paloalto NETWORKS

Securing Applications in AWS

REFERENCE ARCHITECTURE GUIDE

MARCH 2022

Table of Contents

Preface

GUIDE TYPES

Overview guides provide high-level introductions to technologies or concepts.

Reference architecture guides provide an architectural overview for using Palo Alto Networks® technologies to provide visibility, control, and protection to applications built in a specific environment. These guides are required reading prior to using their companion deployment guides.

Deployment guides provide decision criteria for deployment scenarios, as well as procedures for combining Palo Alto Networks technologies with third-party technologies in an integrated design.

DOCUMENT CONVENTIONS

Notes provide additional information.

Cautions warn about possible data loss, hardware damage, or compromise of security.

Blue text indicates a configuration variable for which you need to substitute the correct value for your environment.

In the **IP** box, enter `10.5.0.4/24`, and then click **OK**.

Bold text denotes:

- Command-line commands.

  ```
  # show device-group branch-offices
  ```

- User-interface elements.

  In the **Interface Type** list, choose **Layer 3**.

- Navigational paths.

  Navigate to **Network** > **Virtual Routers**.

- A value to be entered.

  Enter the password **admin**.

Italic text denotes the introduction of important terminology.

> An *external dynamic list* is a file hosted on an external web server so that the firewall can import objects.

Highlighted text denotes emphasis.

> Total valid entries: 755

ABOUT PROCEDURES

These guides sometimes describe other companies' products. Although steps and screen-shots were up-to-date at the time of publication, those companies might have since changed their user interface, processes, or requirements.

GETTING THE LATEST VERSION OF GUIDES

We continually update reference architecture and deployment guides. You can access the latest version of this and all guides at this location:

https://www.paloaltonetworks.com/referencearchitectures

WHAT'S NEW IN THIS RELEASE

Palo Alto Networks made the following changes since the last version of this guide:

- Updated PAN-OS® to 10.1.3, including the VM-Series firewall capacities and system requirements
- Changed phrasing, terminology, and diagrams for clarity

Purpose of This Guide

This reference architecture guide describes how your organization can use Palo Alto Networks VM-Series firewalls to bring visibility, control, and protection to applications built on Amazon Web Services (AWS).

This guide:

- Links the technical design aspects of AWS and the Palo Alto Networks solutions and then explores several technical design models. The design models include two options that span the scale of enterprise-level operational environments.
- Provides a framework for architectural discussions between Palo Alto Networks and your organization.
- Provides an overview of how Prisma™ Cloud helps organizations manage security risks and compliance in a public-cloud infrastructure.
- Is required reading prior to using the Palo Alto Networks AWS deployment guides. The deployment guides provide decision criteria for deployment scenarios, as well as procedures for enabling features of AWS and the Palo Alto Networks VM-Series firewalls in order to achieve an integrated design.

AUDIENCE

This guide is for technical readers, including system architects and design engineers, who want to deploy the Palo Alto Networks VM-Series firewalls and Panorama™ within a public-cloud data center infrastructure. This guide assumes the reader is familiar with the basic concepts of applications, networking, virtualization, security, and high availability. The reader should also possess a basic understanding of network and data center architectures.

To be successful, you must have a working knowledge of networking and policy in PAN-OS.

RELATED DOCUMENTATION

The following documents support this guide:

- **Zero Trust Enterprise: Reference Architecture Guide**—Describes the Zero Trust Enterprise approach to securing users, applications, and infrastructure by eliminating implicit trust and continuously validating every stage of a digital interaction.
- **Public Cloud Security Overview**—Describes key challenges in approaching public-cloud security and securing cloud-native applications. It details how organizations can leverage the Palo Alto Networks portfolio to discover resources, detect risks, mitigate network threats, highlight suspicious behaviors, prevent malware and data leakage, and identify host vulnerabilities.
- **Securing Applications in AWS—Centralized Model: Deployment Guide**—Details the deployment of the Centralized design model. This model provides a hub-and-spoke design for centralized and scalable firewall services for inbound, outbound, and east-west traffic flows. This guide describes deploying the VM-Series firewalls in order to provide protection and visibility for traffic flowing through the transit gateway.
- **Securing Applications in AWS—Isolated Model: Deployment Guide**—Details the deployment of the Isolated design model, which is well-suited for deployments that do not require security between virtual private clouds (VPCs). This guide describes deploying VM-Series firewalls in order to provide visibility and protection for inbound and outbound traffic flows in one or more isolated VPCs.
- **Panorama on AWS: Deployment Guide**—Details the deployment of Palo Alto Networks Panorama management nodes in the AWS VPC. This guide includes setup of Panorama in a high-availability configuration and setup of Cortex™ Data Lake.

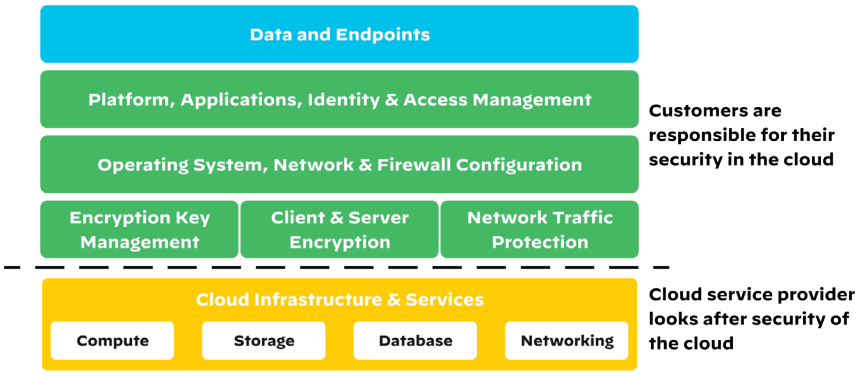
Introduction

Organizations are deploying applications and services on AWS public-cloud infrastructure for a variety of reasons, including:

- **Business agility**—Infrastructure resources are available when and where you need them, minimizing IT staffing requirements and providing faster, predictable time-to-market. Virtualization in both public- and private-cloud infrastructure has permitted IT organizations to respond to business requirements within minutes instead of days, weeks, or months.
- **Better use of resources**—Projects are more efficient, and there are fewer operational issues, permitting employees to spend more time adding business value. Employees have the resources they need, when they need them, to bring value to the organization .
- **Operational vs capital expenditure**—Costs align directly with resource usage, providing a utility model for IT infrastructure requiring little-to-no capital expense. Gone are the large capital expenditures and time delays associated with building private data center infrastructure.

Although Infrastructure as a Service (IaaS) providers are responsible for ensuring the security and availability of their infrastructure, organizations are ultimately still responsible for the security of their applications and data. The security requirements are similar to on-premises deployments, but the specific implementation details of how to properly architect and deploy security technologies in a public-cloud environment, such as AWS, are different.

Figure 1 Security responsibility in the IaaS environment



The VM-Series firewall is an integral security-enforcement and intelligence-gathering component of the Palo Alto Networks product portfolio. The Palo Alto Networks VM-Series firewall deployed on AWS has the same features, benefits, and management as the PA-Series next-generation firewalls you might have deployed elsewhere in your organization. First and foremost, the application-control and threat-prevention capabilities of the VM-Series firewalls protect your AWS workloads and application deployments from threats, data loss, and business disruption. Any observed and collected threat-intelligence information is shared across other portfolio components to improve threat prevention capabilities collectively and continually.

Public-Cloud Concepts

Organizations generally move to the public cloud with the goals of increasing scale and reducing time to deployment. Achieving these goals requires application architectures that are built specifically for the public cloud. Before you can architect for the public cloud, you must understand how it is different from traditional on-premises environments.

SCALING METHODS

Traditionally, organizations scale on-premises deployments through the purchase of devices that have increased performance capacity. Scaling up an on-premises deployment in this method makes sense because organizations typically purchase the devices in order to satisfy the performance requirements during the devices' lifetime.

Public-cloud environments focus on scaling out the deployment instead of scaling up. This architectural difference stems primarily from the capability of public-cloud environments to dynamically increase or decrease the number of resources allocated to your environment. In the public cloud, infrastructure used to satisfy performance requirements can have a lifetime in minutes instead of years. Instead of purchasing extra capacity for use at some time in the future, the dynamic nature of the public cloud allows you to allocate just the right amount of resources required to service the application.

In practice, to architect an application for the cloud, you need to distribute functionality, and you should build each functional area to scale out as necessary. Typically, this means a load balancer distributes traffic across a pool of identically configured resources. When changes occur in the application traffic, the number of resources you have allocated to the pool can dynamically increase or decrease. This design method provides scale and resiliency. However, the application architecture must take into account that the resources are transient. For example, you should not store the application state in the networking infrastructure or in the frontend application servers. Instead, store state information on the client or persistent storage services.

The ability to scale a cloud architecture extends not only to the capacity of an application but also the capacity to deploy applications globally. Scaling an application to a new region in a traditional on-premises deployment requires significant investment and planning. Public-cloud architectures are location-agnostic, and you can deploy them globally in a consistent amount of time.

REDUCED TIME TO DEPLOYMENT

To achieve the goal of reduced time to deployment, you must have a development and deployment process that is repeatable and reacts to changes quickly. DevOps workflows are the primary method for implementing this process. DevOps workflows are highly dependent on the ability to automate, as much as possible, the process of deploying a resource or application. In practice, this means you must be able to programmatically bootstrap, configure, update, and destroy the cloud infrastructure, as well as the

resources running on it. Compared to traditional on-premises deployments where device deployment, configuration, and operation happen manually, automated workflows in a public-cloud environment can significantly reduce time to deployment.

Automation is so core to cloud design that many cloud application architectures deploy new capabilities through the automated build-out of new resources instead of updating the existing ones. This type of cloud architecture provides several benefits, not the least of which is the ability to phase in the changes to a subset of the traffic, as well as the ability to quickly roll back the changes by redirecting traffic from the new resources to the old.

SECURITY INTEGRATION

VM-Series firewalls enable you to securely implement scalable cloud architectures and reduce time to deployment. You leverage the following capabilities of VM-Series firewalls in order to achieve this:

- **Application visibility**—VM-Series firewalls natively analyze all traffic in a single pass to determine the application, content, and user identity. You use the application, content, and user identities as core elements of your security policy and for visibility, reporting, and incident investigation.
- **Prevent advanced attacks at the application level**—Attacks, much like many applications, can use any port, rendering traditional prevention mechanisms ineffective. VM-Series firewalls allow you to use threat prevention and the WildFire® cloud-based threat analysis service to apply application-specific threat-prevention policies that block exploits, malware, and previously unknown threats from infecting your cloud.
- **Consistent policy and management**—Panorama centralized management enables you to manage your VM-Series deployments across multiple cloud environments, along with your physical security appliances, thereby ensuring policy consistency and cohesiveness. Rich, centralized logging and reporting capabilities provide visibility into virtualized applications, users, and content.
- **Automation features that reduce time to deployment**—VM-Series firewalls include management features that enable you to integrate security into your public-cloud development projects. You can use bootstrapping to automatically deploy firewalls. After bootstrapped firewalls deploy, Panorama instances can configure the firewall and keep the firewall policy up to date. Alternatively, you can use automation tools, such as Terraform and Ansible, to deploy and configure the VM-Series firewalls, as well as AWS project resources. You can use firewall performance metrics and health information to create automated actions based on performance and usage patterns. By allowing VM-Series firewalls to consume external data, you can automate policy updates when workloads change by using the fully documented XML API and dynamic address groups. The result is that you can deploy new applications and next-generation security simultaneously in an automated manner.

CLOUD INFRASTRUCTURE PROTECTION

AWS provides basic infrastructure components and has a responsibility to ensure that each customer's workloads are appropriately isolated and ensure that the underlying infrastructure and physical environment are secure. However, the customer has the responsibility to securely configure the instances, operating systems, and any necessary applications, as well as maintain the integrity of the data each virtual machine processes and stores. This shared-responsibility model is often a point of confusion for consumers of cloud services.

Services have default configurations that might be secure upon implementation, but to ensure the integrity of the data itself, it is up to the customer to make the assessment and lock those service configurations down.

Security and compliance risks in cloud computing threaten an organization's ability to drive digital business. The dynamic nature of the cloud, the potential complexity of having multiple cloud service providers in the environment, and the massive volume of cloud workloads make security and compliance cumbersome.

Public-cloud environments use a decentralized administration framework that often suffers from a corresponding lack of any centralized visibility. Additionally, compliance within these environments is complex to manage. Incident response requires the ability to rapidly detect and respond to threats. However, public-cloud capabilities are limited in these areas.

Prisma Cloud offers comprehensive and consistent cloud infrastructure protection that enables organizations to effectively transition to the public cloud by managing security and compliance risks within their public-cloud infrastructure.

Prisma Cloud enables your organization to:

- Improve the visibility of assets and applications.
- Provide security and compliance posture reporting.
- Enforce DevOps best practices, implemented using policy guardrails.
- Implement DevOps threat monitoring, which identifies risky configurations, network intrusions, and host vulnerabilities for the management plane. This complements the capabilities of the VM-Series firewall to secure the in-line data plane.
- Perform anomaly detection to identify compromised accounts and insider threats.
- Gain forensic capabilities that permit the investigation of current threats or past incidents to quickly determine the root cause.
- Use contextual alerting in order to prioritize issues and respond appropriately.

Through proactive security assessment and configuration management that uses industry best practices, Prisma Cloud makes cloud-computing assets harder to exploit. Prisma Cloud enables organizations to implement continuous monitoring of the AWS infrastructure. It provides an essential, automated, and always up-to-date security posture status that organizations can use to make cost-effective, risk-based decisions about service configuration and vulnerabilities inherent in cloud deployments.

Organizations can also use Prisma Cloud to prevent any deployed AWS resource from falling out of compliance. Visibility into the actual security posture of the cloud prevents failed audits and the subsequent fines associated with data breaches and non-compliance.

AWS Concepts and Services

When deployed on AWS, VM-Series firewalls rely upon underlying AWS resources and functionality to integrate into the application traffic flow and protect the workload. The concepts covered in this section give an overview of AWS services relevant to VM-Series firewall deployment. For additional information, see the **AWS documentation**, the definitive source of information on these topics.

MANAGEMENT CONSOLE

AWS provides a variety of interface options for deploying and managing resources. The AWS management console provides a graphical front-end, and AWS CloudShell provides command-line control.

You use the AWS management console to deploy, manage, and monitor resources. *Resources* are the components you use to build applications and services. Resources include, but are not limited to, virtual private clouds, load balancers, virtual machine instances, and storage services.

RESOURCE GROUPS

A resource is anything you can deployed in AWS, such a virtual network, virtual machine instance, or a storage service. Resource groups are logical collections of deployed resources. The AWS management console defaults to a service view, but using resource groups, you can customize the console view by using an organizational, project, or application-based structure. To add hierarchy to organizational structures, you can nest resource groups.

GLOBAL INFRASTRUCTURE

Some key considerations around public-cloud architectures are deciding where to locate resources, designing for efficient network communication, and providing adequate isolation such that an outage in one part of the cloud does not impact the entire cloud compute environment. The AWS global platform offers the ability to deploy and manage an architecture in a cloud environment with scale, resilience, and flexibility.

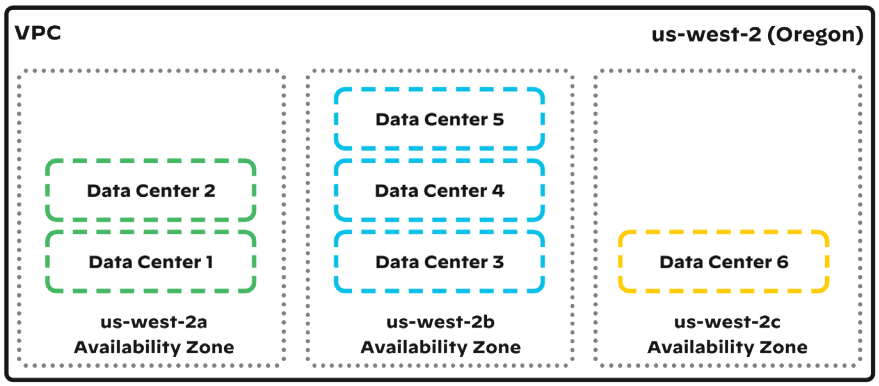
Regions

Regions enable you to place resources, services, and applications in proximity to your customers, as well as to meet government regulatory requirements for customer data residency. Regions represent collections of AWS data-center locations distributed around the globe. A region consists of several physically separate and co-located data center buildings, which provides maximum fault-tolerance and stability. The AWS global backbone provides redundant paths and encrypted transport for network communications between regions.

Availability Zones

Availability zones, or *zones*, provide a logical data center within a region. An availability zone consists of one or more physical data centers, each having separate cooling and power. All data centers within a zone connect with redundant high speed and ultra-low-latency network links. No two availability zones share a common facility and all zones within a region are separated by a substantial distance. Availability zones provide inherent fault tolerance when you distribute well-architected applications across multiple zones within a region.

Figure 2 Example of availability zones within a region



VIRTUAL COMPUTE

Virtual machines and related resources like storage are resources that you can deploy or terminate on demand. You choose a virtual machine's location by associating it with a specific region and availability zone. Most organizations deploy resources based upon an application's access and availability requirements.

