm

```

The screenshot shows the Azure Storage Explorer interface. On the left, there's a sidebar with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Data transfer, Events, and Storage Explorer (preview). The main area shows a blob container named 'config'. Inside 'config', there are subfolders for BLOB CONTAINERS, FILE SHARES, QUEUES, and TABLES. The BLOB CONTAINERS folder is selected and highlighted with a red box. Inside it, there's a single folder named 'config' which is also highlighted with a red box. To the right of the file structure is a table listing files with columns for NAME, ACCESS TIER, ACCESS TIER LAST MODIFIED, and LAST MODIFIED. A red box highlights the first few rows of this table.

NAME	ACCESS TIER	ACCESS TIER LAST MODIFIED	LAST MODIFIED
appserverEast.sh	Hot (inferred)	4/24/2020, 5:19:41 PM	3/25/2020, 10:51:49 PM
AWS_rhel-centos-7fireamplinux_connector.rpm	Hot (inferred)	3/25/2020, 10:51:28 PM	3/27/2020, 4:25:25 PM
cisco.gpg	Hot (inferred)	3/25/2020, 10:51:28 PM	3/25/2020, 10:51:28 PM
nginx.conf	Hot (inferred)	3/25/2020, 10:51:28 PM	3/25/2020, 10:51:28 PM
tetration_installer_intgssopv_enforcer_linux.sh	Hot (inferred)	3/25/2020, 10:51:28 PM	3/25/2020, 10:51:28 PM
tetration_installer_intgssopv_sensor_linux.sh	Hot (inferred)	3/25/2020, 10:51:28 PM	3/25/2020, 10:51:28 PM
webserverEast.sh	Hot (inferred)	4/24/2020, 5:19:53 PM	3/25/2020, 10:51:28 PM
wp-config.php	Hot (inferred)	3/25/2020, 10:51:28 PM	

Now that we have all the files hosted in Azure Storage Account, we will install the extensions. Go to ‘Virtual Machine Scale Sets > [app VSS] > Extensions’ and click on add. Choose ‘Custom Script for Linux’ from the panel on the left-hand side. Create the extension by choosing the web shell script from the Azure Storage Account we created earlier.

The screenshot shows the Azure Virtual Machine Scale Sets interface. The URL in the address bar is 'Home > Virtual machine scale sets > AppScaleSet-East | Extensions > Extensions'. The main area displays the details of an existing extension named 'AppScaleSet-East'. It shows settings like Type (CustomScript), Version (2.0), Publisher (Microsoft.Azure.Extensions), and Auto upgrade minor version (true). Under Settings, there is a JSON configuration block:

```
{
  "fileUris": [
    "https://safeconfigfileseast.blob.core.windows.net/config/appserverEast.sh?sp=r&sse=2020-05-12T05:21:47Z&sv=2019-10-10&sr=b&sig=Skr3Rs3UAfZJpFPIVdvGqtVxrLTKOy1aEibU7nQ%2FeZuI%3D"
  ]
}
```

After adding the extension, go back to VSS and make sure you select all the running instances and click on upgrade to finish the installation.

The screenshot shows the Azure Virtual Machine Scale Sets interface. The URL in the address bar is 'Home > Virtual machine scale sets > WebScaleSet-East | Extensions > Extensions'. The main area displays the details of an existing extension named 'WebScaleSet-East'. It shows settings like Type (CustomScript), Version (2.0), Publisher (Microsoft.Azure.Extensions), and Auto upgrade minor version (true). Under Settings, there is a JSON configuration block:

```
{
  "fileUris": [
    "https://[REDACTED].blob.core.windows.net/vmextensionstemporary-10033ffa61c5539-20200401192752283/webserverEast.sh?sv=2017-04-17&sr=c&sig=wGBXxVYwRlzoH1rqmjWO2eBbqnEE4zDh5x2X0Vq2m%2FE%3D&se=2020-04-02T19%3A27%3A52Z&sp=r"
  ]
}
```

Repeat the same steps for web VSS.

Setting up the App and Web load balancers

At this point we have a highly available pool of web and app workloads running in the corresponding Virtual Machine Scale Sets in web and app tier. Now, we need to create load balancers to load balance traffic onto these two Virtual Machine Scale Sets. We will create a web load balancer in web tier which receives web traffic from users and then load balances it onto the web servers running in web VSS in the Web tier. We will also create an app load balancer sitting in front of the app workloads in the App tier. When the Nginx web servers in web VSS receive web requests from web load balancer, they will forward them to app load balancer to be distributed onto the app workloads in app VSS.

Go to ‘Azure load balancer’ service and click on ‘add’ to create the web and app load balancer. Make sure you select the type as ‘Internal’ and SKU as ‘Standard’. We need ‘Standard’ SKU to be able to integrate these load balancers with the zone redundant Virtual Machine Scale Sets.

The screenshot shows the Microsoft Azure portal interface for managing load balancers. The top navigation bar includes the Microsoft Azure logo, a search bar, and various navigation icons. Below the header, the URL 'Home > Load balancers' is visible. The main content area is titled 'Load balancers' and shows a list of existing resources. A red box highlights the first two items in the list: 'AppLB-East' and 'WebLB-East'. The table columns include 'Name', 'Resource group', 'Location', and 'Subscription'. Both load balancers are listed under the 'SecureRG' resource group, located in 'East US', and belong to the 'Microsoft Azure Enterprise' subscription.

Name	Resource group	Location	Subscription
AppLB-East	SecureRG	East US	Microsoft Azure Enterprise
WebLB-East	SecureRG	East US	Microsoft Azure Enterprise

We assigned these load balancers with static IPs 10.83.1.100 and 10.83.2.100 in web and app subnet respectively. We had specified 10.83.2.100 in our web server configuration file in the previous step; hence web servers would make the requests to app load balancer IP if they need to fetch any dynamic content from app servers.

The screenshot shows the detailed view of the 'AppLB-East' load balancer. The top navigation bar shows the full path: 'Home > Load balancers > AppLB-East'. The main content area has a left sidebar with navigation links: Overview (selected), Activity log, Access control (IAM), Tags, Diagnose and solve problems, and Settings. The main panel displays various configuration settings for the load balancer. A red box highlights the 'Private IP address' field, which is set to '10.83.2.100'. Other visible settings include: Resource group (SecureRG), Location (East US), Subscription (Microsoft Azure Enterprise), Backend pool (AppServerPool-East), Health probe (TCPProbe), Load balancing rule (AppLBRule), and NAT rules (0 inbound).

Resource group (change)	: SecureRG
Location	: East US
Subscription (change)	: Microsoft Azure Enterprise
Subscription ID	: [REDACTED]
SKU	: Standard
Tags (change)	: Click here to add tags
Private IP address	: 10.83.2.100

WebLB-East
Load balancer

Resource group (change) : SecureRG
Location : East US
Subscription (change) : Microsoft Azure Enterprise
Subscription ID : [\[REDACTED\]](#)
SKU : Standard
Tags (change) : Click here to add tags

Backend pool : WebVSS (2 virtual machines)
Health probe : TCPHealth (Tcp:80)
Load balancing rule : WebLBRule (Tcp:80)
NAT rules : 0 inbound
Private IP address : 10.83.1.100

To finish our load balancer configuration, we need to add the backend pool, health probes and load balancing rules. For backend pools, we will add corresponding VSS for each load balancer. Make sure to upgrade VSS instances once the load balancer configuration is finished.

Refer to the [Azure documentation](#) for detailed steps on how to set up internal load balancers. Once the backend pool, the health probes and load balancing rules as set up, the app and web VSS instances would show as below under the load balancer service.

AppLB-East | Backend pools

Backend pool	Virtual machine	Virtual machine status	Network interface	Private IP address
AppServerPool-East (2 virtual machines)	AppScaleSet-East (instance 26)	Running	SecureVNet1-nic01	10.83.2.5
	AppScaleSet-East (instance 25)	Running	SecureVNet1-nic01	10.83.2.4

WebLB-East | Backend pools

Backend pool	Virtual machine	Virtual machine status	Network interface	Private IP address
WebVSS (2 virtual machines)	WebScaleSet-East (instance 1)	Running	SecureVNet1-nic01	10.83.1.5
	WebScaleSet-East (instance 0)	Running	SecureVNet1-nic01	10.83.1.4

Setting up the Firepower Next-Generation Firewalls

In this section of the document, we will set up a pair of Cisco Firepower Next-Generation Firewalls at the network perimeter and onboard them to Cisco Defense Orchestrator for management. You could also use Cisco FMC (Firepower Management Center) available via the Azure marketplace (or an on-premise FMC) for management purposes.

Once the firewalls are set up, we will expose our application to public Internet via an External Azure load balancer. We will also set up an Internal Azure load balancer and update the App and Web tier routing tables to load-balance the outbound traffic from workloads to the Next-Generation Firewalls.

Implementation procedure:

Step 1. Deploy NGFWv using Azure Templates

Step 2. Onboard the NGFWv to CDO

Step 3. Configure interfaces, routes, health probes, NAT and access control on NGFWv

Step 4. Set up the inside and outside load balancers

Step 5. Access the application

Step 1. Deploy NGFWv using Azure templates – Download the NGFW template available at [GitHub](#). Go to ‘Azure Templates’ service on the Azure portal and add this template to deploy two Cisco Firepower Next-Generation Firewalls. Once you click on deploy, you will need to provide the IP addresses; we used the following IP addressing.

NGFWv	Interface Name	IPV4 Address	Azure NIC	Firepower Interface
Safengfw1	Management	10.83.4.10	NIC0	Management
Safengfw1	Diagnostic	10.83.7.10	NIC1	Diagnostic
Safengfw1	Inside	10.83.5.10	NIC2	Gig0/0
Safengfw1	Outside	10.83.6.10	NIC3	Gig0/1
Safengfw2	Management	10.83.4.11	NIC0	Management
Safengfw2	Diagnostic	10.83.7.11	NIC1	Diagnostic
Safengfw2	Inside	10.83.5.11	NIC2	Gig0/0
Safengfw2	Outside	10.83.6.11	NIC3	Gig0/1

Other details that you would be prompted for, while deploying the FTDs, include Resource Group (same as our cloud application), location (US East), hostname (safengfw1 and safengfw2), access credentials, storage accounts for firewalls (Storage-General purpose V1/LRS), Virtual Network (same VNET that we created earlier for the web application) and an Availability Set.

You will need to create an availability set prior to deploying this template and specify it when prompted. Refer to [Azure documentation](#) for information on how to create an Availability Set.

```

1 {
2   "$schema": "http://schema.management.azure.com/schemas/2015-0
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "vm1Name": {
6       "type": "string",
7       "defaultValue": "ngfw",
8       "metadata": {
9         "description": "Name of the first NGFW VM"
10      }
11    },
12  },
13 },
14 },
15 },
16 },
17 },
18 },
19 },
20 },
21 },
22 }

```

You can also customize the GitHub template to deploy the FTDs in different Availability Zones instead of an Availability Set. You will need to ‘Edit’ the JSON template file to remove Availability Set configuration and specify the Availability Zone for each NGFWv. Most of this lab validation was carried out using Availability Set, but we also validated the Availability Zones.

Note: An Availability Set ensures that the firewalls are deployed on separate hardware. The Availability Zones (AZ) on the other hand are two separate physical locations in the same region, connected with a high-speed link.

The sample template config snippet to add Availability Zones is as below:

```

"apiVersion": "2017-12-01",
"type": "Microsoft.Compute/virtualMachines",
"name": "[parameters('vm1Name')]",
"zones": [ "1"],

"apiVersion": "2017-12-01",
"type": "Microsoft.Compute/virtualMachines",
"name": "[parameters('vm2Name')]",
"zones": [ "2"],
```

You will also need to remove the following Availability Set config from the downloaded [GitHub](#) template:

Parameters Section:

```

"availabilitySetResourceGroup": {
    "type": "string",
    "defaultValue": "",
    "metadata": {
        "description": "Name of the existing Availability Set's Resource Group"
    }
},
"availabilitySetName": {
    "type": "string",
    "defaultValue": "",
    "metadata": {
        "description": "Name of the existing Availability Set"
    }
},
```

Resources Section:

```

"availabilitySet": {
    "id": "[resourceId(parameters('availabilitySetResourceGroup'), 'Microsoft.Compute/availabilitySets/',
    parameters('availabilitySetName'))]"
},
```

After a successful template deployment, the FTDs would show as below.

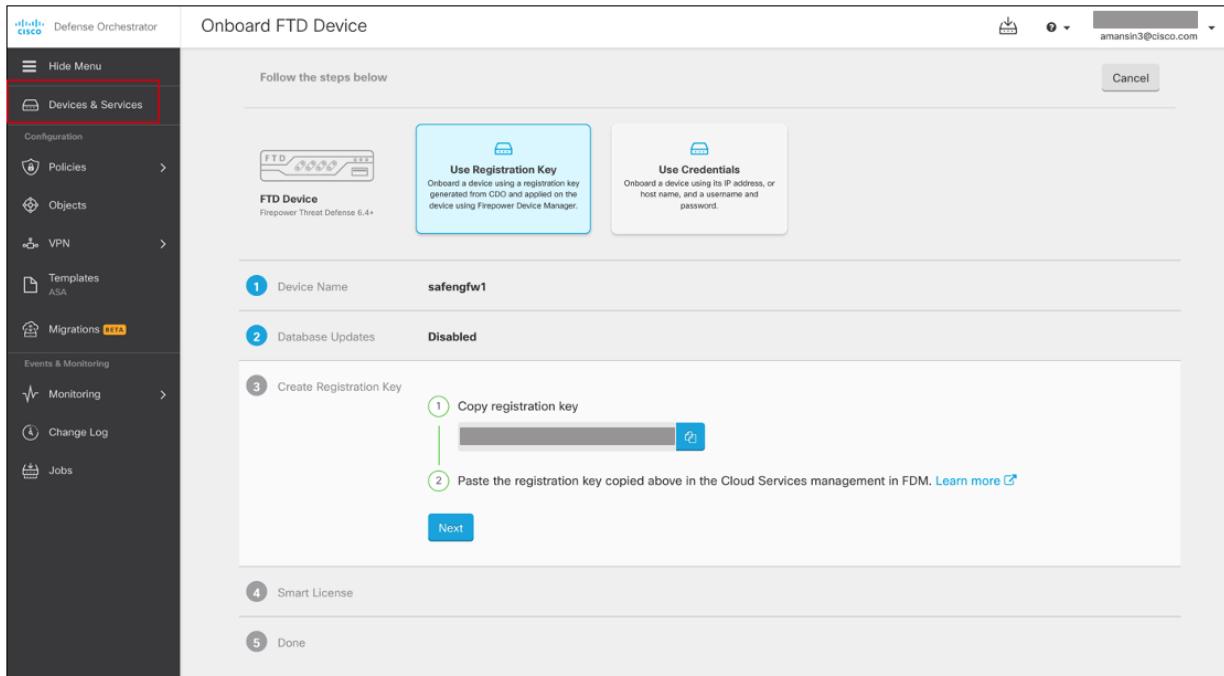
Name	Type	Status	Resource group	Location	Source	Subscription
safengfw1	Virtual machine	Running	SecureRG	East US	Marketplace	- Microsoft Azure Enterprise
safengfw2	Virtual machine	Running	SecureRG	East US	Marketplace	- Microsoft Azure Enterprise

Each Firewall should have four network interfaces, assigned with IP addresses as specified in template while deploying.

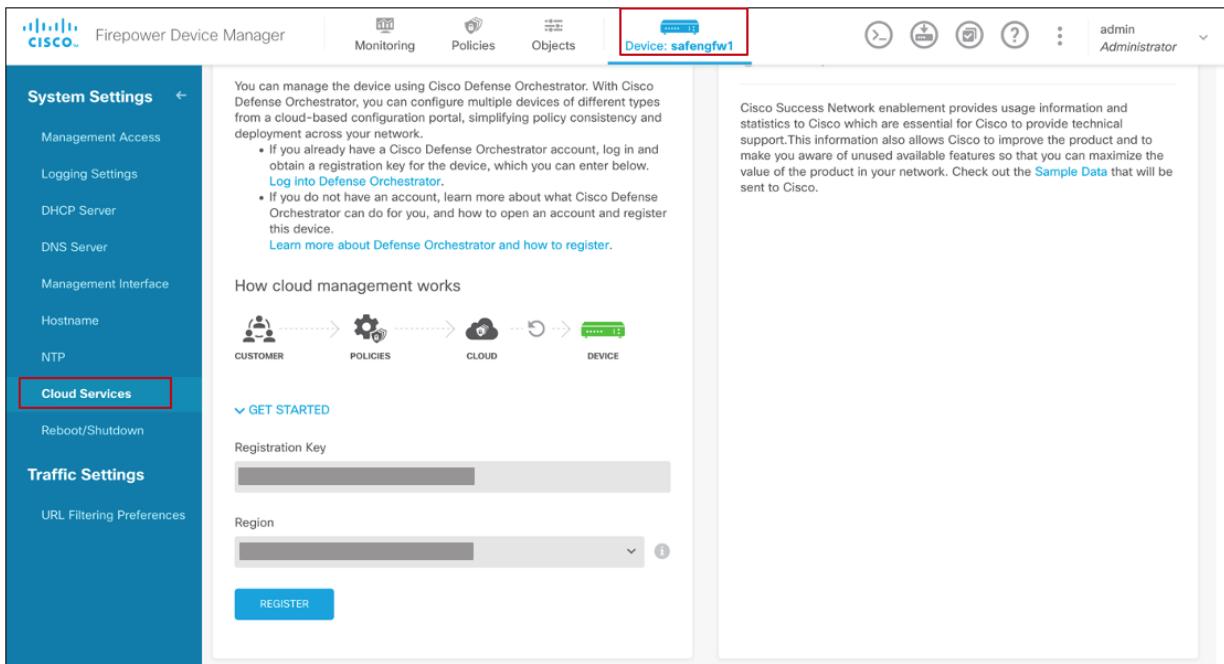
The Management interface will be assigned a public IP address automatically, we will use this IP to do the initial set up of these newly deployed instances. Additionally, associate a public IP to NIC3 (outside interface) of each firewall manually. Follow the [Azure documentation](#) for instructions.

Step 2. Onboard the NGFWv to CDO - Access the Firepower Device Manager (FDM) using the management IP address. Click on skip device set up and acknowledge the 90-day trial license warning (we will configure smart licensing in subsequent steps). You will land at the following page.

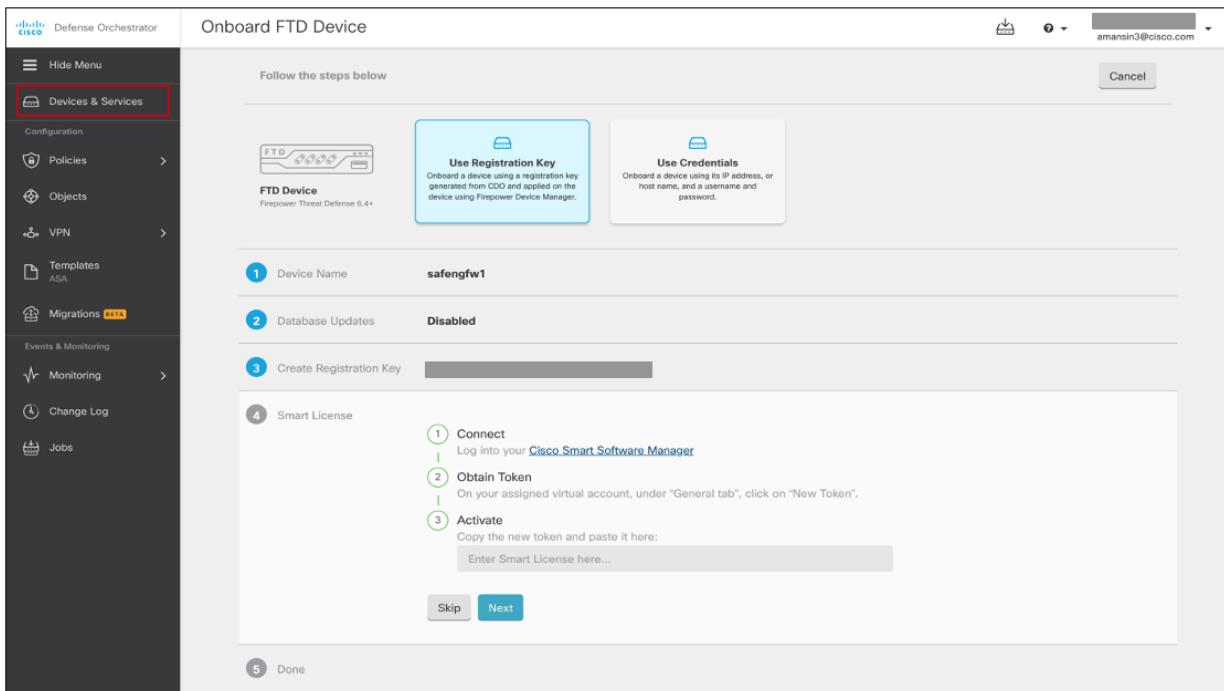
Log on to the CDO web portal and go to ‘Devices and Services’ and click on plus button on the top right-hand side to onboard a Firepower Threat Defense (FTD) device. We use ‘Registration Key’ option to onboard the FTD. Fill in the name of the FTD device and follow the wizard to copy the registration key.



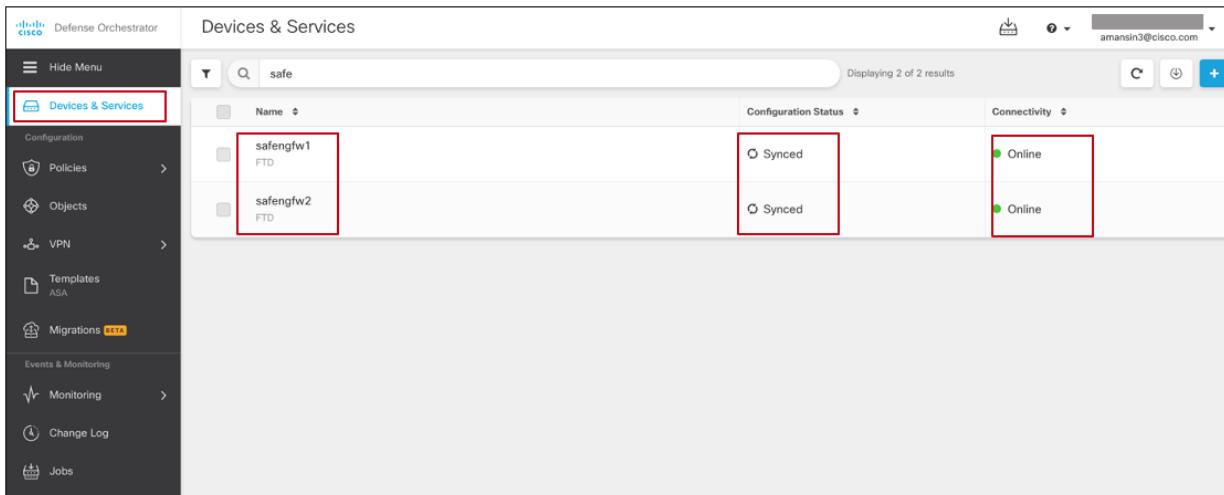
Go back to FDM portal and go to ‘Cloud Services’ option under ‘System Settings’. Paste the registration key, specify the ‘Region’ and click on register.



Come back to the CDO portal, click on next and log into your Cisco smart licensing manager and generate the token. Paste the token and click on next to finish the onboarding process.



Repeat the same steps for the other firepower device, CDO will sync the configuration for both the devices. At this point we have successfully finished onboarding both the FTDs to CDO.



Step 3. Configure interfaces, routes, HTTPS health probes, NAT and access control on NGFWv - We need to configure the following five components:

- Network interfaces
- Static routes
- Enable HTTPS on inside and outside interface for load balancer probes
- NAT rules for inbound and outbound traffic
- Access rules to allow the necessary traffic

Click on the onboarded NGFW appliance on CDO dashboard and then go to ‘Interfaces’ option from the menu that appears on the right-hand side of the dashboard. Make sure Gig0/0 and Gig0/1 are assigned static IP addresses and names as defined in the table in Step 1.

Name	Logical Name	State	IP Address	Interface Type	Mode
GigabitEthernet0/0	inside	Enabled	10.83.5.10 STATIC	Physical Interface	Routed
GigabitEthernet0/1	outside	Enabled	10.83.6.10 STATIC	Physical Interface	Routed
Management0/0	diagnostic	Enabled		Physical Interface	Routed

Go back to the same menu and now go to ‘Routing’ option, set the default route pointing to the gateway on outside the subnet- 10.83.6.1. Also, add the route for internal subnets (web, app and database subnet) pointing to the gateway on the inside subnet- 10.83.5.1. Lastly, add a route for load balancer virtual IP address that sends the health probes, set the next hop as the inside subnet gateway. Find more details about Azure load balancer health probes, refer to the [Azure documentation](#).

Name	Interface	IP Type	Destination Networks	Gateway IP	Metric
Default	outside	IPv4	any-ipv4 0.0.0.0	outsideGW 10.83.6.1	1
insideSubnets	inside	IPv4	appSubnet 10.83.2.0/24 dbSubnet 10.83.3.0/24 inLBprobes 168.63.129.16/32 webSubnet 10.83.1.0/24	InsideGW 10.83.5.1	1

Next, we set up the inside and outside interfaces to listen on TCP port 443; we make sure that we only allow load balancer probe IP. We will need this allowed for inside and outside load balancers to do health check on inside and outside interfaces of the firewalls. Go back to the same device menu on CDO and this time, click on ‘Settings’. Select the ‘Data Interfaces’ tab and click on ‘add’ to allow probes to inside and outside interfaces on port 443 from VIP sending the load balancer health probes. You might also need to update your inside NSG to allow these probes.

Protocol	Allowed Networks	Action
HTTPS	any-ipv4	<input checked="" type="checkbox"/>
HTTPS	any-ipv6	<input checked="" type="checkbox"/>
SSH	any-ipv4	<input checked="" type="checkbox"/>
SSH	any-ipv6	<input checked="" type="checkbox"/>

Interface	Protocols	Allowed Networks	Action
inside	HTTPS	inLBprobes	<input checked="" type="checkbox"/>
outside	HTTPS	inLBprobes	<input checked="" type="checkbox"/>

Next, we set up two NAT rules. We configure a dynamic PAT rule to allow outbound traffic to the Internet from the application workloads. The translation would be as below.

Source: `WorkloadIP:Port => OutsideFWInterfaceIP:Port`

We configure another static NAT rule to do the following translation for traffic from the Internet users accessing the application:

Source translation: InternetUserIP:Port => InsideFWInterfaceIP:Port

Destination translation: OutsideLBIP:HTTP => WebLBIP:HTTP

We need the source translation above to ensure that reply traffic is returned back to the same firewall's inside interface i.e. maintain the traffic symmetry. The destination is translated from public IP of outside load balancer to the IP address of the web load balancer i.e. 10.83.1.100. Web load balancer receives all the incoming traffic for the cloud application and distributes to the web servers in web VSS.

Name	Type	Source Interface	Destination Interface	Original Packet	Translated Packet		
		Source	Destination	Service	Source	Destination	Service
Twice NAT							
OutsideToInside	Static	outside	inside	interface	HTTP	interface	WebLB
InsideToOutside	Dynamic	inside	outside			interface	HTTP

Lastly, we assign inside interface to inside zone and outside interface to outside zone. Then, we configure access control policies to allow traffic from inside zone to outside zone. We also allow HTTP traffic incoming from the Internet users who are trying to access the cloud application.

Go to 'Policies > FTD Rulesets' on CDO portal and click on plus button to add a an FTD ruleset. Within this newly created ruleset, add the access rules and attach it to the newly onboarded FTD appliances.

#	Name	Action	Source	Destination	Layer
1	InsideToOutside	Trust	[ZONES] Inside_zone	[ZONES] outside_zone	Any
2	OutsideToInside	Allow	[ZONES] outside_zone	[ZONES] Inside_zone [NETS] WebLB [PORTS] HTTP	Any

After this, we deploy all the changes and repeat the same for the second FTD device.

Step 4. Set up the 'inside' and 'outside' load balancers - In this step, we will set the 'inside' and 'outside' load balancers and the update the Azure Route Tables for web and app tier to point the default route to the 'inside' load balancer.

Go to 'Load balancer' service on the Azure portal. Click on 'add' to create an 'inside' load balancer. We assigned the static IP address 10.83.5.100 to the load balancer.

Refer to the [Azure documentation](#) for detailed information on creating load balancer. Make sure to choose 'Standard' SKU, it's required to integrate with highly available (Zone Redundant) resources like our firewall deployment.

The screenshot shows the Microsoft Azure portal interface for managing load balancers. The left sidebar lists 'Load balancers' under 'Cisco-SAFE-Architecture'. The main panel shows 'InsideLB' selected. The 'Overview' section displays resource group (SecureRG), location (East US), subscription (Microsoft Azure Enterprise), and SKU (Standard). It also shows a 'Backend pool' (ngfw), 'Health probe' (HTTPMgmt), 'Load balancing rule' (InsideLBrule), and 'NAT rules' (0 inbound). A note at the bottom encourages configuring high availability and scalability.

After the load balancer is created, add the backend pool. Click on the add button and add the inside firewall interfaces to the pool.

The screenshot shows the 'Backend pools' section for 'InsideLB'. It lists two virtual machines: 'safengfw1' and 'safengfw2', both running and assigned private IP addresses 10.83.5.10 and 10.83.5.11 respectively. The 'ngfw' backend pool is selected.

Backend pool	Virtual machine	Virtual machine status	Network interface	Private IP address
ngfw (2 virtual machines)	safengfw1	Running	safengfw1-nic2	10.83.5.10
	safengfw2	Running	safengfw2-nic2	10.83.5.11

After adding the backend pool, next step is to define the probes to monitor the health status of inside interfaces. We had set up the FTD appliances to listen on port 443 from load balancer IP addresses, hence we set up the health probe for TCP port 443.

The screenshot shows the 'Health probes' section for 'InsideLB'. It lists a single probe named 'HTTPMgmt' which uses TCP protocol on port 443 and is used by the 'InsideLBrule' load balancing rule.

Name	Protocol	Port	Used By
HTTPMgmt	TCP	443	InsideLBrule

Last step is to configure the load balancing rule. For 'inside' load balancer, make sure to check the HA port option so that the load balancer can receive traffic for all the TCP and UDP ports.

AllTrafficLBrule

Name *: AllTrafficLBrule

IP Version *: IPv4

Frontend IP address *: 10.83.5.100 (LoadBalancerFrontEnd)

HA Ports:

Backend pool: ngfw (2 virtual machines)

Health probe: HTTPMgmt (TCP:443)

Session persistence: None

Idle timeout (minutes): 4

TCP reset: Disabled

Floating IP (direct server return): Disabled

Create implicit outbound rules: Yes

In a similar manner, we set up the ‘outside’ load balancer. The only major difference is that this is a public load balancer and hence is assigned a public IP. This public IP will be used to access our cloud application.

Resource group (change):	SecureRG	Backend pool:	ngw (2 virtual machines)
Location:	East US	Health probe:	HTTPSMgmt (Tcp:443)
Subscription (change):	Microsoft Azure Enterprise	Load balancing rule:	OutLBrule (Tcp/80)
Subscription ID:	[REDACTED]	NAT rules:	0 inbound
SKU:	Standard	Public IP address:	[REDACTED] (OutLBw-IP)
Tags (change):	Click here to add tags		

The backend pool will consist of the outside interfaces of the firewall.

Backend pool	Virtual machine	Virtual machine status	Network interface	Private IP address
ngw (2 virtual machines)	safengfw2	Running	safengfw2-nic3	10.83.6.11
	safengfw1	Running	safengfw1-nic3	10.83.6.10

The load balancer rule is set up to listen on TCP port 80 and forward the traffic to the backend pool consisting of outside interfaces of NGFW devices on the same TCP port 80. Make sure to set the session persistence to ‘Client IP and protocol’ to ensure that traffic for the same session is forwarded to the same NGFW.

Now that we have the inside and outside load balancers set up, we need to update the routing tables for the Web and App tier to force the outbound traffic for Internet to inside interfaces of NGFW devices. We add the default routes in the web and app routing table with next hop as 10.83.5.100 i.e.. the ‘inside’ load balancer, this ensures load balancing of outbound internet traffic to inside interfaces of the NGFW devices.

Step 4. Access the application – The last step to setting up this application is to register a domain for the application and add an ‘A’ record to point the domain name to the public IP address of the ‘outside’ load balancer. When we access the newly registered domain name for the Azure application, we are prompted for the initial set up and then we land on cloud application home page as below.

Enabling WAF and DDoS protection

At this point we have finished setting up a fully functional cloud application. We will now add the WAF and DDOS protection capabilities in our design. For the purpose of this document, we will demonstrate two different options for adding these capabilities, you can pick either of these two options:

OPTION 1: Azure WAF and DDoS protection

Azure WAF can be deployed with Azure Application Gateway, Azure Front Doors, and Azure Content Delivery Network (CDN) service from Microsoft. We incorporate Azure Front Doors service to validate WAF integration with our application. Based on different application set ups, Azure native WAF may or may not be an available option for WAF capabilities. Refer to the [Azure documentation](#) for more details about Azure WAF service.

Azure provides two tiers for DDoS protection, Basic and Standard. Azure Front Doors platform is by default protected by Azure Basic DDoS Protection. For further security, Standard DDoS protection is enabled at the VNET level to protect any resource with a public IP in outside and management tiers. While Basic service is enabled by default at the VNET level also, Standard tier needs to be enabled manually. Find more details in Azure documentation [here](#).

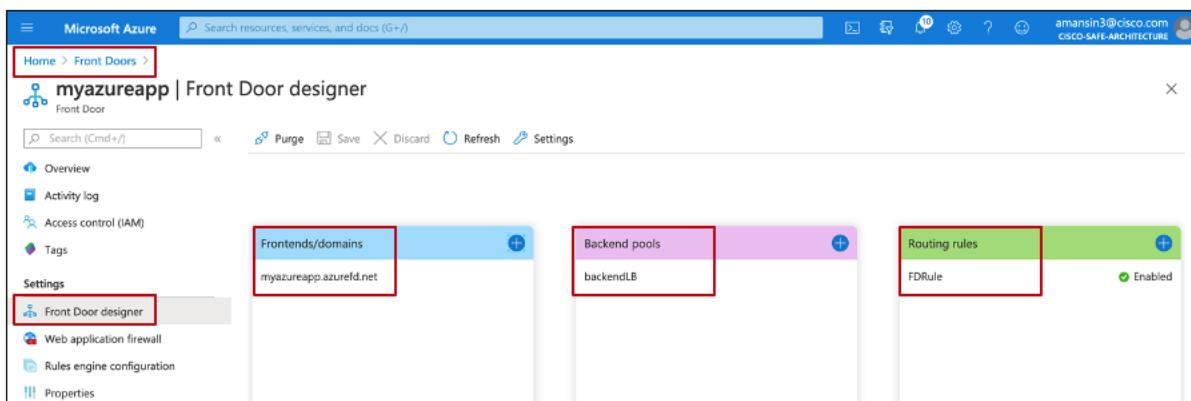
Implementation procedure:

Step 1. Setting up Azure WAF service

Step 2. Setting up Azure DDoS protection for VNET

Step 1. Setting up Azure WAF service - Go to 'Front Doors' service on the Azure portal and click on 'add' to create a new Front Door service for the newly set up cloud application. Configuring Front Door service involves three steps -you will need to specify a domain name for the application, the backend pool which consists of outside interfaces of the NGFWs and lastly the traffic routing rules.

Azure Front Door will receive the traffic from internet users and load balance it to the outside interfaces of NGFWs. Make sure you enable session affinity at 'Frontend/domain' level if your application is stateful. For detailed steps on how to create Azure Front Doors, check out the [Azure documentation](#).



Optionally, you can customize the domain name for your Front Door, refer to [Azure documentation](#).

To lock down the application to accept traffic only from this specific Front Door, you can use ACL on Firewall to lock down the access to Front Door backend IP space. Refer to the Q&A page [here](#) for more details on AFD IP space.

Now that we have our Front Door set up complete, we will enable the Azure WAF. Go to 'Web Application Firewall Policies' on the Azure portal and click on 'add' to enable the WAF service for Front Door that we created above.

AFDwafpolicy | Associated frontend hosts

Front Door WAF policy

You can configure a WAF policy and associate that policy to one or more Front Door frontend hosts for protection. [Learn more](#)

Frontdoor	Subscription
myazureapp	Microsoft Azure Enterprise

Step 2. Setting up Azure DDoS protection for VNET - For enabling the Standard tier DDoS capability, go to ‘Virtual Networks’ on the Azure portal and select the VNET created for the cloud application. On the left-hand side panel, select DDoS to enable the Standard tier. For detailed instructions on how to configure DDoS, refer to the Azure [documentation](#).

SecureVNet1 | DDoS protection