Amazon Machine Image

Amazon Machine Images (AMIs) are virtual machine images available in the Amazon Marketplace. AWS publishes numerous AMIs built on different operating systems with standard software configurations for public use. In addition, members of the AWS developer community publish custom AMIs that fit more specialized requirements. You can also create your own custom AMIs. Doing so enables you to quickly and easily start new instances that have everything you need.

Instance

An *instance* is a virtual machine running in Amazon *Elastic Compute Cloud* (EC2). Much like their physical counterparts, instances allow you to choose operating systems and various performance options such as the processor type and number of CPUs, total memory, storage type and capacity, and number of network interfaces. You can change the instance type for instances that are in the stopped state. AWS bases the hourly operating costs on instance type and region.

Elastic Network Interface

An *elastic network interface* (ENI) is a virtual network interface card that you attach to an instance and that appears as network interface on the instance. Every instance type has a maximum number of network interfaces. For example, the m5.xlarge instance commonly used for a Palo Alto Networks VM-Series firewall supports 4 network interfaces.

Instance network interfaces receive IP addresses, default gateways, and DNS servers from the AWS DHCP service. Static IP addressing is available when you require a persistent IP address. When there is only one IP address on an interface, you do not need to configure static IP addresses in the operating system running on the instance. Instead, you set up that static IP address in AWS. When you configure a static IP address, the instance still receives the IP address through DHCP. However, unlike dynamic IP address allocation, when started, the instance uses the configured IP address, and when stopped, the instance does not release the IP address. The next time the instance starts, the IP address remains the same.

An elastic network interface can include the following attributes:

- A primary private IPv4 address from the address range of your VPC
- One dynamic or elastic public IPv4 address per private IP address
- One or more IPv6 addresses
- One or more security groups
- Source/destination check

Within the same availability zone, you can detach and reattach ENIs to another instance up to the maximum number of interfaces supported by the instance type. The ENI characteristics are then associated with its newly attached instance.

Elastic IP Address

Elastic IP addresses are public IP addresses that belong to AWS or a customer-owned IP address pool. Public IP addresses are associated with the network interface of an instance. After they are associated, AWS configures network address translation in the VPC's internet gateway (IGW), which provides a 1:1 translation between the public IP address and the network interface's private IP address. When an instance is in the stopped state, the public IP address is unreachable from the internet but remains associated with the instance unless you intentionally move or delete it.

VIRTUAL PRIVATE CLOUD

An AWS *Virtual Private Cloud* is a logically segmented network within AWS that allows connected resources to communicate with each other. VPCs are associated with a specific region and span that region's availability zones.

When deploying a new VPC, you specify a classless inter-domain routing (CIDR) IPv4 address block that you can then divide into subnets. VPC IP address blocks are reachable only within the VPC or through services connected to the VPC, such as a VPN. Because AWS isolates VPCs from each other, you can overlap IP address blocks across VPCs. IPv4 address blocks can be any valid IPv4 address range with a network prefix in the range of /16 (65,535 hosts) to /28 (16 hosts). The actual number of host addresses available to you on any subnet is less than the maximum because AWS reserves some addresses for services. You cannot change a VPC's original address block, but you can add secondary address blocks. It's recommended you choose a CIDR prefix that exceeds your anticipated address space requirements for the lifetime of the VPC. There are no costs associated with VPC CIDR address-block sizing, and your VPC is visible only to you.

The primary considerations when choosing a VPC CIDR address block are the same as with any enterprise network:

- Anticipated number of IP addresses needed within a VPC
- IPv4 connectivity requirements across all VPCs
- IP address overlap in your entire organization—that is, between your AWS environment and your organization on-premises IP addressing or other IaaS clouds that you might use

Unlike enterprise networks that are mostly static and where network addressing changes can be difficult, cloud infrastructure tends to be highly dynamic, which minimizes the need to anticipate growth requirements far into the future. Instead of upgrading the resources in a VPC, many cloud deployments build new resources for an upgrade and then delete the old ones. Regardless of network address size, the general requirement for communications across the enterprise network is for all network addresses to be unique. The same requirement applies across your VPCs. When you deploy new VPCs, consider using a unique network address space for each to ensure maximum communications compatibility between VPCs and back to your organization.

Most VPC IP address ranges fall within the private IP address ranges specified in RFC 1918. However, you can use publicly routable CIDR address blocks for your VPC. Regardless of the IP address range of your VPC, AWS does not support direct access to the internet from your VPC's CIDR address block, including a publicly routable CIDR address block. You must set up internet access through a gateway service from AWS or a VPN connection to your organization's network.

Subnets

A *subnet* identifies a portion of its parent VPC CIDR address block as belonging to an availability zone. A subnet is unique to an availability zone and cannot span multiple zones. However, an availability zone may contain several subnets. To associate resources to an availability zone, the zone must have a subnet. When you create a resource, you assign it to an availability zone by associating it to a subnet within that zone. A subnet prefix length can be as large as the configured VPC CIDR address block (VPC with one subnet and one availability zone) or as small as a /28 prefix length. Subnets within a single VPC cannot overlap.

Subnets are either *public subnets*, which means they are associated with a route table that has internet connectivity via an IGW, or they are *private subnets* that have no route to the internet. Newly deployed subnets are associated with the main route table of your VPC. In Figure 3, subnets 1 and 2 are public subnets, and subnets 3 and 4 are private subnets.

Route Tables

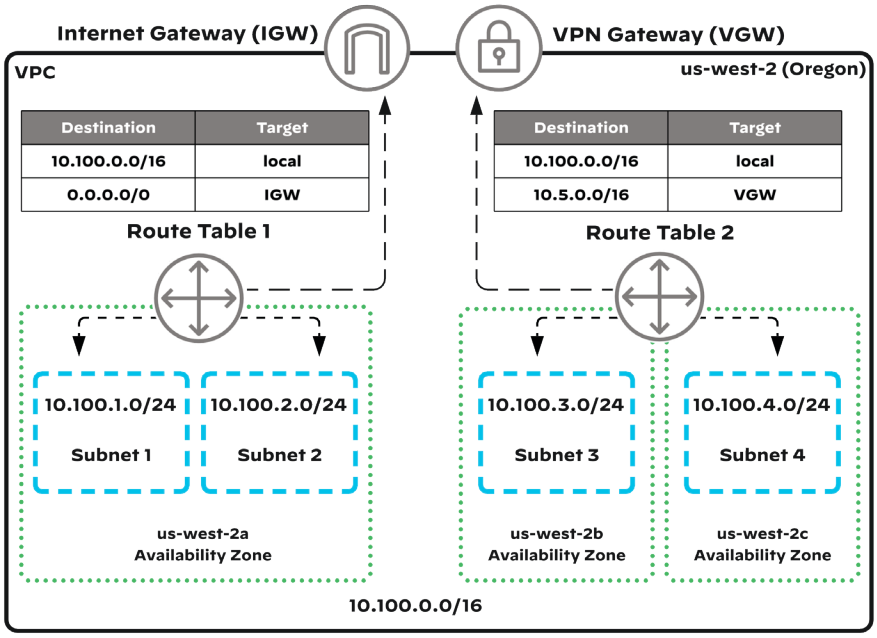
Route tables provide source-based control of Layer 3 forwarding within a VPC, which is different than traditional networking where routing information is bidirectional and might lead to asymmetric routing paths. Subnets are associated with route tables, and subnets receive their Layer 3 forwarding policy from their associated route table. A route table can have many subnets attached, but AWS only allows one route table to attach to a subnet. All route tables contain an entry for the entire VPC CIDR address block in which they reside. Any instance within the VPC has direct Layer 3 reachability to any other instance within the same VPC. This behavior has implications for network segmentation because route tables cannot contain more specific routes than the VPC CIDR address block. Any instance within a VPC can communicate directly with any other instance, and traditional network segmentation by subnets is not an option. An instance references the route table associated with its subnet for the default gateway but only for destinations outside the VPC. Host routing changes on instances are not necessary to direct traffic to a default gateway, because this is part of route table configuration. Routes external to your VPC can have a destination that directs traffic to a gateway or another instance.

Route tables can contain dynamic routing information learned from Border Gateway Protocol (BGP). Routes learned dynamically show in a route table as Propagated=YES.

A cursory review of route tables might give the impression of functionality similar to virtual routing and forwarding, but this is not the case. All route tables contain a route to the entire VPC address space and do not permit the segmentation of routing information less than the entire VPC CIDR address space within a VPC. Traditionally, you must configure a device on a network with a default gateway in order to provide a path outside the local network. In AWS, route tables provide a similar function without the need to change instance configuration to redirect traffic.

Note in Figure 3 that both route tables 1 and 2 contain the entire VPC CIDR address block entry. Route table 1 has a default route pointing to an IGW, and route table 2 has no default route. A route to 172.16.0.0/16 was learned via BGP, which is reachable via its virtual private gateway (VGW). Subnets 1 and 2 are assigned to availability zone 2a, subnet 3 is assigned to availability zone 2b, and subnet 4 is assigned to availability zone 2c.

Figure 3 Subnets and route tables



There are limits to how many routes can be in a route table. The default limit of non-propagated routes in the table is 50, and you can increase to a limit of 1000. However, this might impact network performance. The limit to routes advertised by BGP into the VPC is 100, and you cannot increase this limit. Use IP address summarization upstream or a default route to address scenarios where more than 100 propagated routes might occur.

Network Access Control Lists

Because every route table contains a route to the entire VPC, you must use network access-control lists (ACLs) to restrict traffic between subnets within your VPC. *Network ACLs* provide up to Layer 4 control of network traffic inbound and outbound from subnets in a VPC. To define the inbound and outbound rules, the ACLs use source/destination IP addresses, protocols, and ports. When you deploy a VPC, there is a default network ACL associated with it, which permits all traffic. Network ACLs do not provide control of traffic to Amazon-reserved addresses (the first four addresses of a subnet) or of link-local networks (169.254.0.0/16), which AWS uses for VPN tunnels.

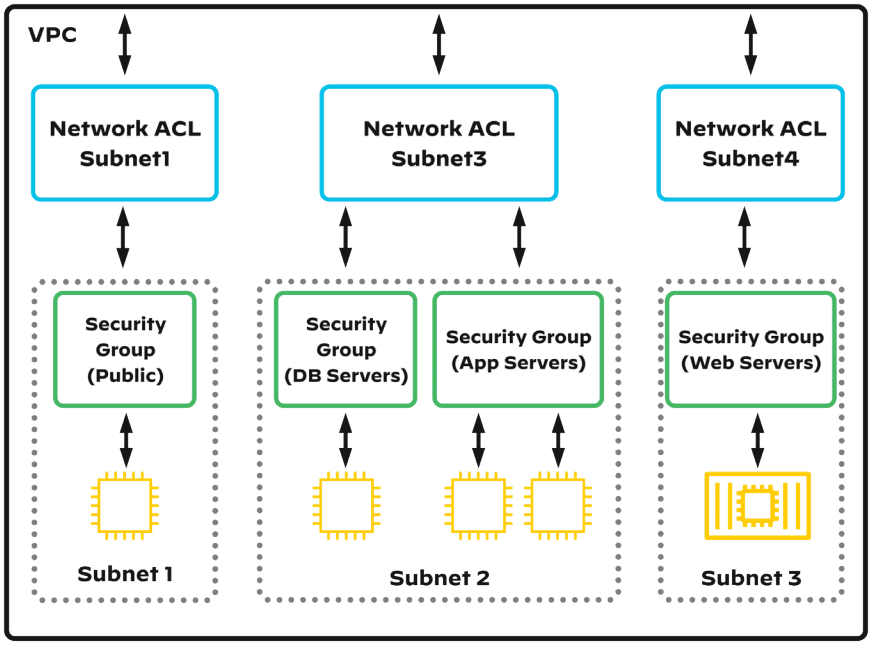
Network ACLs:

- Are applied at the subnet level.
- Have separate inbound and outbound policies.
- Have allow and deny action rules.
- Are stateless—bidirectional traffic must be permitted in both directions.
- Are order dependent—the first match rule (allow/deny) applies.
- Apply to all instances in the subnet.
- Do not filter traffic between instances within the same subnet.

Security Groups

Security groups (SGs) provide a Layer 4 stateful firewall for control of the source/destination IP addresses, protocols, and ports that are permitted. You apply SGs to an instance's network interface, up to five SGs per interface. Amazon Machine Images, which are available in the AWS Marketplace, have a default SG associated with them. SGs define the network traffic that should be explicitly permitted and deny any traffic not explicitly permitted. They have separate rules for inbound and outbound traffic from an instance network interface. SGs are *stateful*, meaning that they also permit return traffic associated with permitted rules. SGs can control traffic on any protocol that has a standard protocol number. When you deploy a new SG, the default setting contains no inbound rule, and the outbound rule permits all traffic. The effect of this default is to permit all outbound network traffic originating from your instance and its associated return traffic and to deny external traffic that is inbound to an instance. Figure 4 illustrates how you can apply network ACLs to traffic between subnets and SGs to network interfaces.

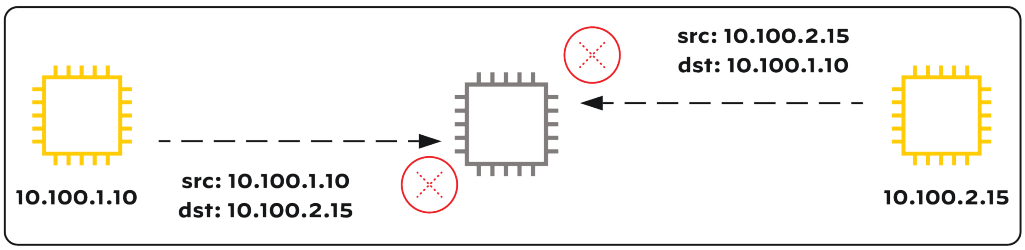
Figure 4 Security groups and network access control lists



Source and Destination Check

AWS enables source and destination checks by default on all network interfaces. The source/dest check feature validates whether traffic is destined to, or originates from, an instance and prevents any traffic that does not meet this validation. A network device (like a virtual firewall) that forwards traffic between its network interfaces within a VPC but is not the source or destination must have the source/dest check feature disabled on all interfaces that forward traffic.

Figure 5 Source and destination check



ACCESSING VPCS

Many use-cases require access to and from the AWS VPC. For example, instances in the VPC need to download their applications, patches, and data from existing customer data centers or vendor sites on the internet. Users might also need inbound access from the internet or need remote private network access to instances in AWS that provide application services. There are many access methods available with AWS; the below sections cover the most commonly used options.

Internet Gateway

An *internet gateway* IGW provides a mapping of an internal VPC IP address to a public IP address owned by AWS. The IGW maps an IP address to an instance for inbound and outbound network access. The public IP address can be:

- Random and dynamic, which means that AWS assigns the IP address to an instance at startup and returned to the pool when you stop the instance. Every time you start the instance, AWS assigns a new address from its pool.
- Random and assigned to an instance as part of a process, which means that the IP address stays with the instance unless you intentionally assign it to another instance or delete it and return it to the pool. This type of public IP address is known as an *Elastic IP address*.

This 1:1 private-to-public IP address mapping is part of a network interface configuration of each instance. After deploying a network interface, you can then associate a dynamic or an Elastic IP address to create the 1:1 IP address translation between public and VPC private IP addresses.

For internet connectivity to your VPC, the VPC must have an associated IGW. The IGW is a horizontally scaled, redundant, and highly available service. After it is associated, the IGW resides in all availability zones of your VPC, available to map to any route table or subnet where direct internet access is required.

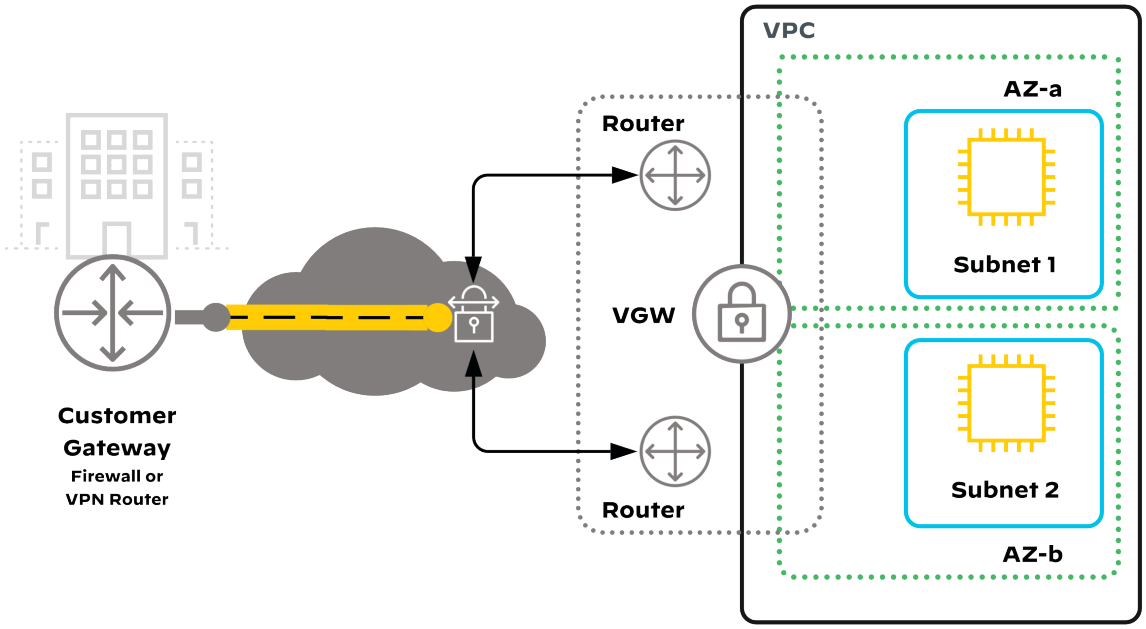
Virtual Private Gateway

A *virtual private gateway* (VGW) provides a VPN service to a VPC for the termination of IPSec tunnels. The tunnels provide confidentiality of traffic in transit and support peering to almost any device capable of supporting IPSec. Like with IGWs, the VGW resides in all availability zones of your VPC, available to map to any route table where VPN network access is required. You can map to the remote-site routes in the route table statically, or the VGW can learn them dynamically.

A *customer gateway* (CGW) identifies the target IP address of a peer device that terminates IPSec tunnels from the VGW. The customer gateway is typically a firewall or a router and must be capable of supporting an IPSec tunnel with required cryptographic algorithms.

VPN connections are the IPSec tunnels between your VGW and CGW. VPN connections represent two redundant IPSec tunnels from a single CGW to two public IP addresses of the VGW in your subscriber VPC.

Figure 6 VPN connections



AWS Gateway Route Tables

By default, inbound packets that enter through the IGW or VGW communicate directly with instances in the VPC. AWS gateway route tables allow you to control the default traffic flow of inbound traffic from an IGW or VGW. A *gateway route table* is a standard route table, only it is associated with a gateway instead of a subnet. You can use it to redirect inbound traffic to a firewall for security inspection and control.

AWS Direct Connect

AWS Direct Connect allows you to connect your network infrastructure directly to your AWS infrastructure by using private, dedicated bandwidth. You can connect from your data center or office via a dedicated link from a telecommunications provider, or you can connect directly in a colocation facility where AWS has a presence. This direct connection provides some advantages:

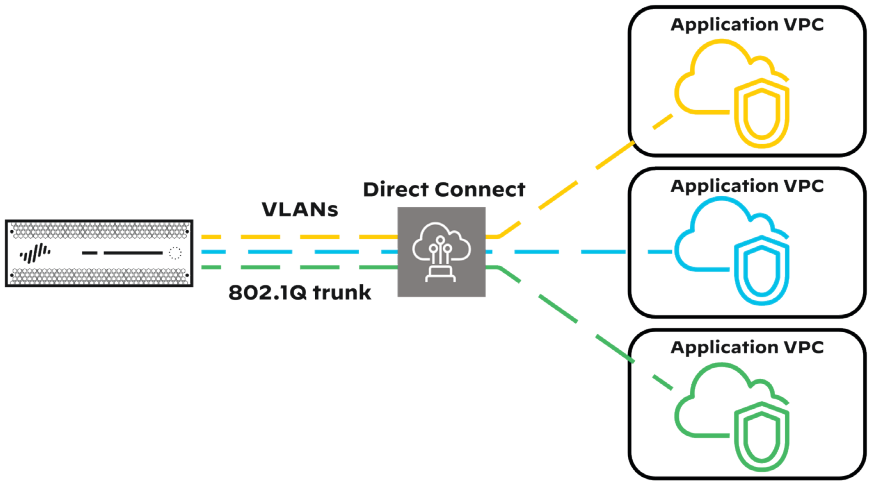
- Support for physical firewall hardware
- Higher-bandwidth network connections
- Lower-bandwidth costs
- Consistent network-transport performance
- Arbitrary inter-VPC traffic inspection and enforcement

AWS Direct Connect terminates the network connection from the AWS backbone network on a network device. This same device also terminates your carrier connection, completing the path between your private network and your AWS infrastructure. Your firewalls exchange BGP routing information with the AWS network infrastructure. Static routing is not available.

You can place your network equipment directly in an AWS regional location, using a direct connection to their network, or you can use a network provider service, such as LAN extension or MPLS, to extend your AWS infrastructure to your network.

When using a direct connection or LAN extension AWS extends the private virtual interfaces configured on the Direct Connect port over 802.1Q trunk links to your on-premises firewall, one VLAN for each VPC to which you are mapping, as shown in Figure 7.

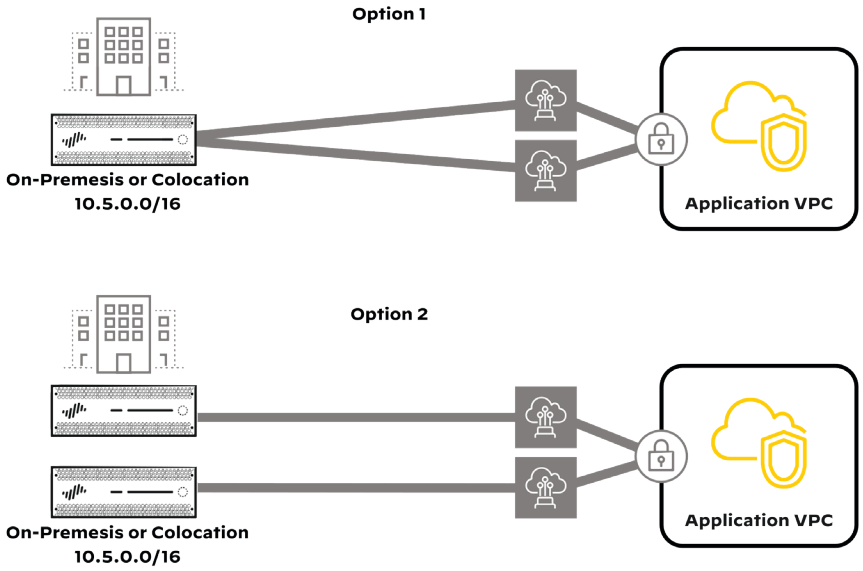
Figure 7 Direct Connect private virtual interfaces



Direct Connect provides many options for device and location redundancy. Figure 8 illustrates two of these options. The first option provides redundant connections from a single firewall to your VPC. When you require multiple instances of Direct Connect, AWS distributes them across redundant AWS backbone infrastructure. The BGP peering IP addresses must be unique across all virtual interfaces connected to a VGW. The VLAN is unique to each AWS Direct Connect port because it is only locally significant, so virtual interfaces can share a common VLAN.

The second option illustrates redundant firewalls distributed across two separate AWS locations servicing the same region. This option provides redundancy for devices, geographic location, and service providers.

Figure 8 Direct Connect redundancy options



AWS Direct Connect Gateway

The AWS Direct Connect gateway complements AWS Direct Connect, Transit Gateway, or Virtual Private Gateway by allowing you to connect one or more of your VPCs to your on-premises network through a single connection, whether those VPCs are in the same or different AWS regions.

The Direct Connect gateway:

- Can be deployed in any public region.
- Can be accessed from most public regions.
- Is a globally available resource.

VPCs connected to the Direct Connect gateway cannot communicate directly with each other. You configure the on-premises gateway to create a BGP peer connection to the Direct Connect gateway, not to every VPC. A single peering connection at the gateway eliminates the configuration overhead and monitoring tasks that would otherwise be required to support BGP peering for each VPC.

INTERCONNECTING VPCS

You can use VPCs to separate workloads across functional environments or administrative domains. You can connect multiple VPCs using VPC peering or an AWS transit gateway.

VPC Peering

VPC peering allows you to logically connect two VPCs within the same region. The peer relationship permits traffic only directly between the two peers and does not provide for any transit capabilities from one peer VPC through another to an external destination. VPC peering is a two-way agreement between member VPCs. It's initiated by one VPC to another target VPC, and the target VPC must accept the VPC peering relationship. The VPCs in a peering relationship can be in the same AWS account or different accounts, and a VPC can be in multiple VPC peering relationships. After you establish the VPC peering relationship, there is two-way network connectivity between the entire IP address block of both VPCs.

VPC peering ensures that traffic traversing the peering connection has source and destination IP addresses of the directly peered VPCs. AWS drops any packets that have a source or destination IP address outside of the two peered VPCs.

VPC peering architecture uses network policy to permit traffic only between two directly adjacent VPCs. The following are two example scenarios:

- **Hub-and-spoke model**—In a hub-and-spoke model, subscriber VPCs (spokes) use VPC peering with the central VPC (hub) to provide direct communications between the instances in the subscriber VPCs and the instances in the central VPC. The subscriber VPCs are unable to communicate with each other because this would require transit connectivity through the central VPC, which is not a capability supported by VPC peering. You can configure additional direct VPC peering relationships to permit communication between subscriber VPCs as required. Figure 9illustrates how a hub-and-spoke model of VPC peering connections could operate.
- **Multi-tiered application model**—For multi-tiered applications, you can use VPC peering to restrict communication to only directly adjacent application tiers. A typical three-tier application might use VPC peering to connect frontend web servers in a public-facing VPC to a VPC containing the application tier. The application-tier VPC would have another VPC peering relationship with a third VPC containing the database tier. VPC peering provides no external connectivity directly to the application or database tiers. Figure 10illustrates how a central VPC with multi-tier VPC connections could operate.

Figure 9 Hub-and-spoke model for VPC peering

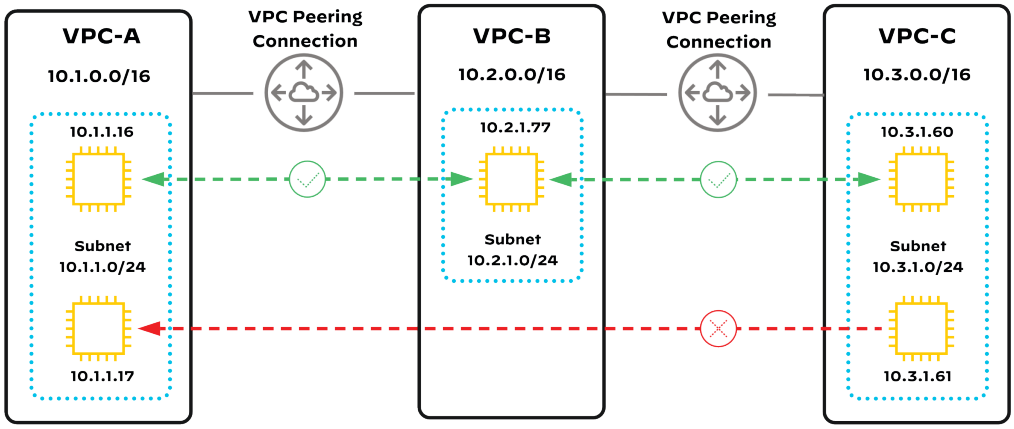
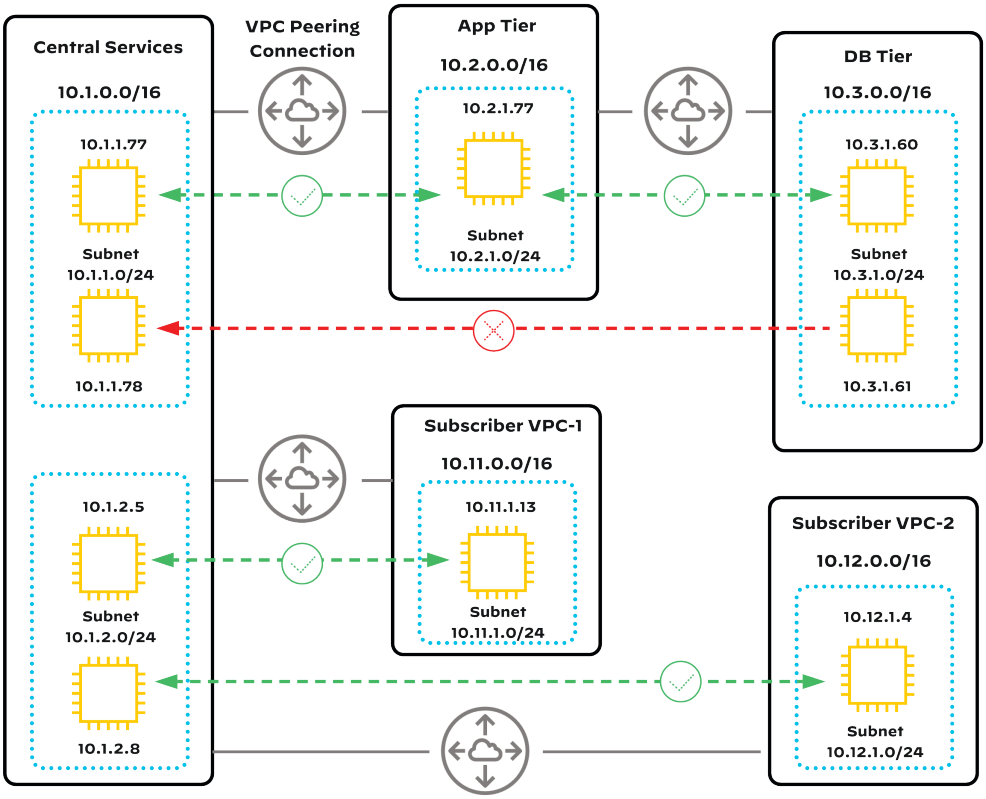


Figure 10 Multi-tiered application model for VPC peering



Transit Gateway

The AWS transit gateway (TGW) service enables you to scale interconnectivity across thousands of VPCs, AWS accounts, and on-premises networks. The TGW enables you to control communications between your VPCs and to connect to your on-premises networks via a single gateway. In contrast to VPC peering, which interconnects two VPCs only, TGWs can act as a hub in a hub-and-spoke model for interconnecting VPCs. The spokes peer only to the gateway, which simplifies design and management overhead. You can add new spokes to the gateway incrementally as your deployment grows.

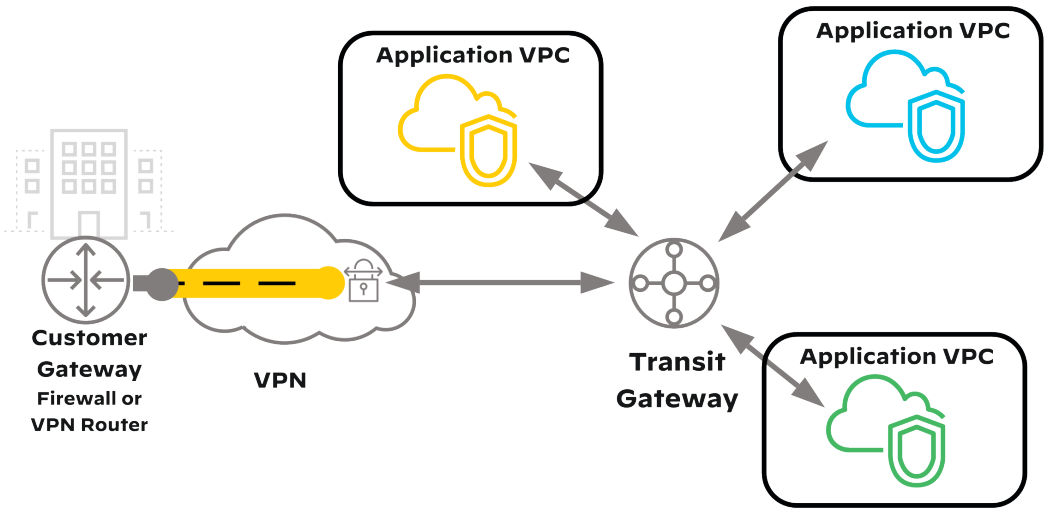
TGWs provide the ability to centrally manage the connectivity and routing between VPCs and from the VPCs to any on-premises connections via VPN or Direct Connect. TGWs allow central control of spoke-to-spoke communication and routing. TGWs support dynamic and static Layer 3 routing between VPCs and VPNs. Routes determine the next hop depending on the destination IP address of the packet, and they can point to a VPC or a VPN connection.

TGWs support the attachment of thousands of VPCs. There are two types of attachments: VPC attachments and VPN attachments.

VPC attachments are attachments from a VPC to the TGW. As part of the VPC attachment, the TGW deploys an ENI interface in each of the VPCs in-use availability zones. The TGW deploys the ENIs to subnets that you create in the attached VPCs, one subnet per availability zone. VPC attachments support only static routing and do not support Equal Cost Multipath (ECMP).

VPN attachments are either VPCs attached via VPN or VPN-attached remote sites. A VPN attachment consists of two IPSec VPN tunnels per attachment to a CGW. VPN attachments have the advantage of supporting dynamic routing and ECMP.

Figure 11 AWS transit gateway



You can configure connections between a TGW and on-premises gateways by using a VPN or Direct Connect. Because a TGW supports ECMP, you can increase bandwidth to on-premises networks by:

- Deploying multiple VPN connections between the TGW and the on-premises firewall.
- Using BGP to announce the same prefixes over each path.
- Enabling ECMP on both ends of the connections to load-balance traffic over the multiple paths.