Virtual network

DDoS standard protection plan metrics can be found by selecting protected public IP addresses in the DDoS metrics blade.

DDoS protection *

Basic Standard

I know my resource ID

DDoS protection plan *

OPTION 2: Radware Cloud Service for WAF and DDOS protection.

Radware Cloud WAF and DDoS services are integrated by using CNAME DNS records. Its independent of any specific Azure Cloud service. For Radware integration, we use DNS to forward traffic to Radware cloud. After scanning the traffic, it is redirected to the Public IP of ‘outside’ load balancer.

Implementation procedure:

Step 1. Onboard the application to Radware Cloud

Step 2. Update the DNS setting to point traffic to Radware Cloud

Step 1. Onboard the application to Radware Cloud - The first step to integrating Radware cloud service in your environment is to add the application onto the Radware cloud. On the Radware cloud portal, go to ‘Assets > Application’ and click on the plus button on the upper right-hand side of the screen. Add the prompted details i.e. the application domain name (www.myazureapp.net), the origin server (in this case it would be the ‘outside’ load balancer’s

public IP address) and the protocol (HTTP). If your application is based on HTTPS protocol, you would need to add the certificate information as well.

As part of the onboarding process, each customer can choose between immediate and learning based protection. Immediate protection will enforce a predefined security policy, preventing known attacks. In order to cover both known and unknown attacks, Radware recommends using the learning-based protection method. During the first 2 weeks (duration can vary depending on traffic), Radware evaluates traffic patterns and can automatically update both negative and positive security models by refining signatures, creating exceptions and building the allowed file extension list per application, greatly reducing false positives.

New Application

Display Name ⓘ
MyAzureapp

Application Domain ⓘ
www.myazureapp.net

Origin Server ⓘ
IPv4 12.3.4

Region ⓘ
North America (Ashburn)

Application Protocol ⓘ Certificate ⓘ
 HTTP HTTPS You must set certificates. You can create one [here](#)

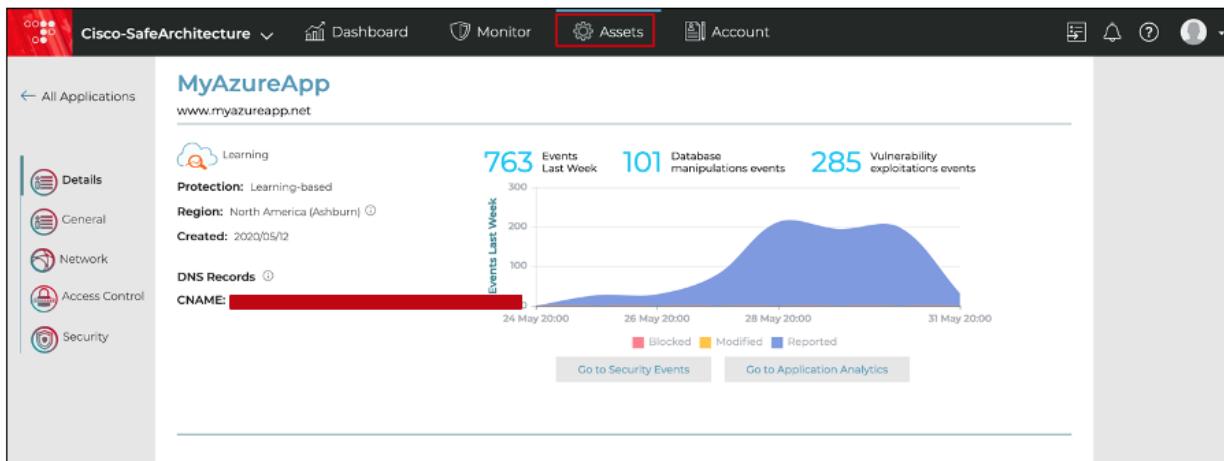
General Information

Cancel Save

Once the details are saved, the application can be seen as below.

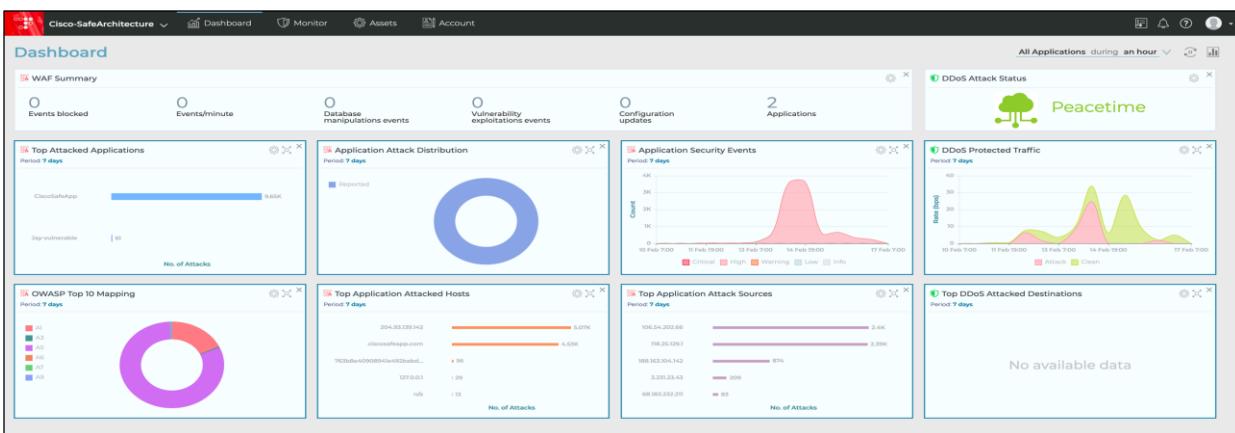
State	Name	Domain	Created	Region	Events (Last 7 Days)
Cloud	MyAzureApp	www.myazureapp.net	2020/05/12	North America (...	763

Click on the application and copy the allocated CNAME. We need to update the DNS records for our application with this Radware CNAME.



Step 2. Update the DNS setting to point traffic to Radware Cloud – Update the DNS record sets for the application domain with Radware CNAME (from the previous step) and corresponding IP address. After this change, it might take a few minutes for the DNS update to propagate. Once the DNS records are fully updated, the traffic will start getting redirected to Radware Cloud servers before it hits the ‘origin server’ in the Azure cloud. To eliminate direct origin attacks, Radware recommends configuring the perimeter firewall to only allow the Cloud WAF to access the application origin server directly. The service IP addresses can be requested from Radware Support Team. For more information, check out the [Radware Cloud WAF Quick Start guide \(login required\)](#).

Radware Cloud portal displays all the traffic statistics related to various onboarded applications. The dashboards are fully customizable based on your requirements.



Integrating with Cisco SecureX threat response

In this last implementation step, we will enable the AMP4E and Umbrella modules in the threat response portal to get a unified view into the Azure environment. We create API keys in AMP4E and Umbrella portals and then configure those keys in the threat response dashboard.

Implementation procedure:

Step 1. Add the AMP4E module

Step 2. Add the Umbrella modules

Step 1. Add the AMP4E module – Log on to the threat response dashboard and go to available modules, add a new AMP4E module. Threat response displays all the steps on how to integrate the AMP module once you click on ‘add module’ as shown in the snapshot below.

Add New AMP for Endpoints Module

Quick Start

When configuring AMP for Endpoints integration, you must first gather some information from your AMP for Endpoints instance and then add the AMP for Endpoints module.

- Log into the AMP for Endpoints console, click the **Accounts** menu and choose **API Credentials**.
- Click **New API Credential**, enter the **Application name** and **Scope** (action you want to allow) and then click **Create**.
- From the **API Key Details** page, copy both the **3rd Party API Client ID** and the **API Key** into a file, or leave the tab open. **Important:** Do not close the tab without retrieving these values; the API key is not retrievable once the tab is closed.
- Complete the **Add New AMP for Endpoints Module** form:
 - Module Name** – Leave the default name or enter a name that is meaningful to you. (You use the default name unless you already have a module named AMP for Endpoints or you need to specify one of several regional Cloud instances.)
 - URL** – From the drop-down list, choose the appropriate URL for the regional AMP Cloud account.
 - 3rd Party API Client ID** – Copy and paste the value obtained earlier.
 - API Key** – Copy and paste the value obtained earlier.
 - Act In the name of Active User** – Check the check box if you want to interact with AMP in the name of the user that is currently logged in.
- Click **Save** to complete the AMP for Endpoints module configuration.

Step 2. Add the Umbrella module - Go to available modules again, this time, add a new Umbrella module. In a similar manner as AMP4E, the integration steps are listed on the threat response page. Create the API keys in the Umbrella portal as instructed and then add the it in the threat response configuration page.

Add New Umbrella Module

Quick Start

When configuring Umbrella integration, you must first gather some information from your Umbrella instances and then complete the **Add New Umbrella Module** form.

Investigate API

- In Umbrella, navigate to **Investigate > Investigate API Access**, click **Create New Token** and enter a title for the token, and then click **Create New Token** again.
- Copy the **Access Token** value into the **API Token** field on the **Add New Umbrella Module** form.

Enforcement API

Note: To integrate the Umbrella Enforcement API, the user must be an admin in an Umbrella standalone org or child org instead of an admin of an Umbrella console.

- In Umbrella, navigate to **Policies > Policy Components > Integrations**, click **Add** and enter a name, and click **Create**.
- Under **What should this API do?**, click the **Umbrella Reporting** radio button and then click **Create**.
- Copy the following values into the **Reporting** fields on the **Add New Umbrella Module** form:
 - API Key** (Your Key)
 - API Secret** (Your Secret)
 - Organization ID** – from browser URL, the set of numbers between /#/ and /#/
 - Request Timeframe (days)** – Enter the timeframe (in days) for enriching sightings from the most recent DNS requests

When you have completed adding the API information, click **Save**.

After we have saved the module configurations, the modules will be listed under ‘Module’ as below.

Validation Testing

Tetration

Validation procedure overview:

- Test Case 1 - Creating the workspace for Azure cloud application
- Test Case 2 - Using ADM to discover the policies for Azure workloads and setting up an app view
- Test Case 3 - Enforcing the policies on workloads
- Test Case 4 - Discovering the vulnerable packages on the Azure workloads

Test case 1: Creating an application workspace for Azure cloud application

This test case involves defining annotations for the Azure environment. These annotated attributes are used later to segregate the tiers and segments within the Azure VNET and hence define a workspace for our tiered cloud application.

Validation procedure:

Step 1. Build an inventory

Step 2. Define scopes

Step 3. Create a workspace

Step 1. Build an inventory – Define the attributes that would help you segregate your tiered application workloads in the cloud and hence construct policies for them. We will upload a CSV file with workload annotations for this purpose.

1.1: Based on the architecture of our tiered application (elaborated in the previous sections of this document), the following annotations were used (Table: Azure Cloud Inventory). Save this in a CSV file.

Table 1. Azure Cloud Inventory

IP	Application	Region	Tier	Type
10.83.1.0/24	Safe3tierApp	US-East	WebServer	Azure-Cloud
10.83.2.0/24	Safe3tierApp	US-East	AppServer	Azure-Cloud
10.83.3.0/24	Safe3tierApp	US-East	Database	Azure-Cloud
10.83.4.0/24	Safe3tierApp	US-East	Management	Azure-Cloud
10.83.5.0/24	Safe3tierApp	US-East	Inside	Azure-Cloud
10.83.6.0/24	Safe3tierApp	US-East	Outside	Azure-Cloud
10.83.7.0/24	Safe3tierApp	US-East	Diagnostic	Azure-Cloud

1.2: Now, log into the Tetration cloud portal and go to ‘Visibility > Inventory Upload’. Click on ‘Select File’ and ‘add’ the CSV file.

The screenshot shows the 'Inventory Upload' interface. On the left, there's a sidebar with icons for 'List', 'Annotations', 'Upload', 'Logs', and 'Metrics'. The main area has a heading 'Managing inventory annotations on the [redacted] scope.' Below it, a note says 'Tetration will generate alerts on any inventory matching an annotation prefix with 'lookout_...'.' Under the 'Upload' section, there's a 'Select File' button (highlighted with a red box), a 'Download Annotations' button, and a note about selecting a CSV file. To the right, there's a list of annotation columns with checkboxes: Application, Location, Region, Tier, and Type. At the bottom right are 'Save' and 'Reset' buttons.

1.3: After a few minutes, you can go to ‘Visibility > Inventory Search’ and test the filters generated, based on annotations from the Step 1.2.

The screenshot shows the 'Inventory Search' interface. The top bar includes the Cisco Tetration logo, 'INVENTORY SEARCH' button, and monitoring status. A filter bar at the top has a red box around the query '*Type contains Azure and *Tier contains web'. Below it, a message says 'Showing 2 of 2 matching results'. The results table has columns: Hostname, VRF, Address, and OS. Two rows are shown: 'webscales000000' (VRF INTGSSOPOV, Address 10.83.1.4, OS CentOS) and 'webscales000001' (VRF INTGSSOPOV, Address 10.83.1.5, OS CentOS). A note at the bottom says 'Results restricted to root scope INTGSSOPOV'. A 'Create Filter' button is also visible.

Step 2. Define scopes – We will define a scope to group together all the workloads in our tiered application in the Azure cloud. We will make use of the annotations/filters that we constructed in Step 1. We created the scope ‘Azure-US-EAST’, which includes all the workloads from our tiered app in ‘US-East’ region in Azure Cloud.

Click on the settings icon in the top right corner of the portal and then go to ‘Scopes’ option. Click on ‘Create New Scope’ and fill in the name of the Scope and a query as below.



Step 3. Create a workspace - Application workspaces are the containers for defining, analyzing and enforcing policies for a particular application. We will create a workspace for our tiered Azure cloud application in this step.

Click on ‘Segmentation’ and then click on ‘Create New Workspace’. Give the workspace a name and select the Scope that we created in Step 2.

Workspace	Enforced Applications	Enforcement Agents	Desired Agent Policies
Azure-Safe3tierApp	0	12	0 / 12
WordPress3TierApp			
AWS-Safe3tierApp			

At this point, we have successfully built the inventory, created a scope and defined a workspace for our tiered cloud application.

Test case 2: Using ADM to discover the policies for Azure workloads and setting up an app view

This test case validates the use of ‘ADM’ to automatically discover the policies based on flow and other data received from workloads. We will refine the discovered workload clusters and update the inventory filters to eventually come up with a set of policies that can be enforced on our cloud workloads.

Validation procedure:

Step 1. Discover policies using ADM

Step 2. Refine inventory filters, clusters and policies

Step 3. Create the App View

Step 1. Discover policies using ADM - Before running the ADM, ensure that all types of traffic flows are generated in the application environment. This would provide ADM the required data to generate an accurate policy set and hence ensure that we don’t miss any critical but less common traffic flows.

Go to the newly created workspace and click on ‘Start ADM Run’ on the top right corner, select a suitable time range to ensure that you cover all the traffic flows.

Step 2. Refine inventory filters, clusters and policies – Post the ADM run, policies and clusters would be generated. At this point, we manually update and customize all the cluster queries and approve them.

2.1: Go to ‘Clusters’ tab, click on any of the clusters, the panel on the right-hand side will show the cluster details like name, description, cluster query, workloads, neighbors. Update name and description to make it more intuitive, update the cluster query if need be. For example, we updated the cluster query for auto scaled workloads. We used previously defined annotation to dynamically identify the workloads in Virtual Machine Scale Sets like the web and application servers.

2.2: Click on ‘Policies’ tab, review the policies keeping the workload flows in mind. We considered the following flows for policies:

- User requests incoming to web workloads via firewall’s inside interface
- Traffic between the workloads
 - Web servers to app load balancer
 - App load balancer to app servers
 - App servers to database
- Management tier to all the workloads

- Outbound internet access from all the workloads for updates/patches, DNS, DHCP, NTP

Priority	Action	Consumer	Provider	Services
100	ALLOW	AppScaleSet	INTGSSOPOV	TCP : 80 (HTTP) ...3 more
100	ALLOW	WebScaleSet	INTGSSOPOV	TCP : 80 (HTTP) ...3 more
100	ALLOW	AppScaleSet	UmbrellaVAs	UDP : 53 (DNS)
100	ALLOW	WebScaleSet	UmbrellaVAs	UDP : 53 (DNS)
100	ALLOW	WebScaleSet	ApplB	TCP : 80 (HTTP)
100	ALLOW	INTGSSOPOV	AppScaleSet	TCP : 80 (HTTP)
100	ALLOW	JumpServer	AppScaleSet	TCP : 22 (SSH)
100	ALLOW	WebScaleSet	AppScaleSet	ICMP ...1 more
100	ALLOW	INTGSSOPOV	WebScaleSet	TCP : 80 (HTTP)
100	ALLOW	JumpServer	WebScaleSet	TCP : 22 (SSH)
100	ALLOW	InsideTier	WebScaleSet	TCP : 80 (HTTP)
100	ALLOW	AppScaleSet	AzureDB	TCP : 3306 (MySQL)

Step 3. Create the App View – Go to ‘App view’ tab and click on ‘Create New App View’. Pin the workloads on the right-hand side panel to include them in your diagram. Double click on each pinned cluster on the view to automatically draw the traffic flows.