You can use TGWs to direct inbound, outbound, and east-west flows through centralized firewalls.

SCALABILITY AND RESILIENCY CONSTRUCTS

The move to public cloud infrastructure allows customers to leverage on-demand resource scalability and incremental redundancy. AWS offers additional options to support distribution of workloads across multiple resources which allow applications and services to scale more gracefully, provide operational separation, and leverage regional resiliency.

Availability Zones

As discussed earlier in this guide, AWS provides resilience within regions by grouping data center facilities into availability zones. For resilience, any applications and services you deploy in AWS should reside in at least two availability zones. AWS services like IGW and VGW are resilient across multiple zones.

Load Balancers

Load balancers distribute traffic across a set of resources based on IP traffic criteria such as the DNS, Layer 4, or Layer 7 information. Load balancers check the traffic targets for health and remove unhealthy resources, thereby enhancing the application's resiliency. *Targets* can be instances, containers, or VPC IP addresses.

AWS offers several types of load balancers. A Gateway Load Balancer (GWLB) provides load distribution and resiliency to network appliances, such as a firewall. An Application Load Balancer functions at Layer 7 and a Network Load Balancer provides Layer 4 network traffic load balancing.

Gateway Load Balancer

A GWLB distributes traffic across a set of network appliances, such as firewalls. The GWLB provides scaling and resiliency without terminating traffic or translating the source or destination IP address.

A GWLB is linked to one or more Gateway Load Balancer endpoints. A *GWLB endpoint* is a VPC endpoint, or network interface, with a private IP address and is associated with a subnet and availability zone. Each GWLB to endpoint connection is provided through a *VPC endpoint service*. A GWLB endpoint can be in the same VPC as the GWLB, a different VPC, or even a different account than the GWLB. An advantage of GWLB and GWLB endpoint separation is that you can place the GWLB endpoint in a VPC separate from the VPC that contains the GWLB. GWLB endpoints are availability zone specific, and you can deploy them only into a zone in which the GWLB is also deployed. The endpoint service automatically deploys in the same availability zones as the GWLB.

To get traffic to a GWLB, you direct network traffic, via a VPC route table, to a Gateway Load Balancer endpoint. The VPC deployed GWLB endpoint sends all traffic it receives to the GWLB endpoint service through an AWS PrivateLink connection, a method for privately connecting services and VPCs. Because PrivateLink is used to connect the endpoint to the GWLB, traffic is sent directly to the GWLB and does not follow the routing rules of your VPC infrastructure, nor do the two VPCs even need to be connected.

After traffic is received from the endpoint, the GWLB transparently forwards traffic to the target appliances without translating the source or destination IP addressing. To perform forwarding with no address translation, the GWLB encapsulates the traffic using the Geneve protocol before sending it to the appliance. Encapsulation allows the GWLB and appliance to keep the traffic's original IP addressing intact. The GWLB adds additional metadata about the flow to the encapsulation header, which allows the GWLB to keep state for the traffic flow when it returns from the appliance. Return traffic from the appliance is accepted by the GWLB and forwarded back to the GWLB endpoint that received the incoming traffic. The route table associated with the GWLB endpoint directs the traffic to its final destination.

The GWLB selects a healthy appliance for each traffic flow from its target group based on the 5-tuple of the traffic flow. Using a 5-tuple ensures a traffic flow always ends up on the same target appliance. To ensure appliance availability, the GWLB uses health checks. Health checks are destined to the appliance itself and can be TCP, HTTP, or HTTPS. Health checks do not traverse the appliance; rather, they occur separately from the Geneve encapsulation.

Application Load Balancer

To distribute Layer 7 traffic to a target group, an *Application Load Balancer* (ALB) uses application layer HTTP and HTTPS information. Traffic destined for the ALB uses DNS names and not a discrete IP address. AWS assigns a fully qualified domain name (FQDN) when it deploys the load balancer. The FQDN maps to IP addresses in each availability zone. If the ALB in one zone fails or has no healthy targets and if it is tied to an external DNS like the Amazon Route 53 cloud-based DNS, then traffic is directed to an alternate ALB in the other zone.

You can configure an ALB with its listener facing the internet and load balance by using public IP addressing, or you can configure it to be internal only and load balance by using IP addresses from the VPC. The ALB targets can be any HTTP or HTTPS applications that are reachable from within the VPC. The ALB supports terminating HTTPS between the client and the load balancer, and it can manage SSL certificates.

The ALB offers content-based routing of connection requests based on either the host field or the URL of the HTTP header of the client request. ALB uses a round-robin algorithm to distribute traffic and supports a slow start when adding targets to the group in order to avoid overwhelming the application target. The ALB determines which targets are healthy based on health checks and HTTP error codes.

Network Load Balancer

To distribute Layer 4 traffic to a target group, the *Network Load Balancer* (NLB) uses transport layer TCP/UDP information.

On a single, static IP address per availability zone, the NLB accepts incoming traffic from clients and distributes the traffic to targets within the same availability zone. Monitoring the health of the targets, NLB ensures that only healthy targets get traffic. If all targets in an availability zone are unhealthy and if you have set up targets in another zone, then the NLB automatically fails-over to the healthy targets in the other availability zones.

The NLB supports a static IP address assignment, including an Elastic IP address, for the listener of the load balancer, making it ideal for services that do not use DNS and rely on the IP address to route connections. If the NLB has no healthy targets and if it is tied to Amazon Route 53 cloud-based DNS, then traffic is directed to an alternate NLB in another region. You can load balance to any IP address target that is reachable from within the VPC. This allows the NLB to load balance to any IP address and any interface on an instance.

Palo Alto Networks Design Details

VM-SERIES FIREWALL ON AWS

The Palo Alto Networks VM-Series firewall is the virtual form factor of a next-generation firewall. It can be deployed in a range of private and public-cloud computing environments. The VM-Series firewall on AWS enables you to securely implement a cloud-first methodology while transforming your data center into a hybrid architecture that combines the scalability and agility of AWS with your on-premises resources. This allows you to move your applications and data to AWS while maintaining a security posture that is consistent with the one you might have established on your physical network. The VM-Series firewall on AWS natively analyzes all traffic in a single pass to determine application, content, and user identity. The application, content, and user are core elements of your security policy and for visibility, reporting, and incident investigation.

VM-Series Firewall Models

VM-Series firewalls on AWS are available on a variety of instance configurations, varying only by overall capacity. A *capacity license* configures the firewall with a model number and associated capacity limits.

Table 1 VM-Series firewall system resources and capacity

	VM-100 equivalent	VM-300 equivalent	VM-500 equivalent	VM-700 equivalent
Instance				
AWS instance size tested (recommended)	m5.large	m5.xlarge	m5.2xlarge	m5.4xlarge
Capacities				
Firewall throughput (App-ID™ enabled)	2.1 Gbps	4.3 Gbps	9.0 Gbps	10.2 Gbps
Threat Prevention throughput	1.0 Gbps	1.9 Gbps	4.1 Gbps	7.8 Gbps
IPSec VPN throughput	0.9 Gbps	1.6 Gbps	3.0 Gbps	3.3 Gbps
System resources				
vCPUs	2	4	8	16
Memory (min)	6.5 GB	9 GB	16 GB	56 GB
Disk size (min)	60 GB	60 GB	60 GB	60 GB

Although the capacity license sets the VM-Series firewall performance limits, the size of the instance on which you deploy the firewall determines the firewall's overall performance and functional capacity. In Table 1, the mapping of the VM-Series firewall performance to AWS instance size is based on requirements for vCPU, memory, disk size, and network interfaces. If you choose an instance that provides more CPU than the model or deployment profile supports, the VM-Series firewalls do not use the additional CPU cores.

It might seem that an instance size smaller than those listed in Table 1 would be appropriate for a smaller VM-Series deployment; however, smaller instance sizes might not have enough network interfaces. AWS provides instances with two, three, four, eight, or fifteen network interfaces. Because VM-Series firewalls reserve an interface for management functionality, two-interface instances are not a viable option. Four-interface instances meet the requirement of a management, public, and private interface.

For the latest detailed information, see **VM-Series on AWS Performance and Capacity** and VM-Series Models on AWS EC2 Instances. Many factors affect performance, and Palo Alto Networks recommends you do additional testing in your environment to ensure the deployment meets your performance and capacity requirements. In general, public-cloud environments are more efficient when scaling out the number of resources versus scaling up to a larger instance size.

License Options

You purchase licenses for VM-Series firewalls on AWS through the AWS Marketplace or through traditional Palo Alto Networks channels.

Note

Whichever licensing model you chose is permanent. After you deploy them, VM-Series firewalls cannot switch between the pay-as-you-go (PAYG) and bring-your-own-license (BYOL) licensing models. Switching between licensing models requires deploying a new firewall and migrating the configuration. In the BYOL model, you can migrate between licensing agreements, including evaluation, regular, and enterprise, because they are all part of the same licensing model.

PAYG

A *pay-as-you-go* license model is also called a *usage-based* or *pay-per-use* license. You can purchase this type of license from the AWS Marketplace, and you are billed hourly.

With the PAYG license, a VM-Series firewall is licensed and ready for use as soon as you deploy it. You do not receive a license authorization code. When the firewall is stopped or terminated in AWS, the usage-based licenses are suspended or terminated.

PAYG licenses support the VM-100, VM-300, VM-500, VM-700 capacity licenses. A PAYG license applies a VM-Series capacity license based on the selected instance size. The PAYG instance checks the number of hardware resources available to the instance and applies the largest VM-Series firewall capacity license allowed for the resources available. For example, if the instance has 2 vCPUs and 16 GB of memory, a VM-100 capacity license is applied based on the number of vCPUs. However, if the instance has 16 vCPUs and 16 GB of memory, a VM-500 license is applied based on the amount of memory.

PAYG licenses are available in the following bundles:

- **Bundle 1**—Includes the VM-Series capacity license, Threat Prevention license (IPS, AV, malware prevention), and a premium support entitlement
- **Bundle 2**—Includes the VM-Series capacity license, Threat Prevention license (IPS, AV, malware prevention), DNS Security license, GlobalProtect™ license, WildFire license, PAN-DB URL Filtering license, and a premium support entitlement

BYOL

A *bring-your-own license* (BYOL) model allows you to purchase a license from a partner, reseller, or directly from Palo Alto Networks. In a BYOL model, VM-Series firewalls support all capacities, support entitlements, and subscription licenses. The BYOL model uses Software NGFW Credits for licensing.

Software NGFW Credits are term-based credits. The term is between 1 and 5 years, which you can use to fund Software NGFWs (VM-Series and CN-Series, Cloud-Delivered Security Services, and virtual Panorama appliances.

You allocate credits by creating a deployment profile in the support portal. The profile's capabilities are dependent on the PAN-OS version you deploy:

- **VM-Series firewalls on PAN-OS version 10.0.3 and earlier**—Compatible with legacy licenses based on **VM-Series Models** and security service bundles.
- **VM-Series firewalls on PAN-OS version 10.0.4 and later**—Flexible number of vCPUs (from 1-16) and flexible selection of security services. You modify the deployment profile to add or decrease the number of vCPUs or add new services as they become available.

If you stop using a firewall, a security service, or Panorama deployment, the credits that were allocated to that resource are refunded to the credit pool and you can reallocate them to a new resource.

You license VM-Series firewalls like a traditionally deployed appliance and apply a license authorization code. The license authorization code maps the firewall to the deployment profile you created. After you apply the code to the device, the device registers with the Palo Alto Networks support portal and obtains information about its capacity and subscriptions. Subscription licenses include Threat Prevention, URL Filtering, GlobalProtect, WildFire, Enterprise Data Loss Prevention, DNS Security, IoT Security, Intelligent Traffic Offload, and SD-WAN.

VM-SERIES FIREWALL INTEGRATION TO AWS

Launching a VM-Series Firewall on AWS

The **Amazon AWS Marketplace** provides a wide variety of Linux, Windows, and specialized machine images, like a Palo Alto Networks VM-Series firewall. There, you can find **AMIs for Palo Alto Networks VM-Series firewall** with various licensing options. After you select one, the AMI launch instance workflow provides a step-by-step guided workflow for all IP addressing, network settings, and storage requirements. You can provide your own custom AMIs to suit your design needs. Automation scripts for building out large-scale environments usually include AMI programming.

Bootstrapping

At deployment, VM-Series firewalls have the factory default configuration and a base software image that varies based on which deployment method you have chosen. You can manually upgrade the software and update the configuration after deploying the instance, or if you are using a BYOL licensing model, you can use bootstrapping to license, configure, and update the firewall software at boot time. *Bootstrapping* allows you to create a repeatable process of deploying VM-Series firewalls.

You can bootstrap the VM-Series firewall with a basic configuration or a complete configuration.

A basic configuration includes just enough information to get the firewall operational and connected to Panorama, allowing Panorama to push the additional configuration required to make the firewall operational. You bootstrap a basic configuration by configuring AWS user data on the VM-Series firewall instances.

You bootstrap a complete configuration through a bootstrap package. The package can contain everything required to make the firewall ready for production. In AWS, you implement the bootstrap package through an AWS S3 file share that contains directories for configuration, content, license, and software. On the first boot, VM-Series firewalls mount the file share and use the information in the directories to configure and upgrade the firewall. After the firewall is out of the factory default state, it stops looking for a bootstrap package.

One of the fundamental design differences between traditional and public-cloud deployments is the lifetime of resources. One method of achieving resiliency in public-cloud deployments is through the quick deployment of new resources and the quick destruction of failed resources. One of the requirements for achieving quick resource build-out and tear-down is current and readily available configuration information for the resource to use during initial deployment. When the configuration is static, the simplest method of achieving this for VM-Series firewalls is to use bootstrapping to configure the firewall policies during firewall deployment.

Management Interface

The first interface attached to the instance (eth0) is the firewall's management interface. In most templates, this interface has an Elastic IP address and DNS hostname associated with it in addition to the internal IP address in the VPC. The firewall's management interface obtains its internal IP address through DHCP. The IGW translates the internal IP address to the public IP address when the traffic leaves the VPC. Because IP addresses might change, use an FQDN to manage the firewall.

Note

If you assign a dynamic public IP address on the instance eth0 interface and later assign an Elastic IP address to any interface on the same instance, upon reboot, AWS also replaces the dynamic IP address on eth0 with an Elastic IP address. The best way to predictably assign a public IP address to your eth0 management interface is to associate an Elastic IP address to the eth0 interface.

If you are managing the firewalls with Panorama, you might not need an Elastic IP address on the firewall's management interface. If Panorama is in the AWS environment or deployed on-premises with VPN connectivity to the VPC, Panorama can use internal IP addresses to manage the VM-Series firewalls.

Managing Deployments with Panorama

The best method for ensuring up-to-date firewall configuration is to use Panorama for the central management of firewall policies. Panorama simplifies consistent policy configuration across multiple independent firewalls through its device group and template stack capabilities. When multiple firewalls are part of the same device group, they receive a common ruleset. Because Panorama enables you to control all of your firewalls—whether they are on-premises or in the public cloud or whether they are a physical or virtual appliance—device groups also provide configuration hierarchy. With device group hierarchy, lower-level groups include the policies of the higher-level groups. Configuration hierarchy allows you to configure consistent rulesets that apply to all firewalls, as well as consistent rulesets that apply to specific firewall deployment locations such as the public cloud.

As bootstrapped firewalls deploy, they can also automatically pull configuration information from Panorama. VM-Series firewalls use a VM authorization key and Panorama IP address in the bootstrap package to authenticate and register to Panorama on its initial boot. You must generate the VM authorization key in Panorama before creating the bootstrap package. If you provide a device group and template in the bootstrap package's basic configuration file, Panorama assigns the firewall to the appropriate device group and template so that the relevant rulesets are applied, and you can manage the device in Panorama going forward.

You can deploy Panorama in your on-premises data center or in a public-cloud environment such as AWS. When deployed in your on-premises data center, Panorama can manage all the PA-Series and VM-Series firewalls in your organization. If you want a dedicated instance of Panorama for the VM-Series firewalls deployed on AWS, deploy Panorama on AWS.

When you have an existing Panorama deployment on-premises for firewalls in your data center and internet edge, you can use it to manage the VM-Series firewalls in AWS. Beyond management, you need to consider your firewall log collection and retention. Log collection, storage, and analysis is an important cybersecurity best practice that organizations perform to correlate potential threats and prevent successful cyber breaches.

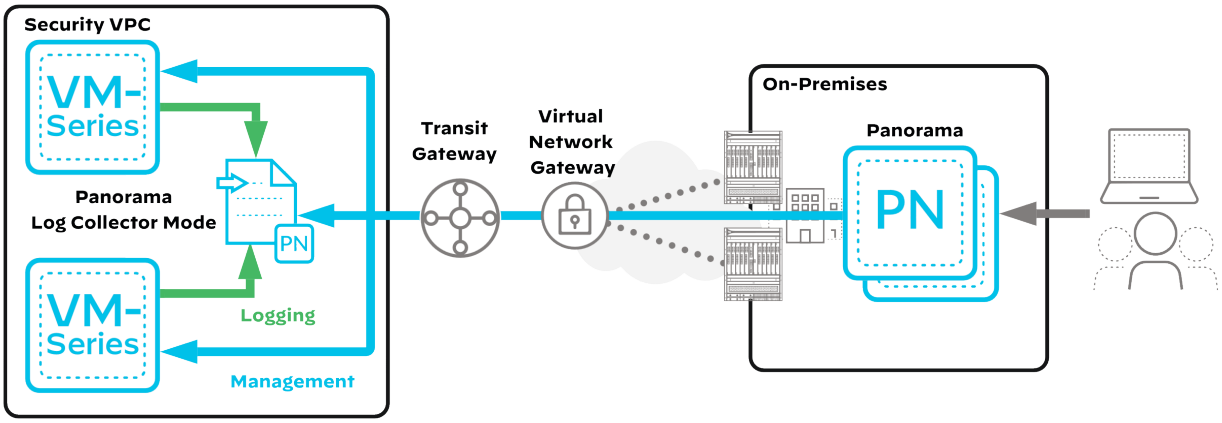
The following three deployment mode options are available for Panorama, which, if necessary, allows for the separation of management and log collection:

- **Log Collector mode**—One or more log collectors collect and manage logs from the managed devices. This assumes that another deployment of Panorama is operating in Management-Only mode.
- **Management-Only mode**—Panorama manages configurations for the managed devices but does not collect or manage logs.
- **Panorama mode**—Panorama controls both policy and log management functions for all the managed devices.

On-Premises Panorama with Dedicated Log Collectors in the Cloud

Sending logging data back to the on-premises Panorama can be inefficient, costly, and pose data privacy and residency issues in some regions. An alternative to sending the logging data black to your on-premises Panorama is to deploy Panorama dedicated log collectors on AWS and use the on-premises Panorama for management. Deploying a dedicated log collector on AWS reduces the amount of logging data that leaves the cloud but still allows your on-premises Panorama to manage the VM-Series firewalls in AWS and have full visibility to the logs as needed.

Figure 12 Panorama Log Collector mode in AWS

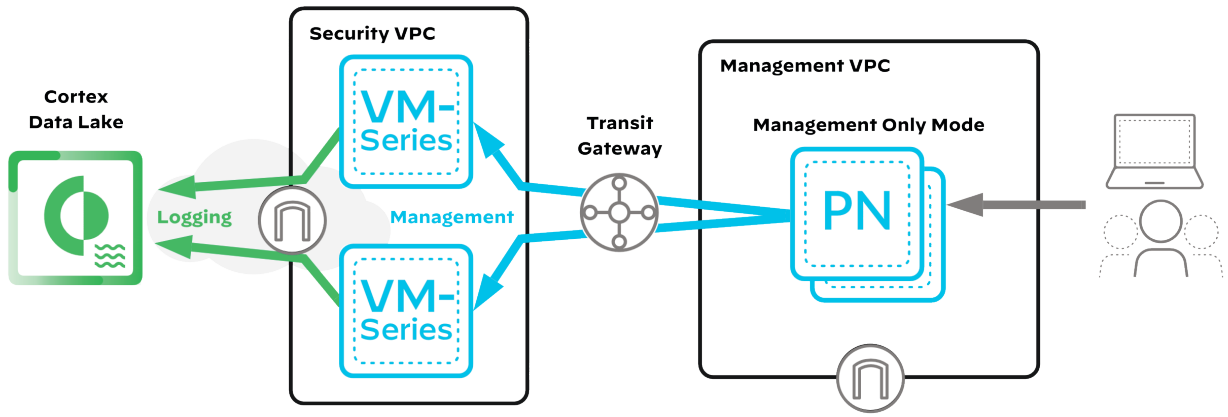


Panorama Management in AWS with Cortex Data Lake

There are two design options when deploying Panorama management on AWS. First, you can use Panorama for management only and use Palo Alto Networks Cortex Data Lake to store the logs generated by the VM-Series firewalls. *Cortex Data Lake* is a cloud-based log-collector service that provides resilient storage and fast search capabilities for large amounts of logging data. Cortex Data Lake emulates a traditional log collector. The VM-Series firewalls encrypt the logs and then send them to the Cortex Data Lake over TLS/SSL connections. Cortex Data Lake allows you to scale your logging storage as your AWS deployment scales because Cortex bases licensing on storage capacity and not the number of devices sending log data.

The benefit of using Cortex Data Lake goes well beyond scale and convenience when tied into the Palo Alto Networks Cortex AI-based continuous security platform. Cortex is a scalable ecosystem of security applications that can apply advanced analytics in concert with Palo Alto Networks enforcement points to prevent the most advanced attacks. Palo Alto Networks analytics applications such as Cortex XDR and AutoFocus™, as well as third-party analytics applications that you choose, use Cortex Data Lake as the primary data repository for all of Palo Alto Networks offerings.

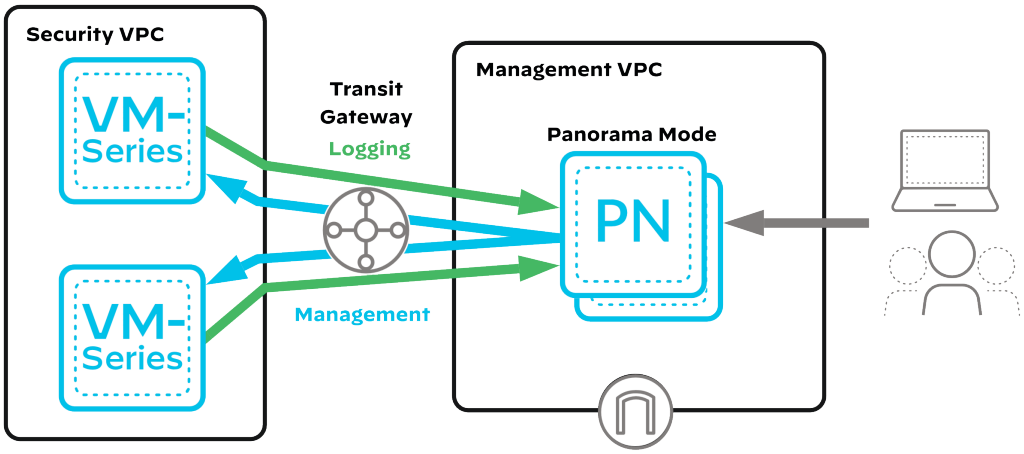
Figure 13 Panorama management and Cortex Data Lake



Panorama Management and Log Collection in AWS

Second, you can use Panorama for both management and log collection. You can deploy the management and log collection functionality as a shared virtual appliance or on dedicated virtual appliances. For smaller deployments, you can deploy Panorama and the log collector as a single virtual appliance. For larger deployments, a dedicated log collector per region allows traffic to stay within the region and reduce outbound data transfers.

Figure 14 Panorama management and log collection in AWS



Panorama is available as a virtual appliance for deployment on AWS and supports Management-Only mode, Panorama mode, and Log Collector mode with the system requirements defined in Table 2. Panorama on AWS is only available with a BYOL licensing model.

Table 2 Panorama Virtual Appliance on AWS

	Management-Only mode	Panorama	Log Collector
Minimum system requirements	16 CPUs 32 GB memory 81 GB system disk	16 CPUs 32 GB memory 2 TB to 24 TB log storage capacity	16 CPUs 32 GB memory 2 TB to 24 TB log storage capacity

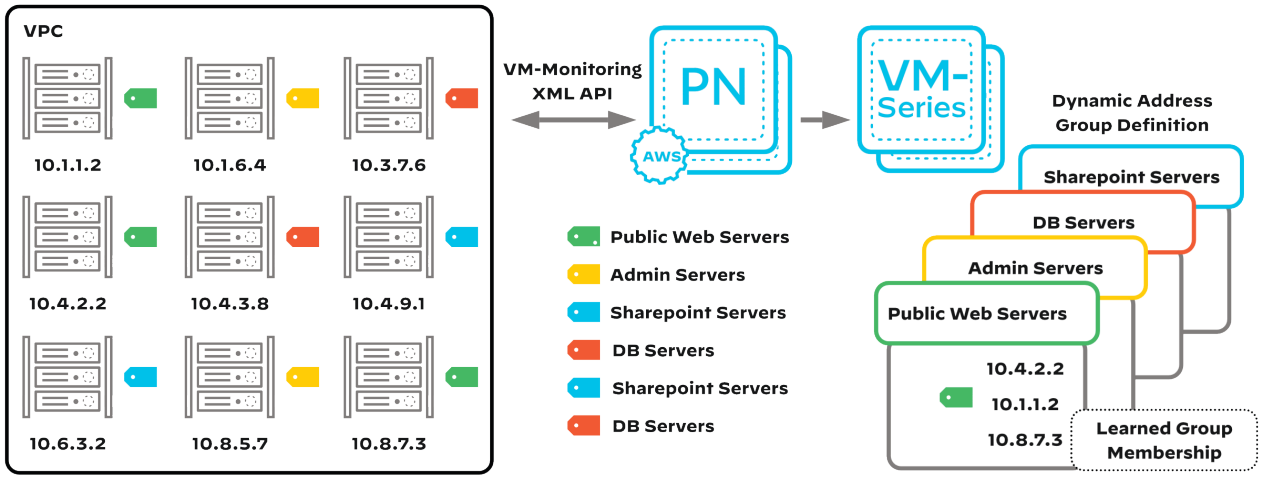
Resource Monitoring

Organizations typically build public-cloud application environments by using a continuous integration/continuous delivery (CI/CD) pipeline. Using CI/CD, you deploy applications and updates quickly and build new infrastructure to accommodate a revised application versus trying to upgrade the existing operational environment. When the new application or update goes online, you remove the now-unused, older application environment. This amount of change presents a challenge to enforcing security policy unless your security platform is compatible with an agile development and deployment process.

Palo Alto Networks firewalls, including the VM-Series, support dynamic address groups. Dynamic address groups (DAGs) allow you to create security policy that automatically adapts to instance additions, moves, or deletions. DAGs also enable the operational flexibility for applying a security policy to a device based on its role.

A dynamic address group uses tags as a filtering criterion in order to determine its members. You can define tags statically or register them dynamically. You can dynamically register the IP address and associated tags for AWS instances by using the AWS plugin on Panorama

Figure 15 VM monitoring of AWS tag to dynamic address group mappings



When using Panorama and the AWS plugin, you can centralize the retrieval of tags from AWS and security policy management in order to ensure consistent policies for hybrid and cloud-native architectures. The Panorama plugin for AWS allows you to monitor up to 1000 VPCs on AWS. Using an IAM role that you create, the plugin polls your AWS accounts for resource tags and correlates the metadata (IP address-to-tag mapping) into dynamic address groups. Panorama relays the dynamic address group content to the VM-Series firewalls, providing scale and flexibility. With the plugin, Panorama can retrieve a total of 32 tags for each virtual machine: 11 predefined tags and up to 21 user-defined tags. The number of tags used impacts the total number of IP addresses you can monitor. For example, Panorama can retrieve 10,000 IP addresses with 13 tags for each, or it can retrieve 5000 IP addresses with 25 tags for each.

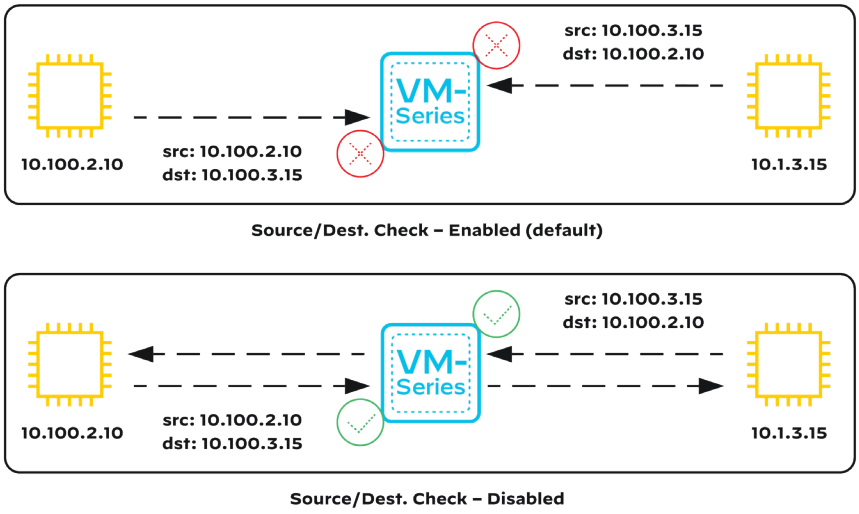
Source and Destination Check

AWS enables source and destination checks by default on all network interfaces within your VPC. The check validates whether traffic is destined to or originates from an instance, and it prevents any traffic that does not meet this validation. Any instance that forwards traffic between its network interfaces must have this check disabled on all forwarding interfaces. You must disable source and destination checking on all of the VM-Series firewall dataplane interfaces.

Note

You can change the SGs and source/dest check feature at the instance level; however, these changes apply only to the first interface of the instance. For a VM-Series firewall, the first interface represents the management interface. To avoid ambiguity, you should apply SGs and disable the source/dest check feature on the individual network interfaces (management, public, and private).

Figure 16 Source and destination check



Scale and Resiliency

Traditionally, you achieve firewall resiliency through a high-availability configuration on the firewall. In a high-availability configuration, a pair of firewalls shares configuration and state information that allows the second firewall to take over for the first if a failure occurs. Although you can configure high availability so that both firewalls are passing traffic, in the majority of deployments, the firewalls operate as an active/passive pair where only one firewall is passing traffic at a time. The VM-Series firewall on AWS does support stateful high availability in active/passive mode for traditional data center-style deployments in the cloud. However, both VM-Series firewalls must exist in the same availability zone, and it can take 60 seconds or longer for the failover to take place due to infrastructure interactions beyond the control of the firewall.

Unlike traditional implementations, you can achieve VM-Series firewall resiliency in AWS through the use of native cloud services. The benefits of configuring resiliency through native public-cloud services instead of firewall high availability are faster failover and the ability to scale out the firewalls as needed. However, in a public-cloud resiliency model, the firewalls do not share configuration and state information. Applications typically deployed in a public-cloud infrastructure, such as web- and service-oriented architectures, do not rely on the network infrastructure to track session state. Instead, they track session data within the application infrastructure, which allows the application to scale out and be resilient independent of the network infrastructure.

The AWS resources and services used to achieve resiliency for the application and firewall include:

- **Availability zones**—Ensure that a failure or maintenance event in an AWS VPC does not affect all VM-Series firewalls at the same time.
- **Load balancers**— A Gateway Load Balancer distributes traffic across two or more independent firewalls that are members of a common target group. Every firewall in the load balancer's target group actively passes traffic, allowing firewall capacity to scale out as required. The load balancer monitors the availability of the firewalls through TCP or HTTPS health checks and updates the availability of targets, as necessary.

Integration with GWLB

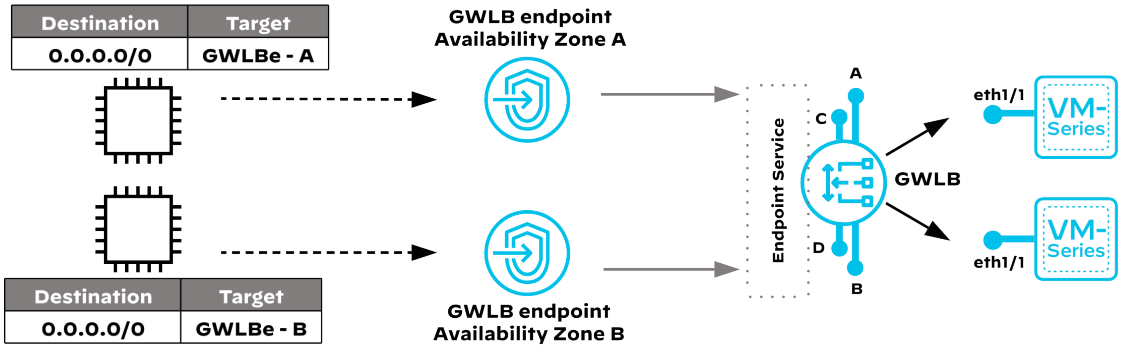
You use a GWLB to distribute traffic across VM-Series firewalls to resiliently and transparently secure inbound, outbound, and east-west traffic flows. To transparently secure traffic flows from a GWLB encapsulated in Geneve, the VM-Series firewalls use tunnel inspection. The VM-Series is not the source or destination of the application traffic flow and does not perform any IP address translation. To ensure traffic symmetry, the firewall returns the original traffic flow to the GWLB unchanged.

The GWLB is deployed in the same VPC as the VM-Series firewalls. Because you cannot modify the GWLB after deployment, you should deploy the GWLB in all of the region's availability zones. You should also configure GWLB to support cross-zone load balancing. Cross-zone load balancing ensures that the GWLB distributes traffic across all VM-Series firewalls in the target group, regardless of the availability zone on which the traffic was received. Without cross-zone load balancing enabled, the GWLB selects only VM-Series firewalls in the same availability zone where the traffic was received.

The GWLB uses a single listener and target group. You can specify the GWLB target as an instance or an IP address. If you define the target as an instance, the GWLB sends the traffic to the instance's first interface. By default, this interface is the firewall's management interface, so to make the first interface a dataplane interface, you must enable the swap-interface option on the firewall. If you define the target as an IP address, you do not have to swap VM-Series interfaces.