```

graph TD
    InsideTier((InsideTier)) --> WebScaleSet((WebScaleSet))
    WebScaleSet --> ApplB((ApplB))
    ApplB --> INTGSSOPOV((INTGSSOPOV))
    ApplB --> AppScaleSet((AppScaleSet))
    ApplB --> UmbrellaVAs((UmbrellaVAs))
    INTGSSOPOV --> AppScaleSet
    AppScaleSet --> JumpServer((JumpServer))
    JumpServer --> UmbrellaVAs
    UmbrellaVAs --> AzureDB((AzureDB))
  
```

Test case 3: Enforcing the policies on workloads.

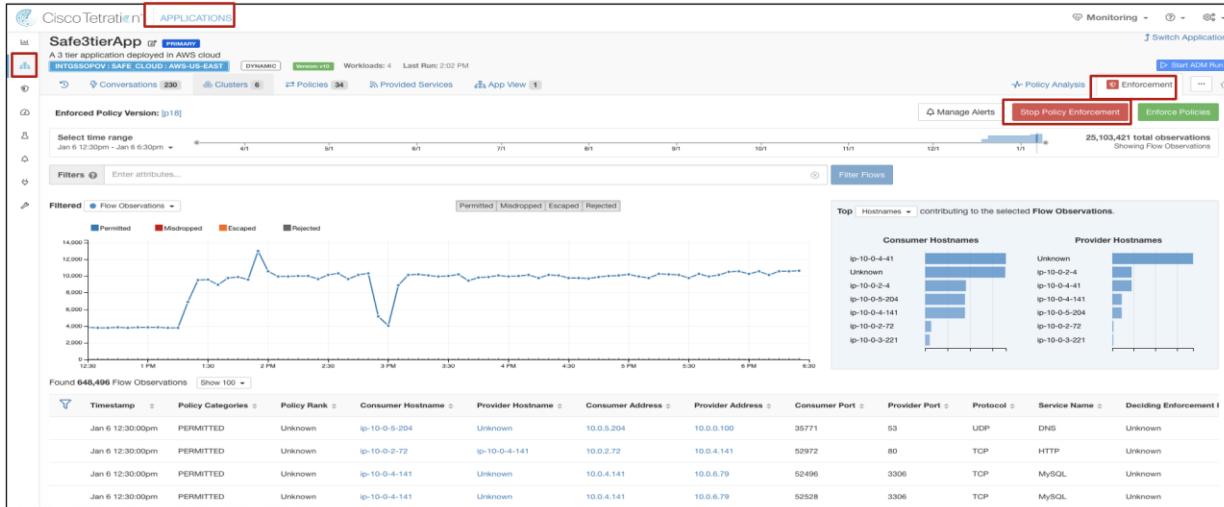
This test case focuses on enforcing the policy set that we formulated in Test Case 2. We will publish the policies and verify if those are enforced as expected.

Validation procedure:

Step 1. Publish the policies

Step 2. Verify policy enforcement on workloads

Step 1. Publish the policies – Select the ‘Enforcement’ tab on the Tetration portal within the application workspace and click on ‘Enforce Policies’.



Step 2. Verify policy enforcement on workloads – Since we had CentOS based workloads, we monitored the '/usr/local/tet/log/tet-enforcer.log' to see if policies are successfully enforced. A simple ping or telnet test can also be used to verify the lockdown of ports and protocols.