To get traffic to the GWLB and ultimately to the VM-Series firewalls, a VPC route table entry forwards traffic to a GWLB endpoint that is associated with the GWLB. The endpoint then forwards to the traffic to the GWLB.

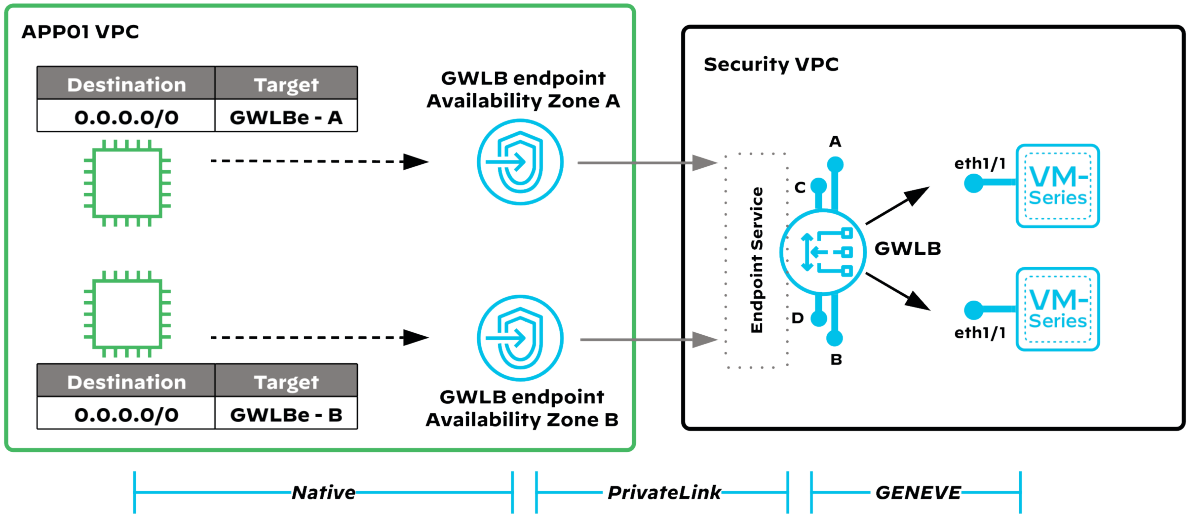
Figure 17 Traffic forwarding to the GWLB endpoint



GWLB endpoints can reside in the same VPC as the GWLB, or they can be in a separate VPC. When the GWLB endpoint is in a separate VPC, you do not need to have routing connectivity between VPCs to get the traffic to the GWLB. Instead, the communication between the GWLB endpoint and the GWLB uses AWS PrivateLink.

A GWLB supports multiple GWLB endpoints. GWLB endpoints are zone-specific, and you can deploy them only into a zone in which the GWLB is deployed. All GWLB endpoints are associated with a single endpoint service attached to the GWLB. The endpoint service terminates the PrivateLink session from the GWLB endpoints and hands the traffic off to the GWLB. The endpoint service automatically deploys in the same zones as the GWLB.

Figure 18 GWLB endpoints with GWLB



When the GWLB receives traffic from the endpoint, the GWLB picks a healthy firewall instance from its target group, encapsulates the traffic by using the Geneve protocol, adds type-length-values (TLVs) to the encapsulation that identifies the source GWLB endpoint and session information, and sends the traffic to the target firewall.

When the firewall receives the Geneve encapsulated traffic from the GWLB and aws-gwlb-inspect is enabled on the firewall, the firewall associates the encapsulated traffic to the security zone defined on the interface that received the traffic. Depending on your overall deployment and requirements, the security policy could be challenging to implement, because all traffic would be intra-zone or within the same security zone. With a single security zone VM-Series deployment, you cannot achieve a multi-tenant architecture and you cannot leverage additional VM-Series features like decryption profiles. A solution to these challenges is mapping specific GWLB endpoints to unique subinterfaces and security zones on the VM-Series firewall. You can use these mappings to easily separate VPCs, applications, and traffic flows.

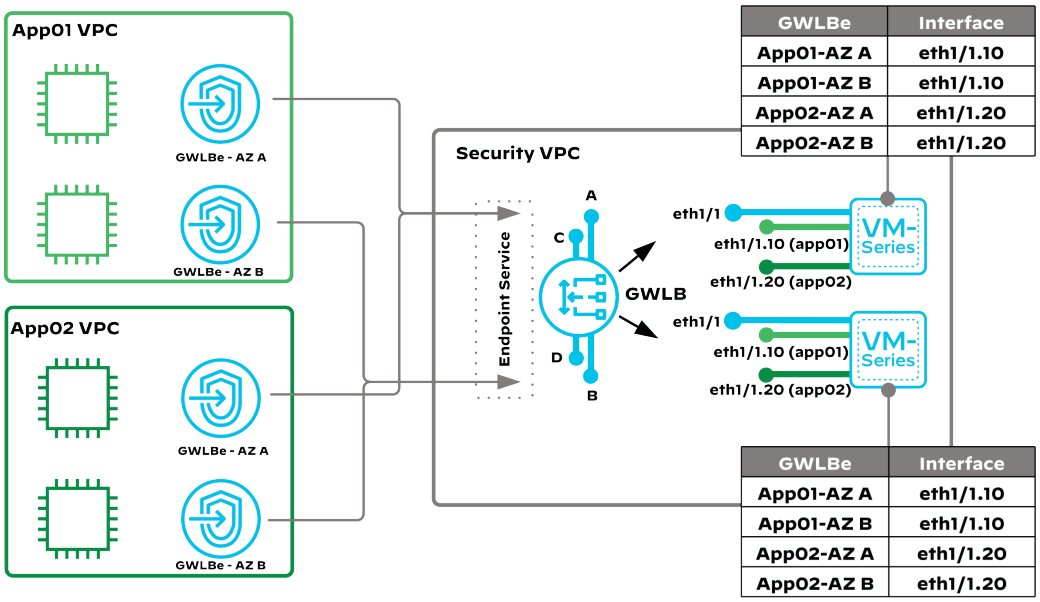
To separate traffic flows based on GWLB endpoint, you associate the GWLB endpoints to subinterfaces on the firewall. When deployed on AWS, VM-Series firewall subinterfaces are logical interfaces associated with a dataplane interface. The subinterfaces are only used to map an endpoint to a security zone; they do not get an IP address and are not used for traffic forwarding.

Because the GWLB endpoint ID is part of the Geneve header, the firewall uses the ID to map to a subinterface and apply a security zone to incoming traffic. If you have GWLB endpoints in multiple availability zones for resiliency, all servicing the same traffic flows, then to ensure the firewall applies the correct security zone, you must map all the GWLB endpoints for that traffic profile to the same VM-Series subinterface.

Caution

Traffic from unmapped GWLB endpoints shows up on the firewall's primary dataplane interface. The firewall does not drop this traffic by default. Ensure that the security policy on the primary dataplane interface is not overly permissive beyond allowing the health checks. Also, consider requiring acceptance of new GWLB endpoints at the endpoint service.

Figure 19 GWLB endpoint, subinterface, and zone mapping

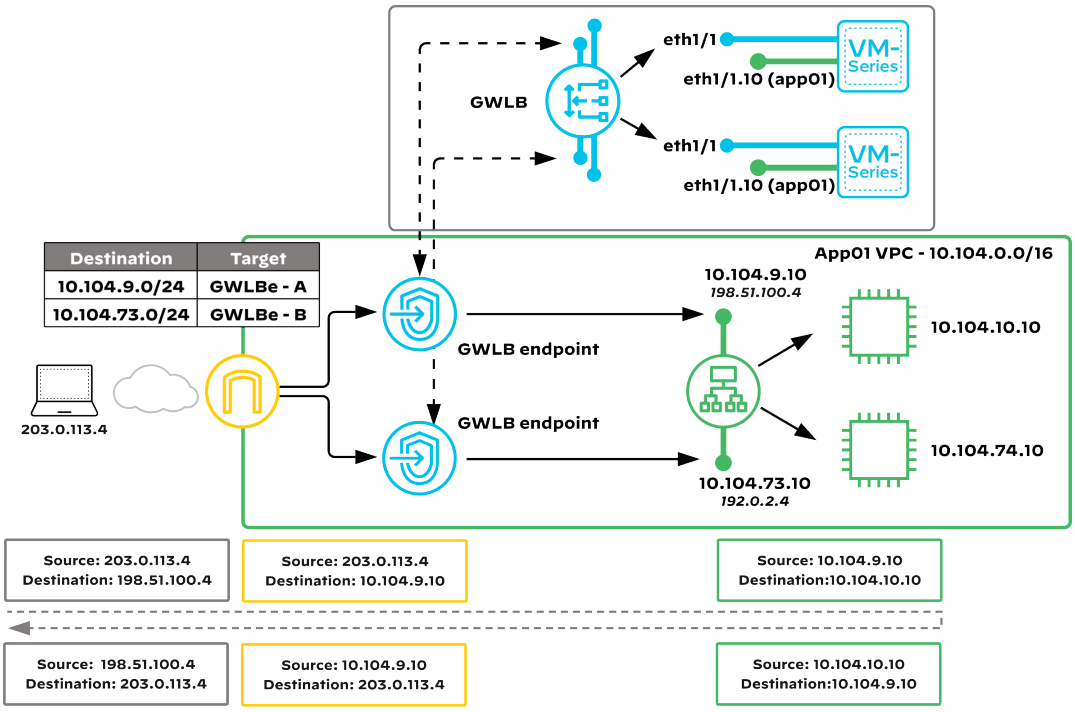


After the firewall receives traffic from the GWLB and maps it to a security zone, the firewall processes the traffic. How the packet is processed is dependent on the specific flow, the VM-Series interface configuration, and whether overlay routing is enabled on the VM-Series.

Overlay routing enables the firewall to perform a Layer 3 route lookup on the inner header of a Geneve encapsulated packet to determine the egress interface. We recommend enabling overlay routing. It allows the firewalls to support outbound traffic flows where the firewall is doing the egress address translation, enabling an easier security policy, better logging, and enabling design models where the VM-Series firewalls support outbound traffic flows without using an AWS NAT gateway.

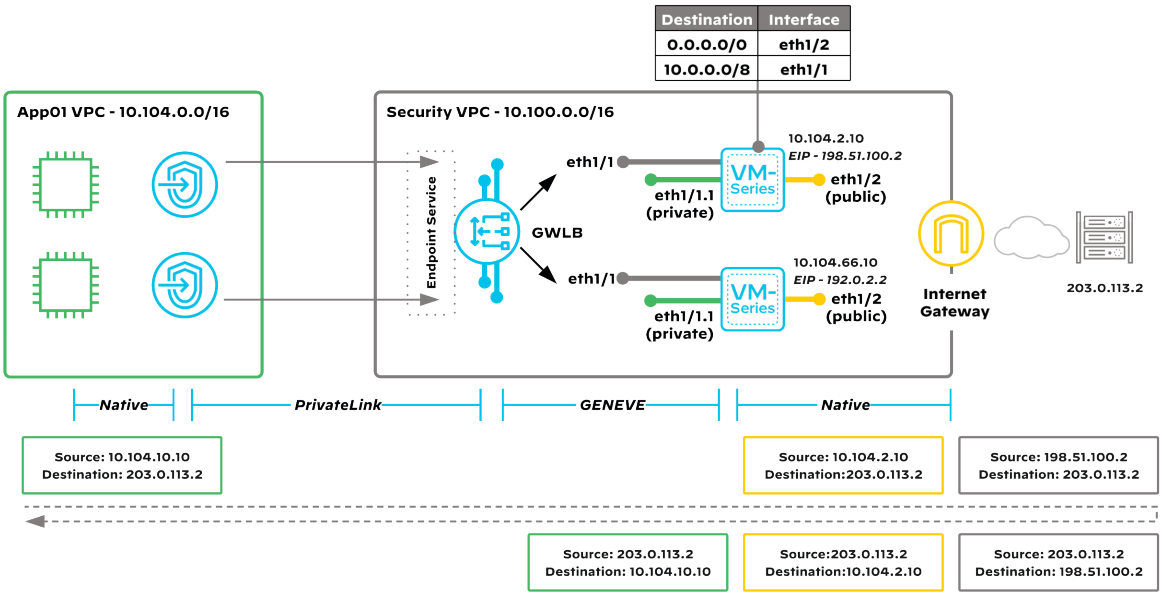
If the egress interface is the same dataplane interface on which it was received, the firewall does not use its routing table in order to determine the egress interface for the traffic. Instead, traffic egresses the same dataplane interface on which it was received and is sent back to the GWLB. This is standard operation for the VM-Series using aws-gwlb-inspect and does not require overlay routing to be enabled.

Figure 20 Single zone operation



If the egress interface is a different interface (for example, a separate VM-Series interface used for outbound traffic), the firewall sends the traffic out of the firewall egress interface, unencapsulated. Any return traffic is mapped to the Geneve session and re-encapsulated before being returned to the GWLB. This traffic flow, because it transits two interfaces, each with a unique security zone, is inter-zone traffic.

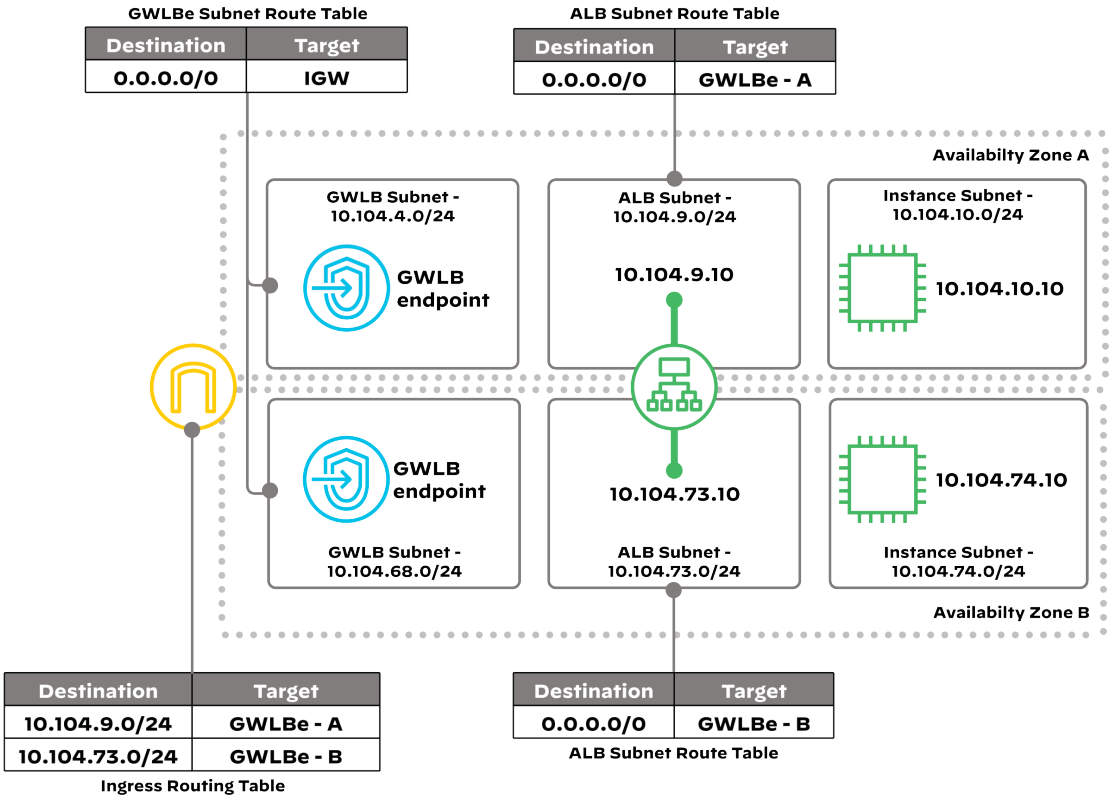
Figure 21 Overlay routing enabled



Whether or not overlay routing is enabled when the firewall returns traffic to the GWLB, the GWLB uses the session information in the Geneve header to return the traffic to the originating GWLB endpoint. The traffic egresses the endpoint and is then forwarded to the destination by using the endpoint's routing table.

Return traffic from the destination must be symmetrically forwarded back to the same GWLB endpoint. Because address translation is not being used on the GWLB and VM-Series, the symmetrical traffic forwarding is determined by the destination's routing table and any IGW or NGW entries that were created in the initial flow setup.

Figure 22 Symmetrical traffic forwarding



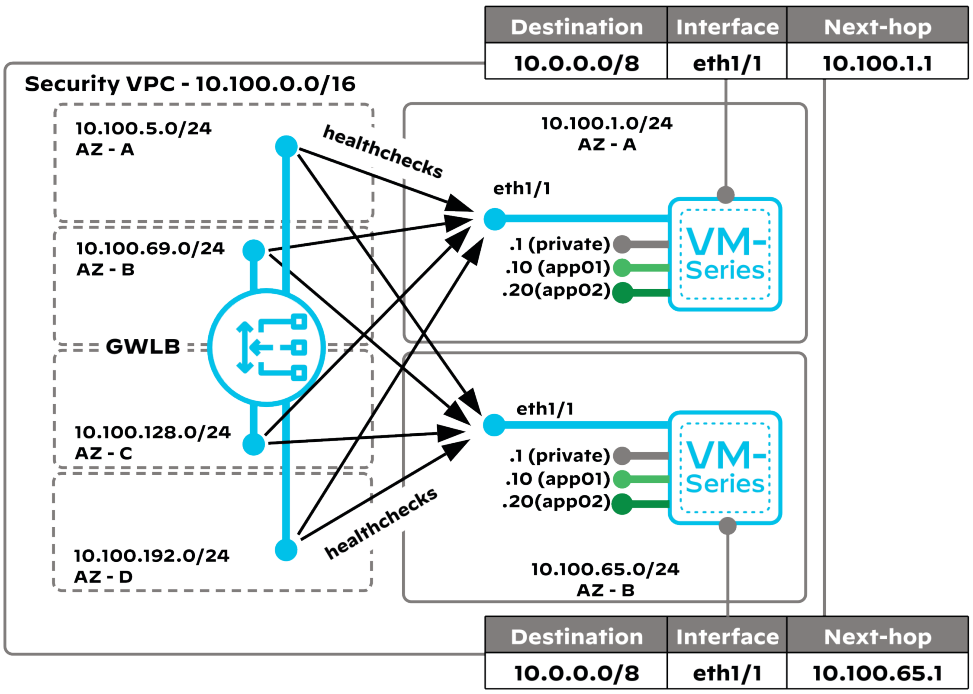
Health Checks

Health checks from the GWLB determine the health of the target firewalls. The GWLB sends health checks to the firewall's primary dataplane interface IP address. You cannot configure the health checks to monitor the full path to the private resources.

To configure the firewall to respond to the health checks, enable an interface management profile on the firewall's dataplane interface that permits access to either the SSH or HTTPS service. You should not configure the management profile on the subinterfaces, because health checks are destined only for the dataplane interface.

Because health checks come from the GWLB IP addresses, the firewall must also have a route to every subnet into which the GWLB is deployed. Additionally, you need to allow the SSH or HTTPS traffic that is sourced from the GWLB subnets in the firewall's security policy. Because the dataplane interface is also used for unmapped GWLB endpoints, do not configure an overly permissive security policy for health checks.

Figure 23 Health checks



PRISMA CLOUD FOR AWS

Prisma Cloud is a cloud infrastructure security solution that provides complete visibility and control over risks within your public-cloud infrastructure. To help ensure that your cloud infrastructure is protected from security threats, this service continuously monitors your cloud environments.

On AWS Marketplace, you can choose one of two Prisma Cloud payment options:

- Pay-as-you-go (with 15-day free trial)
- Annual contract

Within 24 hours of purchase, you'll get access to the Prisma Cloud tenant that Palo Alto Networks and AWS provisioned for you.

Prisma Cloud provides cloud-infrastructure protection across the following areas:

- **Multi-cloud security**—Achieve consistent implementation of security best practices across AWS, Azure, and GCP. Prisma Cloud requires no agents, proxies, software, or hardware for deployment and integrates with a variety of threat intelligence feeds. Prisma Cloud includes pre-packaged policies for securing multiple public-cloud environments.
- **Continuous compliance**—Maintain continuous compliance across CIS, NIST, PCI, FedRAMP, GDPR, ISO, and SOC 2 by monitoring API-connected cloud resources across multiple cloud environments in real time. Prisma Cloud can generate compliance documentation with one-click exportable, fully prepared reports.
- **Cloud forensics**—Go back in time to the moment a resource was first created and see when every change was made chronologically and by whom. Prisma Cloud provides forensic investigation and auditing capabilities of potentially compromised resources across your AWS environment, as well as other public-cloud environments. Historical information extends back to initial creation of each resource, and the detailed change records include who made each change.
- **DevOps and automation**—By setting architecture standards that provide prescribed policy guardrails, you enable secure DevOps without adding friction. This methodology permits agile development teams to maintain their focus on developing and deploying apps that support business requirements.

To analyze and produce concise actionable insights, Prisma Cloud connects to your cloud via APIs and aggregates raw configuration data, user activities, and network traffic.

Prisma Cloud performs a five-stage assessment of your cloud workloads. Contributions from each stage progressively improve the overall security posture for your organization:

- **Discovery**—Prisma Cloud continuously aggregates configuration, user activity, and network traffic data from disparate cloud APIs. It automatically discovers new workloads as soon as they are created.
- **Contextualization**—Prisma Cloud correlates the data and applies machine learning to understand the role and behavior of each cloud workload.
- **Enrichment**—External data sources—such as vulnerability scanners, threat intelligence tools, and SIEMs—further enrich the correlated data to deliver critical insights.
- **Risk assessment**—Prisma Cloud scores each cloud workload for risk, based on the severity of business risks, policy violations, and anomalous behavior. Aggregated risk scores enable you to benchmark and compare risk postures across different departments and across the entire environment.
- **Visualization**—An interactive dependency map shows the entire cloud infrastructure environment, providing context beyond the raw data.

Threat Defense

Prisma Cloud enables you to visualize your entire AWS environment, including every component within the environment. Prisma Cloud dynamically discovers cloud resources and applications by continuously correlating configuration, user activity, and network traffic data. Combining this deep understanding of the AWS environment with data from external sources, such as threat intelligence feeds and vulnerability scanners, enables Prisma Cloud to produce context around risks.

Prisma Cloud includes policies that adhere to industry-standard best practices right out-of-the-box. You can also create custom policies based on your organization's specific needs. Prisma Cloud continuously monitors for violations of these policies by existing resources as well any new resources that are dynamically created. You can easily report on the compliance posture of your AWS environment to auditors.

Prisma Cloud automatically detects user and entity behavior within the AWS infrastructure and management plane. It establishes behavior baselines, and it flags any deviations. It also computes risk scores—similar to credit scores—for every resource, based on the severity of business risks, violations, and anomalies. The risk score helps you to quickly identify the riskiest resources and enables you to quantify your overall security posture.

Prisma Cloud reduces investigation-time from weeks or months to seconds. To quickly pinpoint issues and perform upstream and downstream impact analysis, you can use Prisma Cloud graph analytics. Prisma Cloud provides you with a DVR-like capability to view time-serialized activity for any given resource. You can review the history of changes for a resource and better understand the root cause of an incident, past or present.

Prisma Cloud enables you to quickly respond to an issue based on contextual alerts. Alerts are triggered based on a risk-scoring methodology and provide context on all risk factors associated with a resource. This feature makes it simple to prioritize the most important issues first. When a resource has a high risk score, you can choose to send alerts, orchestrate policy, or perform auto-remediation. Prisma Cloud can also send alerts to Cortex XSOAR and third-party tools such as Slack, Splunk, and ServiceNow so that you can remediate the issue.

Prisma Cloud provides the following visibility, detection, and response capabilities:

- **Host and container security**—Configuration monitoring and vulnerable image detection.
- **Network security**—Real-time network visibility and incident investigations. Suspicious/malicious traffic detection.
- **User and credential protection**—Account and access key compromise detection. Anomalous insider activity detection. Privileged activity monitoring.
- **Configurations and control plane security**—Compliance scanning. Storage, snapshots, and image configuration monitoring. Security group and firewall configuration monitoring. IP address management configuration monitoring.

Continuous Monitoring

The dynamic nature of the cloud creates challenges for risk and compliance professionals tasked with measuring and demonstrating adherence to security and privacy controls. With the Prisma Cloud portal, you can view the collected continuous security-monitoring data collected by Prisma Cloud and verify compliance of your resources to CIS v1.0, CSA CCM v3.0.1, GDPR, HIPAA, ISO 27001:2013, NIST 800.53 R4, PCI DSS v3.2, and SOC2 standards. This capability eliminates the manual component of compliance assessment.

Prisma Cloud provides security and compliance teams with a view into the risks across all of their cloud accounts, services, and regions by automating monitoring, inspection, and assessment of your cloud infrastructure services. With real-time visibility into the security posture of your environment, you can identify issues that do not comply with your organization's required controls and settings and send automated alerts.

Design Models

There are many ways to use the concepts in the previous sections in order to secure application environments in AWS. The design models in this section offer example architectures that secure inbound and outbound traffic flows, traffic between VPCs, and the connection to your on-premises networks.

Each of the design models uses a separate management VPC to centralize management so that a single Panorama deployment can manage VM-Series firewalls deployed across all of your organization's VPCs. Panorama streamlines and consolidates core tasks and capabilities, enabling you to view all your firewall traffic, manage all aspects of device configuration, push global policies, and generate reports on traffic patterns or security incidents. You deploy Panorama in Management-Only mode and securely access it over the public internet. The VM-Series firewalls encrypt and send all firewall logs to Cortex Data Lake over TLS/SSL connections.

The design models presented here differ in how they provide resiliency, scale, and services for the design. The design models in this reference design are:

- **Centralized**—Supports interconnecting a large number of VPCs, with a scalable solution to secure outbound, inbound, and east-west inter-VPC traffic flows using a transit gateway to connect the VPCs.
- **Isolated**—Supports outbound and inbound traffic flows. This design model does not support east-west traffic flows between VPCs. This design is useful for regional security where one or more isolated VPCs need protection.

CHOOSING A DESIGN MODEL

When choosing a design model, consider the following factors:

- **Scale**—Is this deployment an initial move into the cloud and a proof of concept? Will the application load need to scale quickly and modularly? Are there requirements for outbound, inbound, and east-west flows? The Isolated design model provides outbound and inbound security to one or more VPCs but does not provide VPC-to-VPC connectivity or security. The Centralized design model offers the benefits of a highly scalable design for multiple VPCs connecting to a central hub for inbound, outbound, and VPC-to-VPC traffic control and visibility.
- **Segmentation**—Understanding application flows and how to scale and troubleshoot is important to the design. You might need to segment pieces of the application from each other for the best security control. Consider the *Centralized design model* for a scalable design when you need to segment pieces of the application into separate VPCs. When an application VPC is self-contained and does not need to communicate with another VPC, you can apply network security in several ways. In the *Isolated design model*, you centralize the security instances in a dedicated security VPC while providing security services for one or more isolated VPCs.

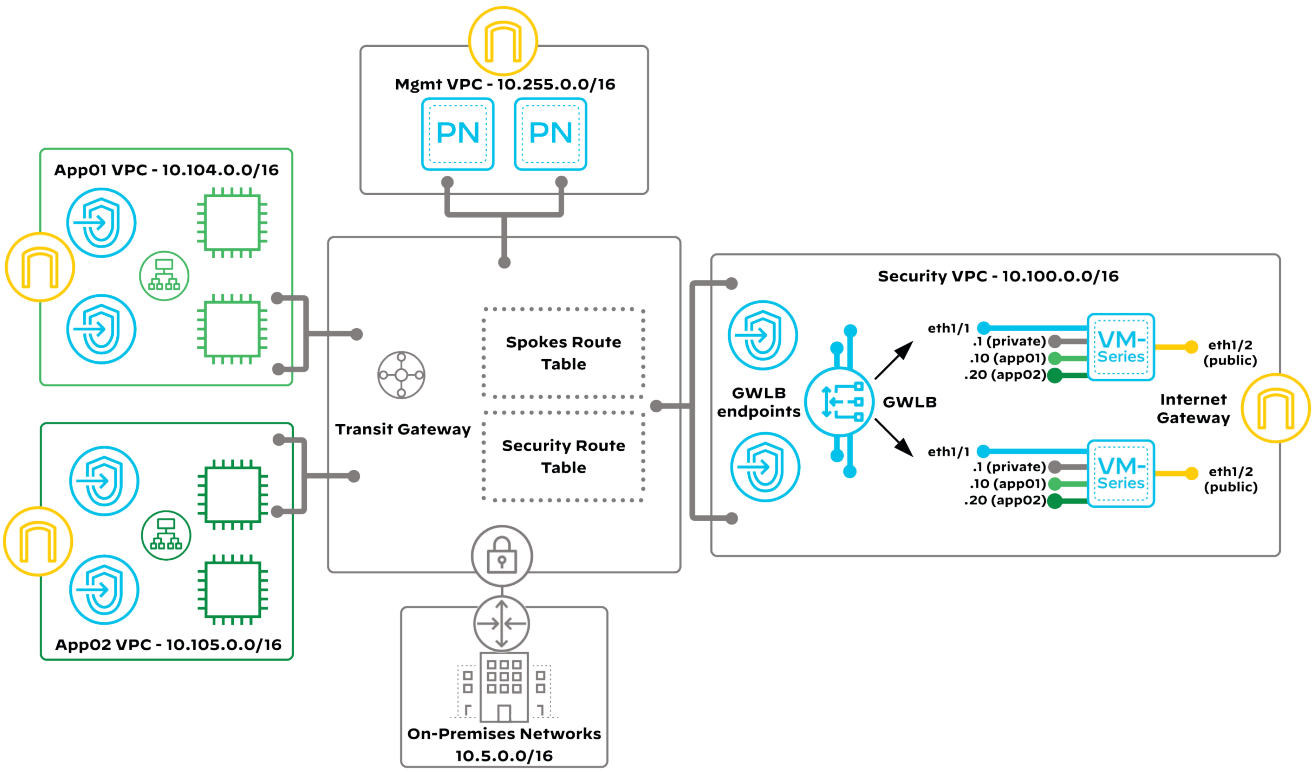
CENTRALIZED DESIGN MODEL

In the Centralized design model, you segment application resources across multiple VPCs that connect in a hub-and-spoke topology. The hub of the topology, or transit gateway, is the central point of connectivity between VPCs and on-premises resources attached through a VPN or AWS Direct Connect.

This model has a dedicated VPC for security services where you deploy VM-Series firewalls for traffic inspection and control. The security VPC does not contain any application resources. The security VPC centralizes resources that multiple workloads can share.

The TGW ensures that all spoke-to-spoke and spoke-to-on-premises traffic transits the VM-series firewalls.

Figure 24 Centralized design



Transit Gateway Design

In this design, there are two or more spoke VPCs for application resources. For resiliency, you should deploy the resources in the spoke VPCs across availability zones. You can dedicate each VPC to an application, or if you need inspection and control between pieces of an application, each VPC can contain the application resources that share a common level of security control. You use separate VPCs for segmentation because traffic between resources in the same VPC cannot be redirected to a firewall. The traffic always flows directly.

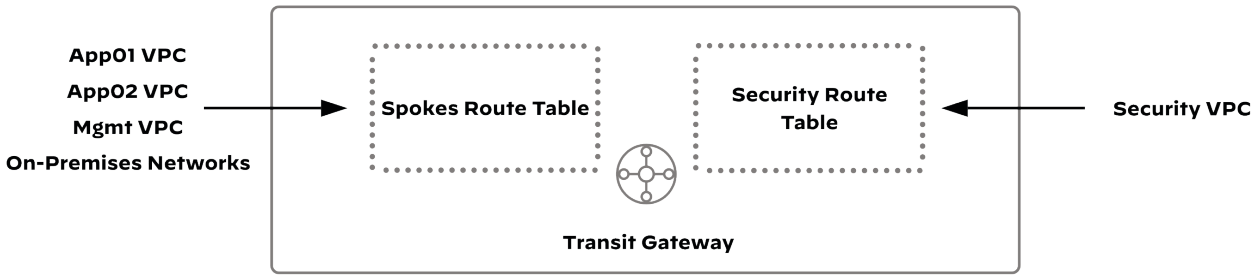
Because this design uses Panorama to manage the VM-Series firewalls, and because you might have multiple sets of firewalls to manage, deploy Panorama in a VPC dedicated to management. Use another VPC to deploy the VM-Series firewalls. Like with the spoke VPCs, deploy Panorama and the VM-Series firewalls across availability zones for resiliency.

The TGW has VPC attachments in the availability zones of each spoke VPC, the management VPC, and the security VPC.

TGW route tables behave like route domains. You create isolated networks, allowing you to steer and control traffic flow between VPCs and on-premises connections by deploying multiple route tables on the TGW and associating the attachments to them. This design uses two TGW route tables: security and spokes.

You set which TGW route table an attachment uses by associating the attachment to a route table. An attachment can only be associated with one TGW route table. However, each TGW route table can associate with multiple attachments. Associate the VPN attachment for the on-premises networks, as well as the VPC attachments for the spokes and management VPCs, to the spokes route table. Associate the VPC attachment for the security VPC to the security route table.

Figure 25 Attachments and associations



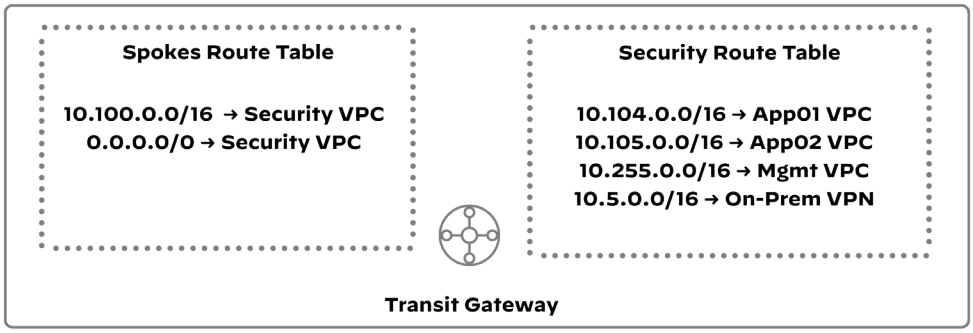
The spokes route table on the TGW has the security VPC routes propagated to it and a static default route pointing to the security VPC. The spokes route table does not have more specific routes to other VPCs or on-premises networks to ensure that spoke-to-spoke communication can occur only through the VM-Series firewalls in the security VPC.