```
AWS:
I0107 04:25:25.275367 468 agent_controller.cpp:426] IPC is ready, start writing the message to IPC
I0107 04:25:25.275394 468 agent_controller.cpp:445] Done writing to shared memory /tmp/tet_enforcer_ipc
I0107 04:25:25.275499 471 agent_enforcer.cpp:813] Message from AgentController available, start processing
I0107 04:26:25.216424 468 ssl_client.cpp:410] Received message body length: 12
Linux Academy:
Administrator
I0107 04:26:30.277519 468 ssl_client.cpp:510] Calling callback fn to process msg w/KoFTopZD
I0107 04:26:30.277545 468 agent_controller.cpp:179] Start message processing from EFE
I0107 04:26:30.277563 468 agent_controller.cpp:222] Write the protobuf to AgentEnforcer
I0107 04:26:30.277696 472 agent_enforcer.cpp:966] Process EFE Message
I0107 04:27:05.342911 473 agent_enforcer.cpp:1283] Received Policy config version: 1545811665 #KoFTopZD#
I0107 04:27:05.344177 473 agent_enforcer.cpp:1392] Processing network policy config from EFE, version: 1545811665
I0107 04:27:05.344187 473 agent_enforcer.cpp:1396] Storing the policy and enforcing
I0107 04:27:05.344703 473 firewall_context.cpp:140] Policy has been validated, applying the policy
I0107 04:27:05.344712 473 firewall_context.cpp:169] Applying all firewall rules to the system firewall
I0107 04:27:05.714491 473 iptables_context.cpp:529] Staged rules have been committed
I0107 04:27:05.726775 473 agent_enforcer.cpp:1403] Policy config has been applied successfully, current version: 1545811665, highest version: 1545811665
Wordpress:
```

Use the CLI command ‘ipset list’ to view the ipset firewall settings enforced by Tetration agent on the CentOS workloads.

```
Name: ta_520a2f879b4e9ab37a6e620e928b
Type: hash:net
Revision: 6
Header: family inet hashsize 1024 maxelem 65536
Size in memory: 504
References: 2
Number of entries: 2
Members:
10.0.3.196
10.0.2.57

Name: ta_58225b8365be52f7b49f8254d96e
Type: hash:net
Revision: 6
Header: family inet hashsize 1024 maxelem 65536
Size in memory: 440
References: 30
Number of entries: 1
Members:
10.0.2.16

Name: ta_9c21cc33b4bedbed655487444413
Type: hash:net
Revision: 6
Header: family inet hashsize 1024 maxelem 65536
Size in memory: 504
References: 6
Number of entries: 2
Members:
10.0.5.129
10.0.4.89

Name: ta_bdf04e36735d84cc7cc983454e94
Type: hash:net
Revision: 6
Header: family inet hashsize 1024 maxelem 65536
Size in memory: 504
References: 8
Number of entries: 2
Members:
10.0.9.253
10.0.8.179
```

Test case 4: Discovering the vulnerable packages on the Azure workloads.

This test case looks for vulnerable packages/software installed on various workloads in our Azure cloud. We identify a vulnerable package/software on our workloads, patch those and then rerun the report.

Validation procedure:

Step 1. Check the vulnerability report

Step 2. Fix a vulnerability and rerun the report

Step 1. Check the vulnerability report – Go to ‘Security > Vulnerabilities’, click on ‘Packages’ tab to see all the vulnerable packages installed on various workloads in our three-tier application. For the sake of this test, let’s consider ‘telnet-0.17-64.el7’ as shown below.

Package Details - telnet-0.17-64.el7

Package: telnet-0.17-64.el7

CVEs: CVE-2020-10188 (v2: 10, v3: N/A)

Affected workloads: appscale00000P

Package	Worst Score (V2)	Worst Score (V3)
bpftrace-3.10.0-1062.9.1.el7	10	N/A
kernel-3.10.0-1062.9.1.el7	10	N/A
kernel-3.10.0-693.el7	10	N/A
telnet-0.17-64.el7	10	N/A
bpftrace-3.10.0-1062.12.1.el7	8.3	N/A

WORKLOAD PROFILE

APPSCALES00000P

CVE	Package Name	Version	Score (V2)	Score (V3)	Severity (V2)	Base Severity (V3)	Access Vector (V2)	Access Complexity (V2)	Authentication (V2)	Confidence
CVE-2020-10188	telnet	0.17-64.el7	10	HIGH	NETWORK	LOW	NONE	COMPLETE		

We see that the workload ‘appscale00000P’ is affected by this CVE. Logon to this workload and verify the telnet package.

```
[centos@appscales00000P ~]$ rpm -q telnet
telnet-0.17-64.el7.x86_64
[centos@appscales00000P ~]$
```

Step 2. Fix the vulnerability and rerun the report – We uninstall the telnet package from this workload.

```
[centos@appscale00000P ~]$ 
[centos@appscale00000P ~]$ sudo yum remove telnet
Loading mirror http://mirror.tengxun.com/epel
Resolving Dependencies
--> Running transaction check
--> Package telnet.x86_64 1:0.17-64.el7 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

Transaction Summary
  Remove 1 Package

  Removing: telnet.x86_64 1:0.17-64.el7

  Transaction Summary
    Remove 1 Package

  Installed size: 113 k
  1 M (0% of 1.7 M)
  Downloading packages:
  Running transaction check
  Running transaction test
  Transaction test succeeded
  Running transaction
    Erasing : 1:telnet-0.17-64.el7.x86_64
    Verifying : 1:telnet-0.17-64.el7.x86_64

  Removed:
    telnet.x86_64 1:0.17-64.el7

  Complete!
[centos@appscale00000P ~]$
```

Wait for a few minutes after the uninstall, go back to Tetration portal and check the vulnerability report again. We can see that none of the CVEs related to Telnet show up anymore.

The screenshot shows the Cisco Tetration Workload Profile interface. At the top, there's a navigation bar with tabs like Summary, Long Lived Processes, Process Snapshot, Interfaces, Packages, Vulnerabilities, Config, Stats, Policies, Container Policies, Network Anomalies, File Hashes, and Visit History. Below the navigation bar, there's a search bar with a filter applied: "Filter" and "Package Name contains telnet". A red box highlights this filter. The main area displays a table with columns for ID, Name, Status, and Actions. The status column for all rows shows "Running". A red box highlights the "Status" column header. At the bottom right, there's a Cisco logo and some legal text: "TetrationOS Software, Version 3.3.2-PATCH-3.3.2.28 (Test) Privacy and Terms of Use TAC Support: http://www.cisco.com/tac © 2019-2020 Cisco Systems, Inc. All rights reserved."

Advanced Malware Protection for Endpoints

Test Case: Quarantine a suspicious file

This test case involves the detection of using AMP for endpoint ‘simple custom detections’ to quarantine a suspicious PDF file.

Validation procedure:

Step 1. Setting up AMP4E policy to quarantine a suspicious file

Step 2. Verifying the deletion of a suspicious file

Step 1. Setting up AMP4E policy to quarantine a suspicious file – For the validation purpose, we consider a 1 MB PDF file that we will block list using AMP ‘Simple Custom Detections’. We will then try to download the same PDF file on a cloud workload and assert that our policy works as expected.

As per our initial AMP4E set up, we had configured the group ‘Secure Cloud’ (Management > Groups) for our workloads in the Azure cloud.

The screenshot shows the Cisco AMP for Endpoints Management interface. The top navigation bar includes links for Dashboard, Analysis, Outbreak Control, Management (which is selected), and Accounts. A search bar and user profile for Amandeep Singh are also present. The main content area is titled "Edit Group: Secure Cloud". On the left, there are several configuration fields:

- Name: Secure Cloud
- Description: Cloud workloads
- Parent Group: (dropdown menu)
- Windows Policy: Default Policy (Protect Policy)
- Android Policy: Default FireAMP Android
- Mac Policy: Protect Policy for FireAMP Mac
- Linux Policy:** CloudApp-LinuxPolicy (highlighted with a red box)
- iOS Policy: Protect

Below these fields are two buttons: "Cancel" and "Save".

On the right side, there is a sidebar titled "Computers" which lists 8 direct members:

- appscalces00000P
- appscalces00000Q
- ip-10-0-2-34.safeapp.lab
- ip-10-0-3-18.mysafeapp.net
- ip-10-0-4-199.safeapp.lab
- ip-10-0-5-169.mysafeapp.net
- webscales000000
- webscales000001

Below the sidebar, a note says "No child members" and "Assign computers to groups on the [Computers](#) page".

At the bottom of the main content area, there are sections for "Child Groups" and "Add Child Groups". The "Child Groups" section has a "Select All" and "Deselect All" button. The "Add Child Groups" section lists several group names:

- DMZ Shared Services
- Domain Controller
- Industrial Workstations
- Orbital Group
- Protect
- Secure Campus
- Secure DC
- Server
- Triage

Buttons for "Remove Selected" and "Add Selected" are located at the bottom of these sections.

Note: During our implementation phase we had used the AMP4E agent tied to this specific group ‘Secure Cloud’, which we had created as part of the initial AMP4E set up (not elaborated in this guide, follow AMP4E documentation for detailed steps on setting up AMP4E policies). All the workloads in Azure VNET register with AMP Cloud under this specific group.

It can be seen in the snapshot above that we tied the specific group to Linux policy ‘CloudApp-LinuxPolicy’. Go to ‘Management > Policies’ and select the specific Linux policy.

The screenshot shows the AMP4E policy configuration interface for a policy named 'CloudApp-LinuxPolicy'. The policy is described as 'Policy for linux workloads in Cloud'. The interface includes sections for 'Modes and Engines' (Files, Network, ClamAV), 'Exclusions' (Quarantine, Audit, On), 'Proxy' (Not Configured), and 'Groups' (Secure Cloud, 8 workloads). The 'Outbreak Control' section shows 'Custom Detections - Simple' (CloudApp-CSD) and 'Custom Detections - Advanced' (Not Configured). The bottom of the screen displays status information: View Changes, Modified 2020-05-27 17:12:00 UTC, Serial Number 237, and action buttons: Download XML, Duplicate, Edit, and Delete.

CloudApp-LinuxPolicy Policy for linux workloads in Cloud				Groups
Modes and Engines		Exclusions	Proxy	Groups
Files	Quarantine	Not Configured	Not Configured	Secure Cloud 8
Outbreak Control				
Custom Detections - Simple	Custom Detections - Advanced	Application Control	Network	
CloudApp-CSD	Not Configured	Not Configured	Not Configured	

View Changes Modified 2020-05-27 17:12:00 UTC Serial Number 237
 Download XML
 Duplicate
 Edit
 Delete

Note: We had preconfigured the Linux policy associated with AMP4E group ‘Secure Cloud’. We also tied a new Simple Custom Detection ‘CloudApp-CSD’ to the Linux policy. If there was no initial config on AMP console, then you would see default policies here.

As we see in the snapshot, the Linux policy above is tied to Simple Custom Detections ‘CloudApp-CSD’ (Outbreak Control > Simple).

Go to ‘Outbreak Control > Simple Custom Detections’ and click on edit ‘CloudApp-CSD’ to upload the PDF file that we want to block list in the Azure cloud environment. Uploading the PDF file will add the SHA value to the SCD policy and quarantines the file associated with it from all the cloud workloads registered under the specific group.

The screenshot shows the Cisco AMP for Endpoints Cloud portal interface. At the top, there's a navigation bar with 'Dashboard', 'Analysis' (with a dropdown for 'Outbreak Control'), 'Management', 'Accounts', and a search bar. A user 'Amandeep Singh' is logged in. Below the navigation, a section titled 'Custom Detections - Simple' lists three items:

- IoT Demo**: 2 files, Created by Andrew Mcphee on 2020-05-21 04:19:30 UTC. Used in policies: Industrial Workstation Policy. Used in groups: Industrial Workstations.
- CloudApp-CSV**: 8 files, Created by Amandeep Singh on 2020-01-10 17:09:46 UTC. Used in policies: CloudApp-LinuxPolicy. Used in groups: Secure Cloud.
- Quick SCD**: 0 files, Created by Bart McGlothin on 2016-05-27 03:37:38 UTC. Used in policies: Audit Policy, Audit Policy for FireAMP Linux, Audit Policy for FireAMP Mac, Domain Controller Policy, Orbital Policy, Protect Policy, Protect Policy for FireAMP Linux, Protect Policy for FireAMP Mac, Server Policy, Triage Policy, Triage Policy for FireAMP Mac. Used in groups: Audit, Domain Controller, Orbital Group, Protect, Secure Cloud, Secure DC, Server, Triage.

On the right side, there's a form for adding SHA-256 values. It includes fields for 'Add SHA-256' (with a red box around it), 'Upload File' (with a red box around it), and 'Upload Set of SHA-256s'. Below these are fields for 'File' (set to 'file-example_PDF_1MB.pdf') and 'Browse'. There's also a 'Note' field and a 'Upload' button. A section titled 'Files included' shows several file entries, with one entry ('5e4d40fc...bb4be37a') highlighted with a red box. This entry is described as a 'Sample PDF' created by uploading 'file-example_PDF_1MB.pdf' via Web from [REDACTED]. It was added by Amandeep Singh on 2020-05-27 17:16:15 UTC. A 'Delete...' button is shown next to this entry.

Step 2. Verify the deletion of the suspicious program – Log on to a cloud workload, we picked one of the web servers in web VSS. We downloaded the PDF file that we block listed above. We can see that the file is immediately quarantined by the AMP agent on the workload.

```
[centos@webscales00000 ~]$ wget https://file-examples.com/wp-content/uploads/2017/10/file-example_PDF_1MB.pdf
--2020-05-27 19:46:41-- https://file-examples.com/wp-content/uploads/2017/10/file-example_PDF_1MB.pdf
Resolving file-examples.com (file-examples.com)... 185.135.88.81
Connecting to file-examples.com (file-examples.com)|185.135.88.81|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1042157 (1.01MB) [application/pdf]
Saving to: 'file-example_PDF_1MB.pdf'

100%[=====] 1,042,157 1.43MB/s in 0.7s

2020-05-27 19:46:42 (1.43 MB/s) - 'file-example_PDF_1MB.pdf' saved [1042157/1042157]

[centos@webscales00000 ~]$ ls -lh
total 16K
-rwxr-x--x. 1 root root 38 May 27 16:59 mysample.sh
-rw-r--r--. 1 root root 38 May 27 16:59 mysample.txt
drwxr-xr-x. 2 root root 22 May 27 17:43 testdir
-rw-r--r--. 1 root root 18 May 27 17:48 text.txt
-rw-rw-r--. 1 centos centos 17 May 27 17:52 text.txt.i
[centos@webscales00000 ~]$
```

We also confirm the quarantine event from the event logs on the AMP Cloud portal. Log on to the AMP Cloud portal and go to 'Analysis > Event', we see a 'Quarantine successful' event post our steps above.

▼ webscales000000 detected file-example_PDF_1MB.pdf as Simple_Custom_Detection			Medium				Quarantine: Successful	2020-05-27 17:17:19 UTC
File Detection	Detection	Simple_Custom_Detection						
Connector Info	Fingerprint (SHA-256)	Se4d40fc...bb4be37a						
Comments	File Name	file-example_PDF_1MB.pdf						
	File Path	/home/centos/file-example_PDF_1MB.pdf						
	File Size	1017.73 KB						
	Parent Fingerprint (SHA-256)	782bed6a...5f896bd2						
	Parent Filename	wget						
Report 25 1 Restore File All Computers			Add to Allowed Applications File Trajectory					

Stealthwatch Cloud

Test Case: Monitor suspicious activity

This test case involves using the Stealthwatch cloud to monitor the activity within the Azure cloud environment.

Validation procedure:

Step 1. Monitor suspicious activity in Stealthwatch Cloud

Step 1. Monitor suspicious activity in Stealthwatch Cloud - Login to the Stealthwatch cloud portal.

Go to ‘alerts’, we see the alert ‘Excessive Access Attempts’ as shown below. This alert indicated that there were numerous attempts to get SSH access from an unexpected geo location, which is a suspicious behavior.

Alerts			
Search...		Status	Tags
9 open alerts sorted by newest			
!	Excessive Access Attempts (External) i-032dc6c1e859be077 #299	2 hours ago	44
!	Excessive Access Attempts (External) i-031bb97fc8aa5a9b1 #298	2 hours ago	49
!	Excessive Access Attempts (External) ScaleWebServers i-0fa81682fd2ca2dfb, i-01b15f0e2c9d254f9 #364	6 hours ago	34
!	Excessive Access Attempts (External) i-09e0d2badc2cf3a1c #496	11 hours ago	26
!	Inbound Port Scanner Network #331	1 day, 2 hours ago	23
!	Excessive Access Attempts (External) i-0b071afe7f70b7134 #397	2 days, 5 hours ago	20
!	Permissive AWS Security Group Created (Amazon Web Services) 904585389016ianswarm #727	1 week ago	2
!	Geographically Unusual Remote Access ScaleWebServers i-0fa81682fd2ca2dfb, i-01b15f0e2c9d254f9 #695	1 week, 4 days ago	
!	Geographically Unusual Remote Access i-031bb97fc8aa5a9b1 #530	2 weeks, 4 days ago	

Excessive Access Attempts (External)

Status Open

ID 364

Description Device has many failed access attempts from an external device. For example, a remote device trying repeatedly to access an internal server using SSH or Telnet would trigger this alert. The alert uses the Multiple Access Failures observation and may indicate the device is compromised.

Updated May 27, 2020 12:00:00 PM

Created Apr 29, 2020 8:00:00 AM

IPs at the time of alert: 10.0.3.18, 10.0.2.34, 18.234.175.79

Hostname at the time of alert: i-0fa81682fd2ca2dfb, i-01b15f0e2c9d254f9

Assignee Nobody

Tags

After reviewing an alert, closing it will let the rest of your team know it's been resolved. In addition, closing alerts sends important feedback.

Close Alert

Supporting Observations

Multiple Access Failures Observation

Device had multiple failed application (e.g., FTP, SSH, RDP) access attempts.

Time	Device	Port	Profile	Connected Device	Failed Attempts
5/27/20 12:00 PM	ScaleWebServers	22 (ssh)	SSHServer	218.59.234.3	93
5/26/20 10:00 PM	ScaleWebServers	22 (ssh)	SSHServer	37.49.226.64	73
5/26/20 3:00 PM	ScaleWebServers	22 (ssh)	SSHServer	37.49.226.157	64
5/26/20 12:00 AM	ScaleWebServers	22 (ssh)	SSHServer	51.159.0.77	105

Cisco Umbrella

Test Case: DNS security

This test case involves adding DNS layer security to the Azure workloads. We created a DNS policy for our three-tier application workloads to block malicious domains. To verify the blocks, we accessed a test domain 'examplemalwaredomain.com' and then confirmed the same from Umbrella reporting.

Validation procedure:

Step 1. Set up DNS policy for Azure workloads

Step 2. Confirm if malware domain is blocked

Step 1. Set up DNS policy for Azure workloads – Go to ‘Policies > Management > DNS Policies’, add a new policy and make sure ‘Malware’ is set to block under security settings. Save the change.

This screenshot shows the Cisco Umbrella interface for managing DNS Policies. The left sidebar navigation includes 'Overview', 'Deployments', 'Policies', 'Management' (selected), 'DNS Policies' (highlighted with a red box), 'Firewall Policy', 'Web Policies', 'Policy Components' (Destination Lists, Content Categories, Application Settings, Security Settings, Block Page Appearance, Integrations, Selective Decryption Lists), 'Reporting', 'Admin', and 'Investigate'. The user profile is Amandeep Singh, CVD Team - Security Architecture.

The main content area is titled 'DNS Policies' and displays a summary of the policy 'AzureCloud_DNSPolicy'. Key details shown include:

- Policy Name:** AzureCloud_DNSPolicy
- 1 Identity Affected**
- 2 Destination Lists Enforced**: 1 Block List, 1 Allow List
- File Analysis Not Enabled**: Requires Intelligent Proxy, File Inspection Not Enabled
- Umbrella Default Block Page Applied**
- Security Setting Applied: TestCategory**: Command and Control, Malware, Phishing Attacks, plus 3 more will be blocked. No integration is enabled.
- Content Setting Applied: High**: Blocks adult-related sites, illegal activity, social networking sites, video sharing sites, and general time-wasters.
- Application Setting Applied: Default Settings**: DeviantArt, Facebook, LinkedIn, plus 31 more will be blocked.

Buttons at the bottom right include CANCEL, PREVIOUS, and SAVE.

Step 2. Confirm if malware domain is blocked – Run ‘nslookup’ on a test malware domain as shown in snapshot below. Utility returns Umbrella block page IP address as below.

```
[centos@webscales00000 ~]$ nslookup www.examplemalwaredomain.com
[centos@webscales00000 ~]$ nslookup www.examplemalwaredomain.com
Server: 10.83.4.100
Address: 10.83.4.100#53

Non-authoritative answer:
Name: www.examplemalwaredomain.com
Address: 146.112.61.107
Name: www.examplemalwaredomain.com
Address: ::ffff:146.112.61.107

[centos@webscales00000 ~]$
```

To further confirm the block action, select ‘Reporting > Activity Search’ and filter the accessed malware domain. Events show the action as ‘Blocked’.

This screenshot shows the Cisco Umbrella Reporting / Core Reports interface for Activity Search. The left sidebar navigation includes 'Overview', 'Deployments', 'Policies', 'Reporting' (selected), 'Core Reports' (Security Overview, Security Activity, Activity Search (highlighted with a red box)), 'App Discovery', and 'Additional Reports' (Total Requests, Activity Volume).

The main content area is titled 'Activity Search' and shows search results for 'examplemalwaredomain.com'. The search bar has '1 DOMAIN' selected and 'examplemalwaredomain.com' entered. The results table includes columns for Destination, Identity Used by Policy, Internal IP, External IP, Action, and Categories.

Destination	Identity Used by Policy	Internal IP	External IP	Action	Categories
myazureapp.net	www.examplemalwaredomain.com	10.83.1.4	40.121.19.47	Blocked	Malware
myazureapp.net	www.examplemalwaredomain.com	10.83.1.4	40.121.19.47	Blocked	Malware

Filter options on the left include Response (Allowed, Blocked, Proxied) and Protocol (HTTP, HTTPS). Buttons at the top right include SCHEDULE, DOWNLOAD, and LAST 24 HOURS.

Cisco Firepower NGFWv and CDO

Test Case: Enforce an access policy using CDO

In this test case, we will try to lock down the outbound access for our cloud workloads for specific TCP ports. We will use CDO to manage the FTDs we deployed in our Azure cloud. We already onboarded the FTDs to the CDO in the implementation section of this document.

Validation procedure:

Step 1. Configure and enforce the access policy

Step 2. Verify the access block

Step 1. Configure and enforce the access policy - We log on to a web server workload and try to access a non-standard TCP port on a server on the Internet. We can see in the snapshot below that the web server workload is able to connect at this point.

```
[centos@webscales00000 ~]$ telnet portquiz.net 666
Trying 52.47.285.216...
Connected to portquiz.net.
Escape character is '^]'.
^AC
Connection closed by foreign host.
[centos@webscales00000 ~]$
```

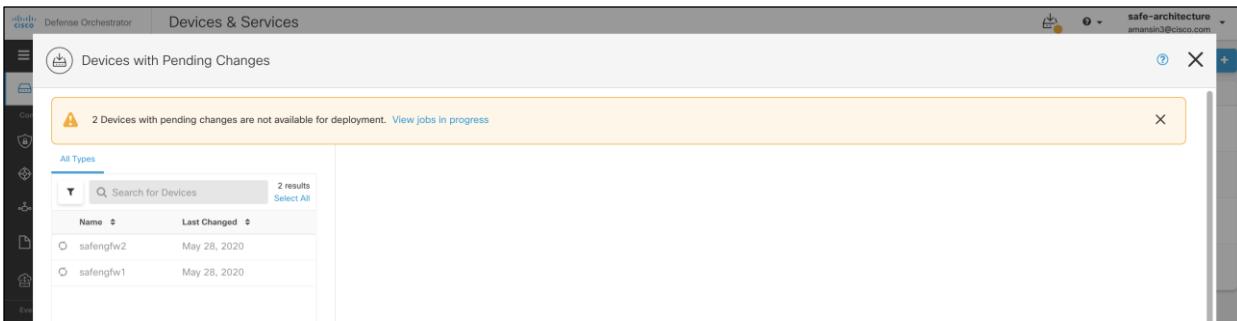
We want to block outbound access to such random TCP ports from our cloud workloads. Log on to the CDO portal and go to ‘Policies > FTD policies’ and select the policies corresponding to our Azure FTDs. We can see that the ‘inside_outside’ policy allows the cloud workloads to access any destination on any port on the Internet.

#	Name	Action	Source	Destination	Layer 7
1	AllowHTTP	Allow	[ZONES] outside_zone [NETS] any-ipv4	[ZONES] inside_zone [NETS] WebLB [PORTS] HTTP	Any
2	inside_outside	Trust	[ZONES] inside_zone [NETS] any-ipv4	[ZONES] any-ipv4 [NETS] any-ipv4	Any

We update the policy to lock it down to just the HTTP and HTTPS ports going out to the internet.

#	Name	Action	Source	Destination	Layer 7
1	AllowHTTP	Allow	[ZONES] outside_zone [NETS] any-ipv4	[ZONES] inside_zone [NETS] WebLB [PORTS] HTTP	Any
2	inside_outside	Trust	[ZONES] inside_zone [NETS] any-ipv4	[ZONES] outside_zone [NETS] any-ipv4 [PORTS] HTTP, HTTPS	Any

After making the policy change, click on the notification on the top right-hand side of the portal to push the changes to the specific FTD devices.



Step 2. Verify the access block - Now that we have updated the policy, we will try and attempt to verify the access. We SSH to a web server again and try to access websites on a random TCP port 666. We can see the connection timing out or getting blocked now. We can also see that an outbound access to a server on the internet on standard HTTP and HTTPS is still allowed.

```
[centos#webscales000000 -$]
[centos#webscales000000 -$] telnet portquiz.net 666
Trying 52.47.209.216...
telnet: connect to address 52.47.209.216 Connection timed out
[centos#webscales000000 -$]
[centos#webscales000000 -$]
```

```
[centos#webscales000000 -$] telnet www.cisco.com 443
Trying 23.196.192.158...
Connected to www.cisco.com.
Escape character is '^A'.
<Connection closed by Foreign host.
[centos#webscales000000 -$]
[centos#webscales000000 -$] telnet www.cisco.com 80
Trying 23.196.192.158...
Connected to www.cisco.com.
Escape character is '^A'.
<^Z
^Z
Connection closed by foreign host.
[centos#webscales000000 -$]
```

Likewise, we can leverage the FTDs for network IPS and AMP capabilities.

Web Application Firewalls and DDoS Protection

Azure WAF and DDoS

Test Case: Monitor Web and DDoS activity on Azure cloud

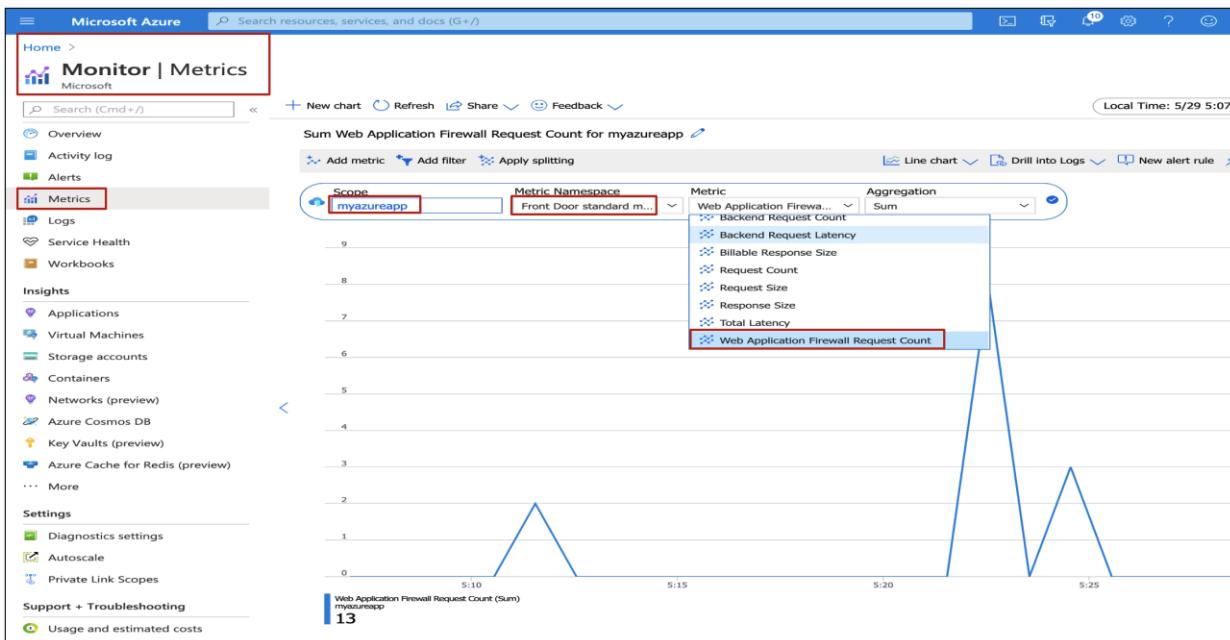
This test case involves monitoring the security events generated by the Azure cloud native WAF and DDoS service. For monitoring WAF and DDoS activity, you can either look at the Azure Metrics or do a deep dive using Azure log analytics. We will cover both the methods for each service.

Validation procedure:

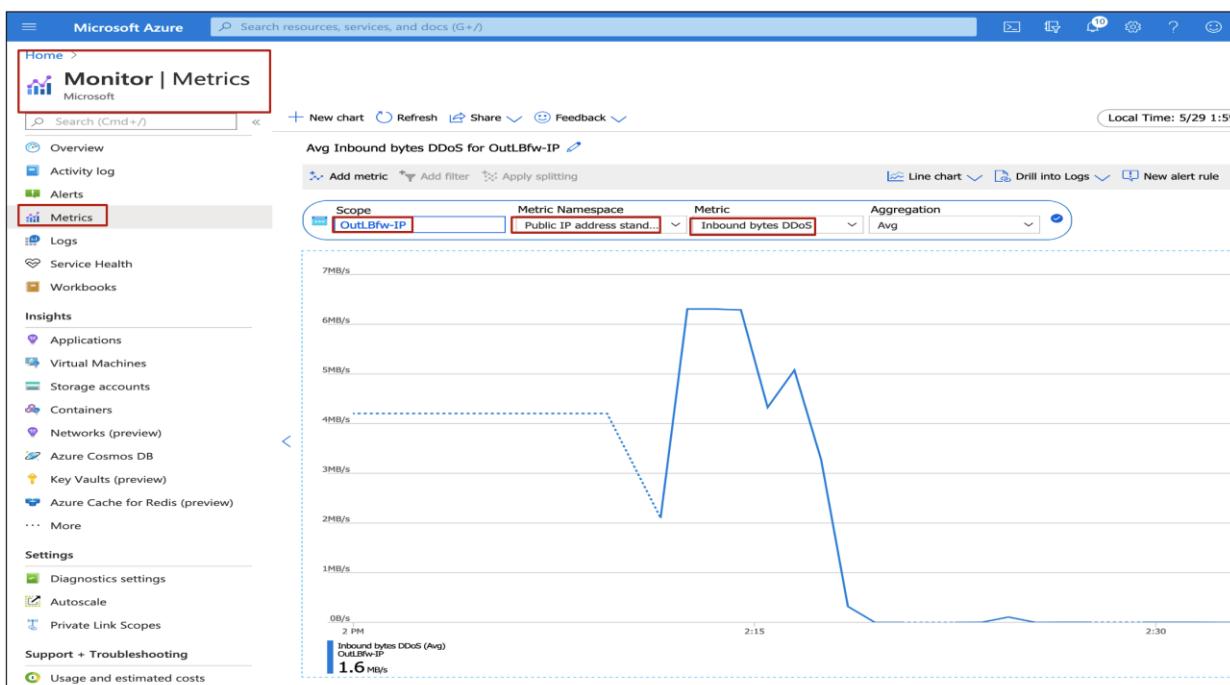
Step 1. Monitor WAF and DDoS metrics on Azure cloud

Step 2. Analyzing WAF and DDoS logs on Azure cloud

Step 1. Monitor WAF and DDoS metrics on Azure cloud - Go to 'Monitor' service on the Azure portal and select metrics option on the left-hand side panel. You will be prompted to select the Scope or the service that you want to look metrics for. For viewing the metrics associated with WAF, select Azure Front Doors service that we created earlier, and then select 'Web Application Firewall Request Count' from the drop-down under 'Metrics' option, and it will allow you to plot the WAF statistics.



In a similar manner, you can select the Public IP of the outside NGFW interfaces under Scope and plot all the DDoS related metrics.



Step 2. Analyzing WAF and DDoS logs on Azure cloud – Now, we want to dive deeper into the logs, go to ‘Log Analytics workspaces’ and create a workspace for WAF and DDoS services.

LogSpaceforWAFandDDoS
Log Analytics workspace

Overview

Status : Active	Resource group (change) : securerg	Workspace Name : LogSpaceforWAFandDDoS
Location : East US	Subscription ID : 879940a4-f36c-4f80-87a6-79b4737c80	Pricing tier : Pay-as-you-go
Subscription (change) : Microsoft Azure Enterprise	Access control mode : Use resource or workspace permissions	
Subscription ID : 9d1e5c82-29bb-43db-b312-a61a9077fa01		
Tags (change) : Click here to add tags		

Get started with Log Analytics

Log Analytics collects data from a variety of sources and uses a powerful query language to give you insights into the operation of your applications and resources. Use Azure Monitor to access the complete set of tools for monitoring all of your Azure resources.

Go to ‘Azure Front Doors’ service that we created earlier and select ‘Diagnostics Settings’ from the panel on the left-hand. Click on ‘+ Add Diagnostic setting’ to add a log subscription to the log analytics space we created above. Make sure you include ‘FrontdoorAccessLog’ and ‘FrontdoorWebApplicationFirewallLog’ to the diagnostic settings.

myazureapp | Diagnostic settings

Diagnostics settings

Name	Storage account	Event hub	Log Analytics workspace	Edit setting
FrontDoorDiagLogs	safewafstorage	-	safespaceforwaf	Edit setting

Logs

Now, go to ‘Azure Monitor’ service and select the ‘Diagnostics Settings’ on the left-hand side of the panel. Select the Public IP addresses of the NGFWv from the available selection panel and then click on ‘+ Add Diagnostic setting’ to add a log subscription to the log analytics space we created above. Make sure you include ‘DDoSProtectionNotification’, ‘DDoSMitigationFlowlogs’, ‘DDoSMitigationReports’ in the diagnostic settings.

At this point we have added both the required logs for WAF and DDoS service to our log analytics space. Now, we can run queries to look for any events related to WAF block or DDoS for a deep dive and plot custom charts and dashboards based on these queries.

TimeGenerated [UTC]	ResourceId	Category
5/29/2020, 11:01:26.528 PM	/SUBSCRIPTIONS/9D1E5C82-29BB-43DB-B312-A61A9077FA01/RESOU...	FrontdoorWebApplicationFirewallLog
5/29/2020, 11:01:27.086 PM	/SUBSCRIPTIONS/9D1E5C82-29BB-43DB-B312-A61A9077FA01/RESOU...	FrontdoorWebApplicationFirewallLog
5/29/2020, 11:01:38.796 PM	/SUBSCRIPTIONS/9D1E5C82-29BB-43DB-B312-A61A9077FA01/RESOU...	FrontdoorWebApplicationFirewallLog
5/29/2020, 11:01:39.648 PM	/SUBSCRIPTIONS/9D1E5C82-29BB-43DB-B312-A61A9077FA01/RESOU...	FrontdoorWebApplicationFirewallLog

Radware Cloud WAF and DDoS

Test Case: Monitor Web and DDoS activity on Radware cloud

This test case involves monitoring the security events generated in the Radware cloud portal.

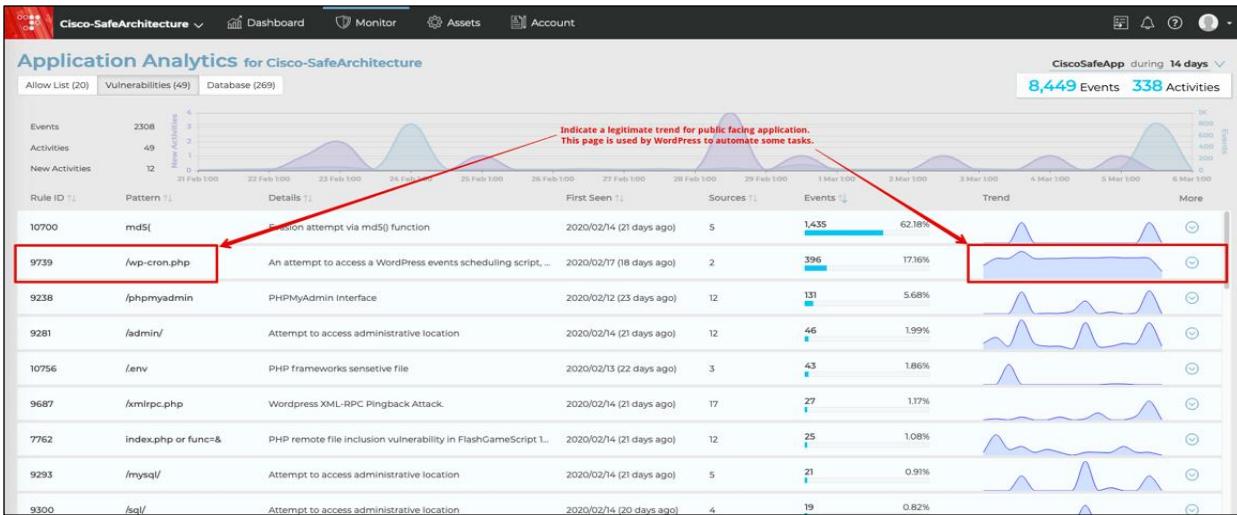
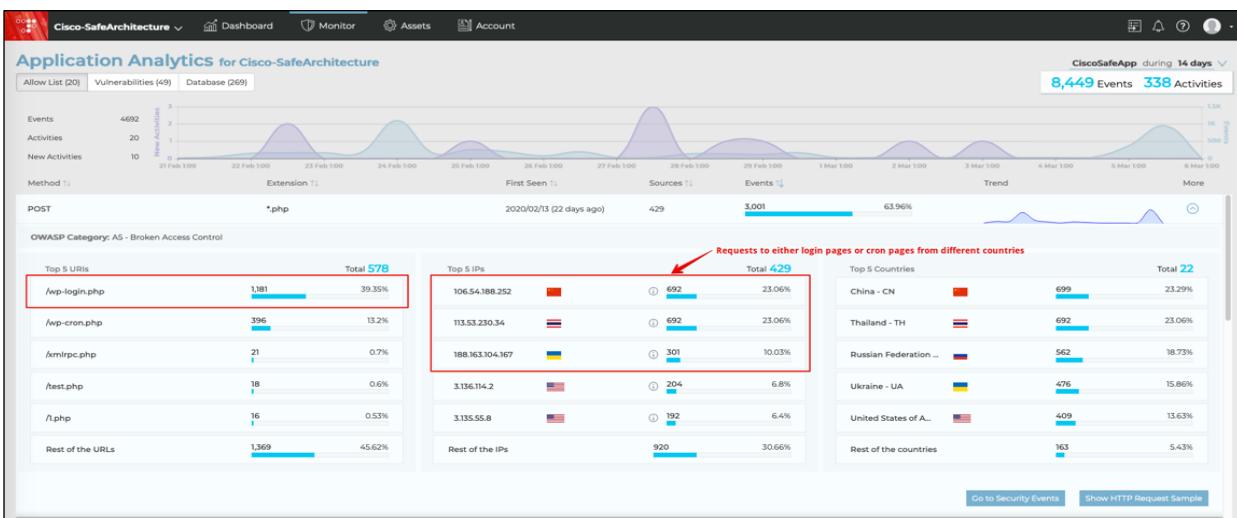
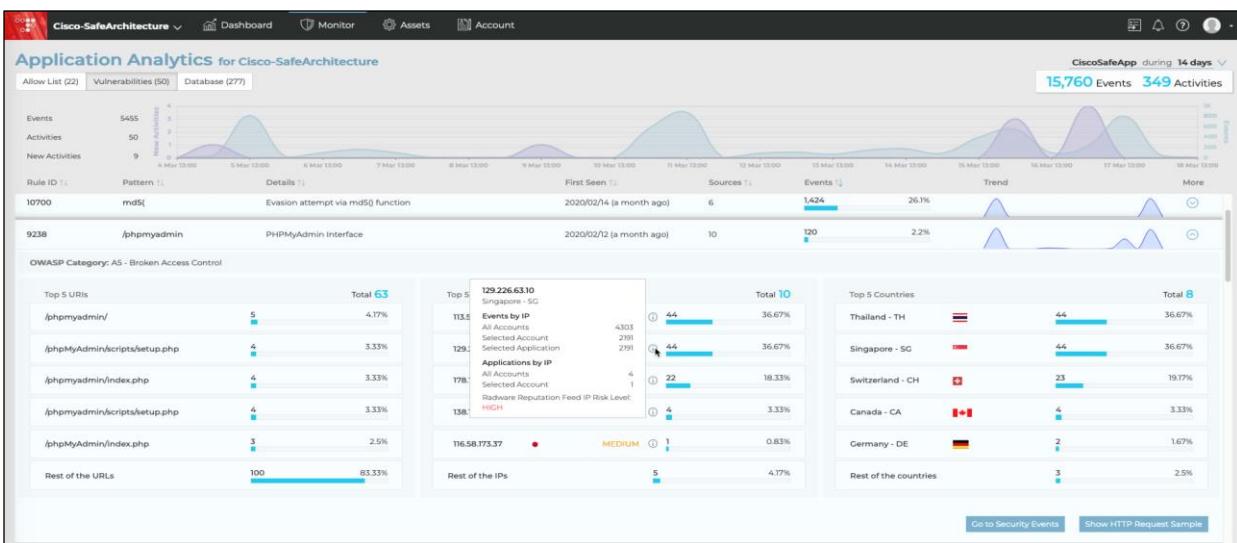
Validation procedure:

Step 1. Monitor Web and DDoS activity on Radware cloud WAF and DDoS Portal.

Step 1. Monitor Web activity and DDoS activity on Radware cloud - On the Radware Cloud portal, go to ‘Monitor > Security Events’ to see all the WAF and DDOS events generated from any malicious activity targeting your application.

Radware’s Application Analytics combines a large number of similar events and consolidating them into small, manageable sets of recurring activities. This helps to streamline response by providing additional context to security events needing attention.

In addition, the integrated ERT Active Attacker Feed will help you identify if listed requests are legitimate or not by identifying known attackers. As illustrated below, we are able to gain intelligence if the IP's attempting to access the phpMyAdmin pages are from known malicious IPs (with risk level assessments).



Duo Beyond

Validation procedure overview:

- Test Case 1 - Set up the cloud application for Two-Factor Authentication (2FA)
- Test Case 2 - Monitor 2FA activity from Duo admin portal

Test Case 1: Set up the cloud application for Two-Factor Authentication (2FA)

This test case involves logging into the application for the first time and activating the duo plugin. Previously, during the implementation phase, we had already downloaded the plugin to application workloads using custom scripts (Azure VM Extensions). Follow the [Duo documentation](#) (skip step 2 under ‘Install and Configure the Plugin’) to activate WordPress Duo plugin. After activating the plugin, log out and log in again. This time Duo will prompt the user to enroll their phone for 2FA. After successful enrollment, user gets the ability to approve subsequent login attempts.

Validation procedure:

Step 1. Set up Duo 2FA for a new user

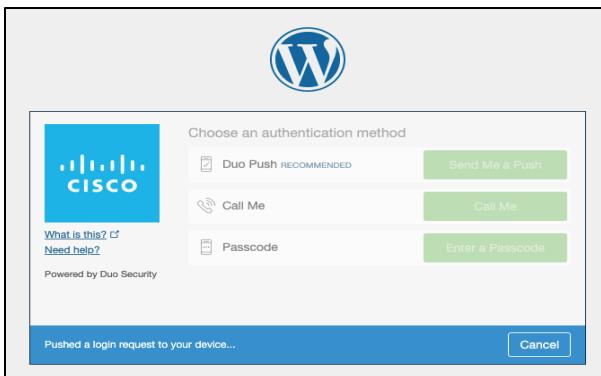
Step 2. Log onto the cloud application

Step 1. Set up Duo 2FA for a new user - After the initial plugin activation, the Duo MFA kicks in and since this is the first authentication attempt, the user is prompted to enroll for MFA.

The figure consists of four screenshots arranged in a 2x2 grid, illustrating the process of setting up Duo 2FA for a new user. Each screenshot shows a WordPress-themed interface with a Cisco logo and a 'Powered by Duo Security' watermark.

- Screenshot 1:** A 'Start setup' button is visible at the bottom right of the main content area.
- Screenshot 2:** A 'Continue' button is visible at the bottom right of the main content area.
- Screenshot 3:** An 'Enter your phone number' form is displayed, showing a dropdown for 'United States' and a text input field containing '+1'. A green checkmark indicates the number is correct.
- Screenshot 4:** A 'My Settings & Devices' page is shown, featuring a list of devices. An 'Android' device is listed with a blue progress bar labeled 'JUST ADDED'. A checkbox for 'Automatically send this device a Duo Push' is checked.

Step 2. Log onto the cloud application - After the enrollment, we continue to log onto the application, this time the user is presented with Duo authentication methods instead of ‘setup’. Once the user approves the authentication request, they are allowed to login.



Test Case 2: Monitor 2FA activity from Duo admin portal

This test case involves monitoring the 2FA enrollment and login activity in the Duo admin portal.

Validation procedure:

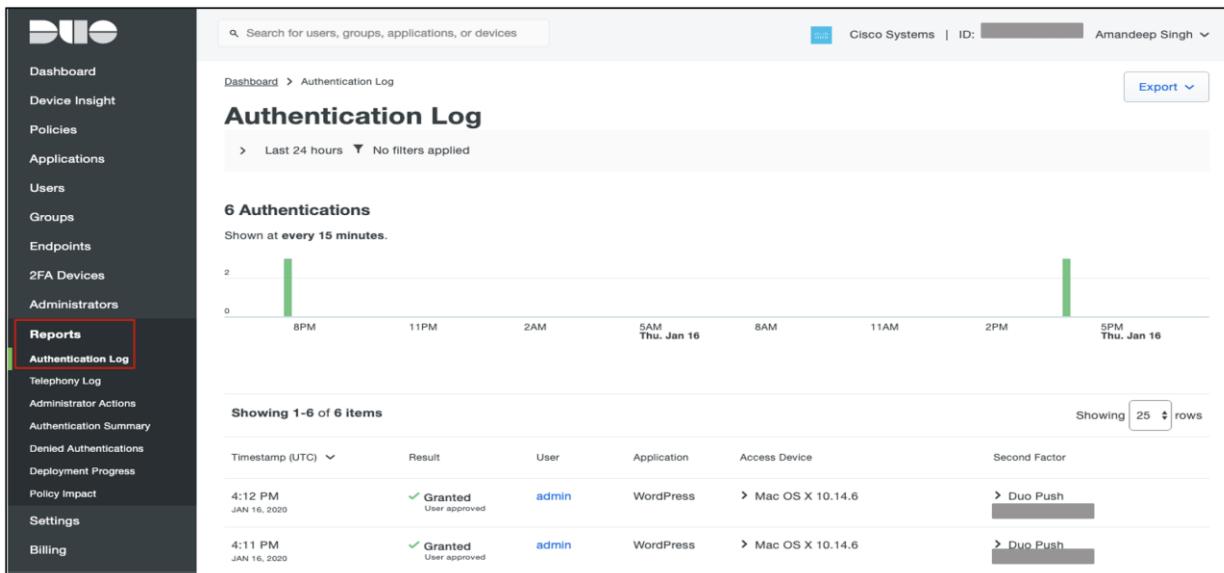
Step 1. Verify the 2FA enrolled devices

Step 2. Track the user logins in authentication logs

Step 1. Verify the 2FA enrolled devices - Logon to the Duo admin portal and select '2FA Devices', the portal shows the list of enrolled devices along with other details like platform, hardware model and usernames.

Device	Platform	Model	Duo Mobile	Security Warnings	Users
Android 8.0.0	Samsung Galaxy S7	3.30.0	No warnings	No warnings	admin, amansin3

Step 2. Track the user logins in authentication logs - Go to 'Dashboard > Authentication log', to track user 2FA login activity as shown in the snapshot below.



Cisco SecureX Threat Response

Test Case: Track malicious Activity on threat response

In this test case, we track the life cycle of the malicious PDF that we quarantined using AMP4E in previous steps. We will use the same SHA value and see what threat response offers in terms of visibility in our environment.

Implementation procedure:

Step 1. Investigate a malicious SHA value

Step 2. Track the file trajectory

Step 1. Investigate a malicious SHA value - Log on to the threat response portal and select 'Investigate'. Add the SHA value in provided space and click on 'Investigate'. Threat response pulls all the information about the associated file and what workloads the specific file had interacted with. Under the 'Observables' section, we can see that AMP4E detected this SHA value as malicious based on our custom AMP policy, threat response displays the specific AMP4E policy name as well.

The screenshot shows the Cisco Threat Response interface. At the top, there are tabs for Threat Response, Investigate (which is selected), Snapshots, Incidents, Setx, Intelligence, and Modules. Below the tabs, there are buttons for New Investigation, Assign to Incident, and Snapshots ...

On the left, a sidebar displays the following counts: 1 Target, 1 Observable, 1 Indicator, 0 Domains, 1 File Hash, 0 IP Addresses, 0 URLs, and 1 Module.

The main area shows an investigation titled "Investigation 1 of 1 enrichments complete" for file hash 5e4d40fc... The Relations Graph section shows nodes for file names like "file-example_P..." and "wget", file paths like "/home/centos/f...", SHA-256 values like "5e4d40f...", and a Target Endpoint named "webscales000000". A dropdown menu for the SHA-256 node is open, showing options like Copy to Clipboard, Create Judgement, Add to New Case, AMP for Endpoints, and File trajectory.

To the right, there are sections for Sightings (a chart showing 6 sightings from May 27, 2020, at 17:17:19 UTC to 19:46:42 UTC), Observables (listing the file hash with 6 sightings), and Judgements (a table showing an entry for AMP for Endpoints with disposition Malicious).

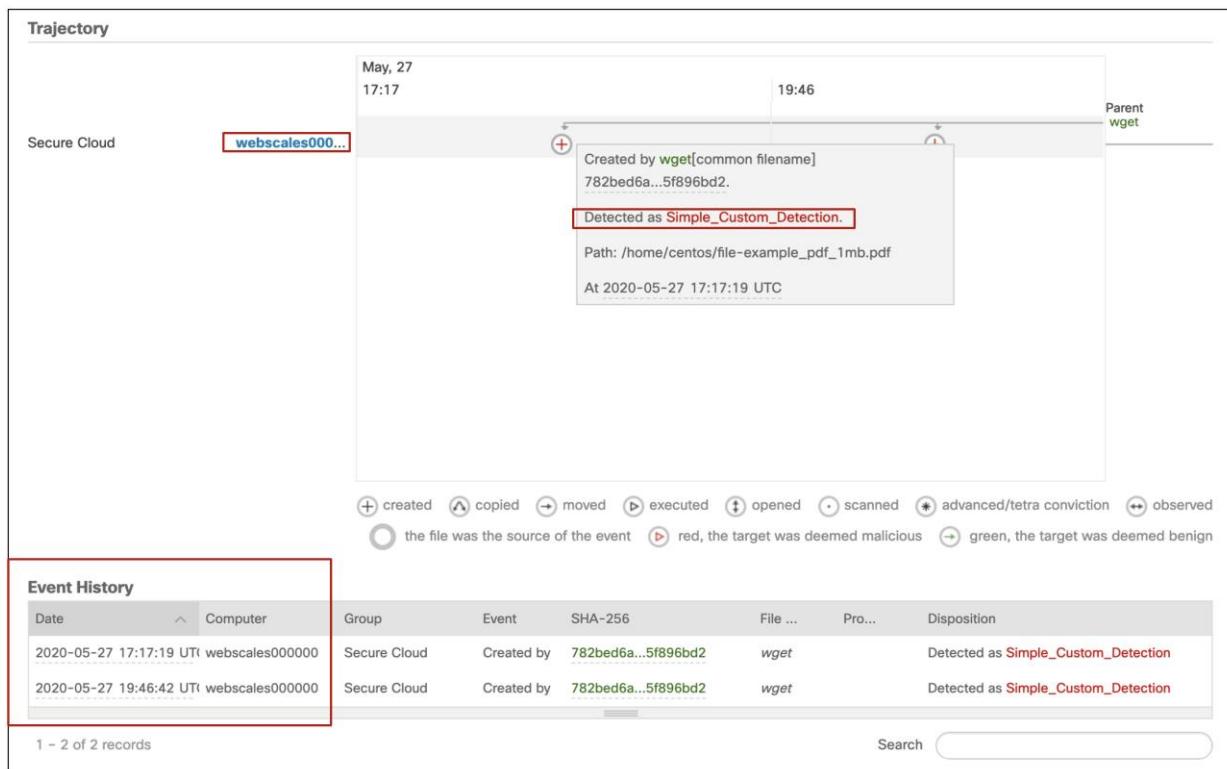
Step 2. Track the file trajectory - Click on the 'SHA-256 Hash' shown in the Relations Graph. Expand the drop-down menu and click on 'File trajectory'.

This screenshot shows the same Threat Response interface as above, but with a focus on the Relations Graph. The context menu for the SHA-256 node (5e4d40fc...) is open, and the "File trajectory" option is highlighted with a red box.

The Relations Graph now shows a "File Path Of" relationship between the SHA-256 node and the "/home/centos/f..." folder. The "File Path Of" node has a "Targeted" relationship to the Target Endpoint "webscales000000".

The Observables and Judgements sections remain the same as in the first screenshot.

Clicking on 'File trajectory' should redirect you to AMP4E portal page which displays the trajectory of the malicious file on the specific workload. Clicking on a particular timestamp displays the related events. The event history shows all the events associated with the specific file.



Appendix

Appendix A- Acronyms Defined

AFD - Azure Front Doors

AMP4E - Advanced Malware Protection for Endpoints

AVC - Application Visibility and Control

CDO - Cisco Defense Orchestrator

CSD - Custom Simple Detection

CVD - Cisco Validated Design

ERT - Emergency Response Team

FDM - Firepower Device Manager

FMC - Firepower Management Center

FTD - Firepower Threat Defense

FQDN - Fully Qualified Domain Name

IOC - Indicators of Compromise

MFA - Multi-Factor Authentication

NGFW - Next Generation Firewall

NSG - Network Security Group

PaaS - Platform as a Service

PIN - Places in Network

SaaS - Software as a Service

SWC - Stealthwatch Cloud

VA - Virtual Appliance

VNET - Virtual Network

VSS - Virtual Machine Scale Set

2FA - Two Factor Authentication

Appendix B- Software Versions

Product	Platform	Version
Tetration	Software agent	3.3.2.35-enforcer
AMP4E	Software agent	1.11.1.663
Stealthwatch Cloud	Cloud Offering	SaaS
Umbrella VAs	Appliance (Azure VM Instance)	2.6.2
Cisco NGFWv	Appliance (Azure VM Instance)	6.5.0-115
CDO	Cloud Offering	SaaS
Duo WordPress Plugin	Software Plugin	Version 2.5.5
Radware Cloud	Cloud Offering	SaaS
SecureX threat Response	Cloud Offering	SaaS
Workloads (Azure VSS)	Linux	CentOS 7.7
Azure Database for MySQL	MySQL database	mysql-5-7

Appendix C- References

This section lists all the references.

- **Cisco SAFE:**
https://www.cisco.com/c/en/us/solutions/enterprise/design-zone-security/landing_safe.html
- **Azure N-Tier Architecture:**
<https://docs.microsoft.com/en-us/azure/architecture/guide/architecture-styles/n-tier>
- **Cisco Tetration:**
<https://www.cisco.com/c/en/us/products/security/tetration/index.html>
- **Cisco Stealthwatch Cloud:**
<https://www.cisco.com/c/en/us/products/security/stealthwatch-cloud/index.html>
- **Cisco AMP for Endpoint:**
<https://www.cisco.com/c/en/us/products/security/amp-for-endpoints/index.html>

- **Cisco Firepower Threat Defense:**
https://www.cisco.com/c/en/us/td/docs/security/firepower/quick_start/azure/ftdv-azure-qsg.html
- **Cisco Duo Beyond:**
<https://duo.com/docs/wordpress>
- **Cisco Umbrella:**
<https://docs.umbrella.com/deployment-umbrella/docs/deploy-vas-on-microsoft-azure>
- **Cisco Defense Orchestrator:**
<https://www.cisco.com/c/en/us/products/security/defense-orchestrator/index.html>
- **Radware for AWS (WAF and DDoS):**
<https://www.radware.com/products/cloud-waf-service/>
- **NGINX:**
<https://www.nginx.com/resources/wiki/start/topics/recipes/wordpress/>
- **WordPress:**
<https://wordpress.org/download>
- **Azure VNET:**
<https://docs.microsoft.com/en-us/azure/virtual-network/>
- **Azure Route Tables:**
<https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-udr-overview>
- **Azure NSG:**
<https://docs.microsoft.com/en-us/azure/virtual-network/security-overview>
- **Azure Database for MySQL:**
<https://docs.microsoft.com/en-us/azure/mysql/>
- **Azure VSS:**
<https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/overview>
- **Azure Virtual Machines:**
<https://azure.microsoft.com/en-us/services/virtual-machines/>
- **Azure Load balancer (Internal and External):**
<https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview>
- **Azure Storage Containers:**
<https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction>
- **Azure Templates:**
<https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/overview>
- **Azure Private Links:**
<https://docs.microsoft.com/en-us/azure/private-link/private-link-overview>
- **Azure Front Doors:**
<https://docs.microsoft.com/en-us/azure/frontdoor/>

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at <https://www.cisco.com/go/offices>.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/go/trademarks>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)