The security route table on the TGW allows the VM-Series firewalls to reach all VPCs and on-premises networks. Propagate routes from every attachment to the security route table. You can use static routes in the TGW route table for on-premises connections, or you can propagate routes over BGP from a VPN attachment into a TGW route table. Routes propagated with BGP support ECMP.

Note

Even though the TGW route tables can support up to 10,000 routes, the BGP prefix limitation is 100 prefixes per virtual gateway.

Figure 26 TGW Route Tables

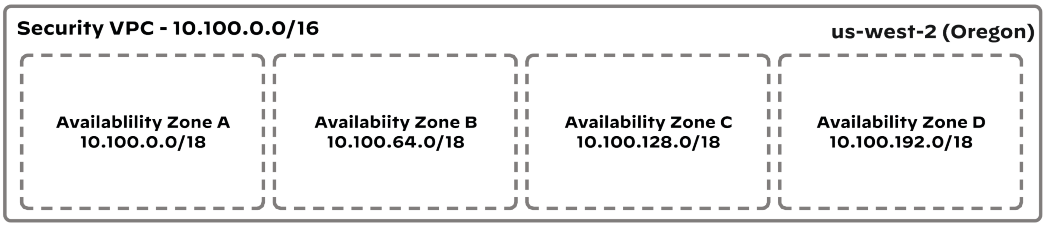


Security VPC

This model deploys a VPC dedicated to security. In the security VPC, you deploy the VM-Series firewalls in separate availability zones and deploy a GWLB to distribute traffic to the firewalls.

When deploying the security VPC, configure it with a non-overlapping IP address block that is appropriately sized for your organization's needs. For resiliency, use multiple availability zones and consider breaking your subnet ranges into availability zone-based blocks that you can summarize in security groups and routing tables.

Figure 27 VPC availability zone-based IP blocks



To supply inbound and outbound internet access, you must deploy an IGW into the VPC. The IGW performs network address translation of the VM-Series management interface's private IP address to its associated public IP address and allows outbound traffic from the firewall to reach the internet.

VM-Series Firewalls

Deploy two or more VM-Series firewalls in separate availability zones.

Although the VM-Series firewall supports multiple interface deployment configurations such as virtual wire, Layer 2, and tap mode, VM-Series firewall interfaces on AWS are always Layer 3 interfaces because of AWS networking requirements. In AWS, you should always configure VM-Series firewall interfaces to obtain their IP address through DHCP.

Because this design model uses overlay routing, the VM-Series firewall requires three Ethernet interfaces: a management interface, a private dataplane interface for traffic from the GWLB, and a public dataplane interface for outbound traffic.

When you deploy a VM-Series instance from the AWS Marketplace, it has a single interface by default. You have to create the additional interfaces and associate them with the VM-Series instance. You also need to create two Elastic IP addresses. Assign one to the management interface so you can manage the firewall and the other to the public interface so the VM-Series firewall can support outbound traffic flows.

Although the dataplane interfaces both get their IP addresses from DHCP, only the public interface should accept the default route.

Attach each firewall interface to separate subnets. The management and public subnets have separate route tables as follows:

- The management route table has all the management subnets assigned to it, a default route to the IGW for internet access, and a route to the TGW for access to Panorama.
- The public route table has all the public subnets assigned to it and a default route to the IGW for internet access.

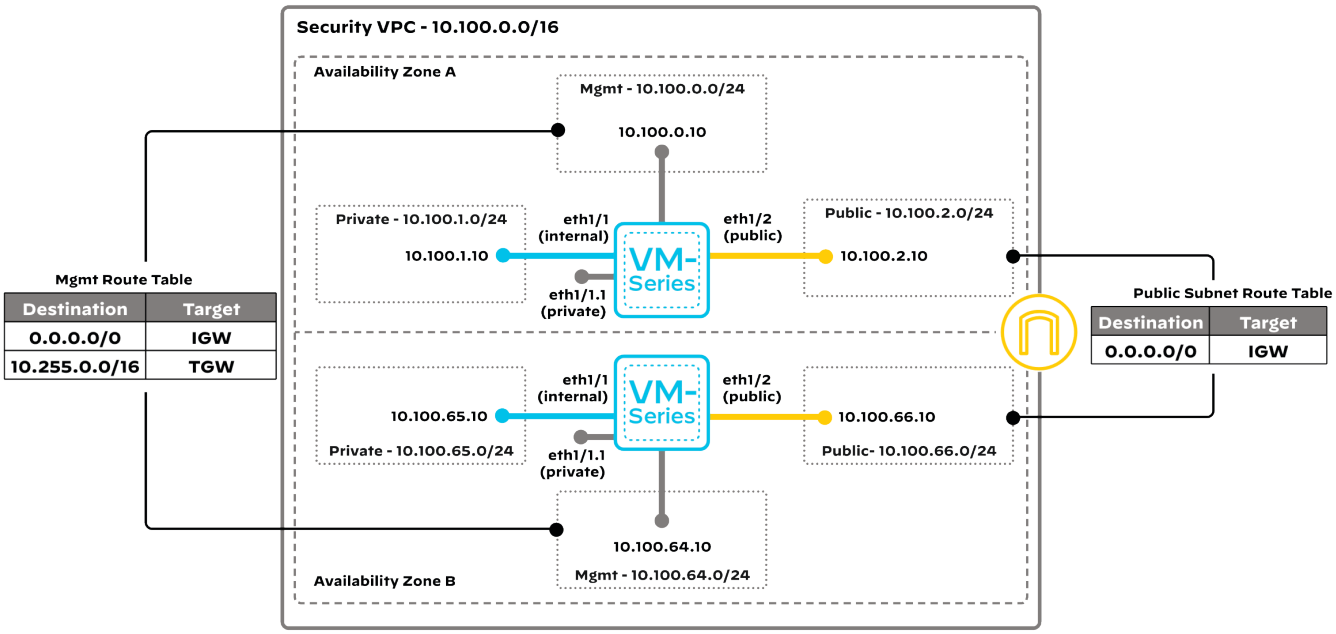
You do not need to modify the default routing of the subnets dedicated to the private dataplane interface. However, you should configure a security group on the firewall's private dataplane interface that allows health checks and UDP traffic destined to port 6081 from all the GWLB subnets. The security group should deny all other traffic.

The private dataplane interface should have at least one subinterface. The GWLB endpoints should map to subinterfaces instead of the dataplane interface. Mapping all endpoints to a subinterface allows you to have a restrictive security policy on the dataplane interface that allows health checks only from the GWLB subnets. This security policy provides a layer of protection if an endpoint is misconfigured or newly added to the GWLB.

Note

Although the firewall does not use the subinterfaces for traffic forwarding, you must configure them to receive a DHCP address and to have a unique interface name. AWS does not assign the subinterface an IP address, which is expected and does not affect traffic forwarding with a GWLB.

Figure 28 VM-Series firewall interfaces and AWS route tables



You must also enable Geneve inspection and overlay routing on the firewall.

Note

With Geneve inspection enabled, the firewall is able to look inside the Geneve encapsulation to the actual traffic. If you don't enable Geneve inspection, the firewall sees traffic destined to the dataplane interface IP address with a destination port of 6081.

You can automate the VM-Series configuration management by using Panorama in the management VPC to provide a centralized, secure, scalable, and automated architecture. Bootstrapping speeds up the process of configuring and licensing the firewall and making it operational on the network. This process allows the deployment of the firewall with only a basic configuration so that it can connect to Panorama and obtain the complete working configuration.

Although you can enable aws-gwlb-inspect and map the GWLB endpoints to subinterfaces after deployment, it must be done directly on the firewall and cannot be configured using Panorama. Because a manual per-device configuration can be challenging as you scale out devices, you should configure Geneve inspection and overlay routing during bootstrap instead.

GWLB

A GWLB in the security VPC transparently distributes traffic across the VM-Series firewalls. You should deploy the GWLB in all of the security VPC's availability zones, with a single endpoint service.

The GWLB targets the VM-Series firewall instances by using their primary dataplane interface IP address and uses HTTPS health checks to monitor their availability.

TGW Attachment

The security VPC attaches to the TGW through a VPC attachment. The VPC attachments terminate into the security VPC in dedicated subnets, one per availability zone.

You must enable appliance mode on the attachments in the security VPC in order to ensure that traffic routes through the same attachment zone even when the source and destination of the traffic are in different zones. If appliance mode is not enabled, it is possible to have asymmetric traffic flowing through different firewalls, which the firewalls drop.

Outbound Traffic

This design uses overlay routing for outbound security on the VM-Series firewalls. Outbound traffic from instances in the spoke VPCs traverses the TGW and egresses the AWS environment through the VM-Series firewalls.

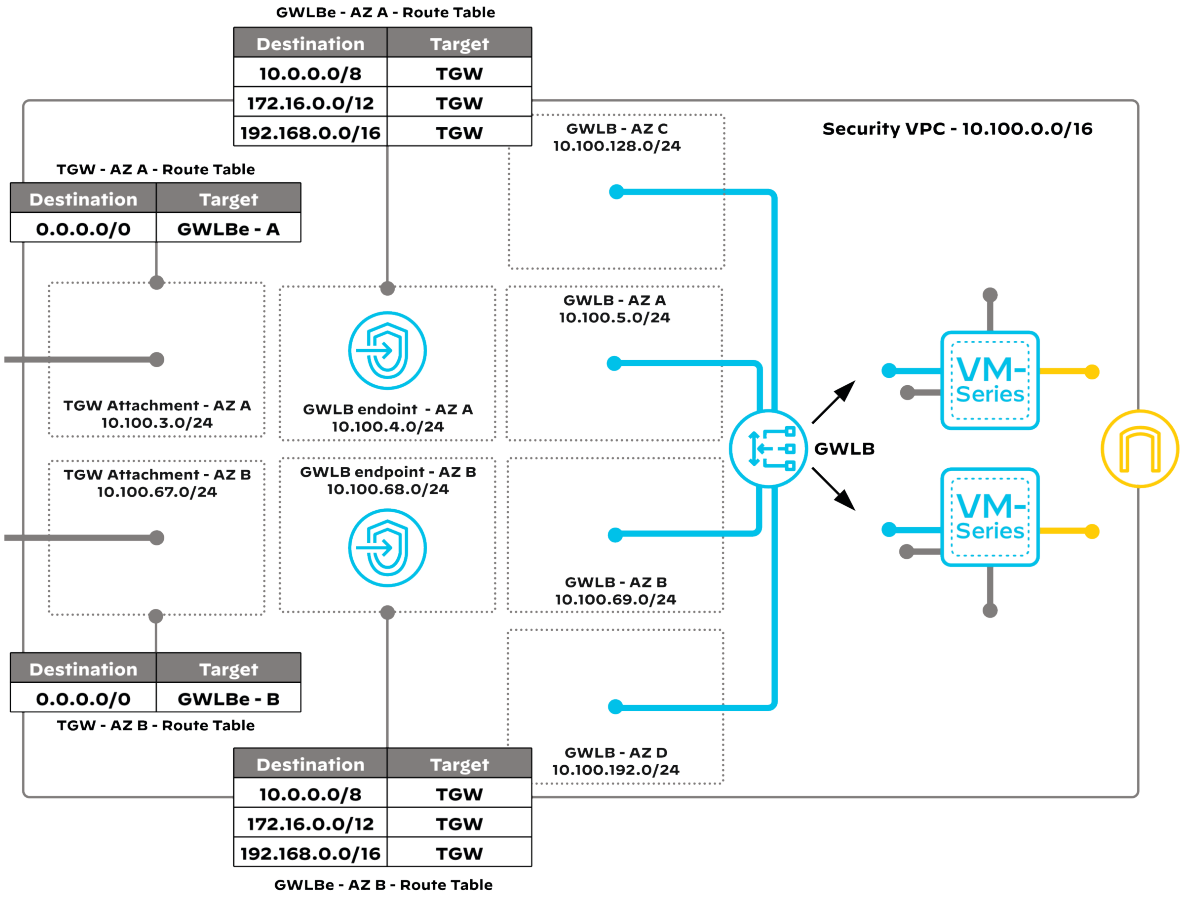
Default routes in the spoke VPCs direct outbound traffic to the TGW. To direct traffic to the security VPC, the TGW uses the spokes route table and the static default route pointing to the security VPC.

To support outbound traffic flows, the security VPC has GWLB endpoints in each of the availability zones in which the firewalls are deployed. The GWLB endpoints are associated with dedicated subnets, which allows them to have a unique routing table. The route tables in the security VPC for outbound traffic are as follows:

- A routing table for each of the TGW attachment subnets directs all traffic from the TGW to the GWLB endpoint in the attachment's availability zone.
- A routing table the GWLB endpoint subnets directs return traffic to the spoke VPCs address space to the TGW. In most cases, routes to the RFC 1918 address space is the simplest method for a configuration that does not need to be modified as you add spoke VPCs to the TGW.

You do not need to modify the default routing of the subnets dedicated to the GWLB. By default, they can reach all the IP addresses within the security VPC.

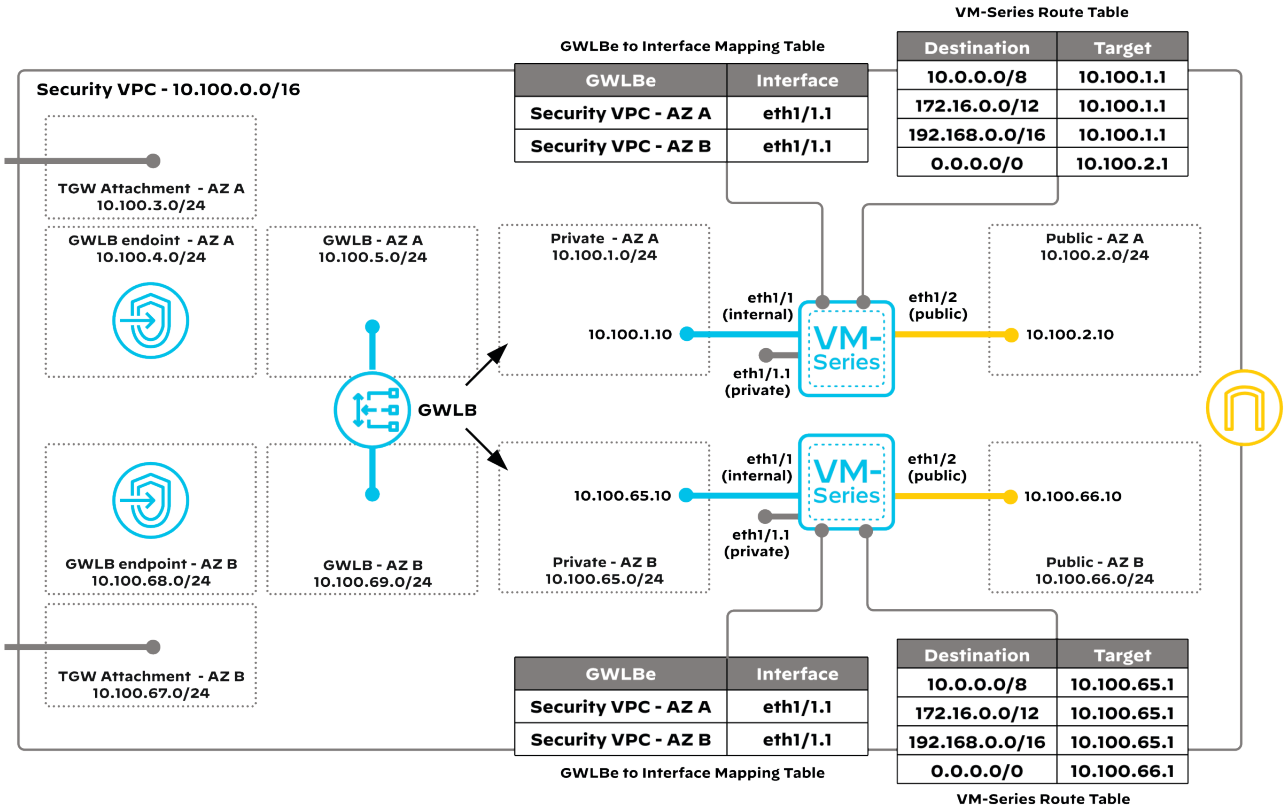
Figure 29 GWLB and route table details



On the VM-Series firewalls, map the GWLB endpoints to a subinterface on the private dataplane interface. The subinterface must have a unique security zone and interface name. The security zone is used in policy, but the subinterface itself is not used in traffic forwarding.

The firewall also has routes to the spoke VPC address space—or for simplicity, routes to all the RFC 1918 address space—that point to the private dataplane interface's default gateway.

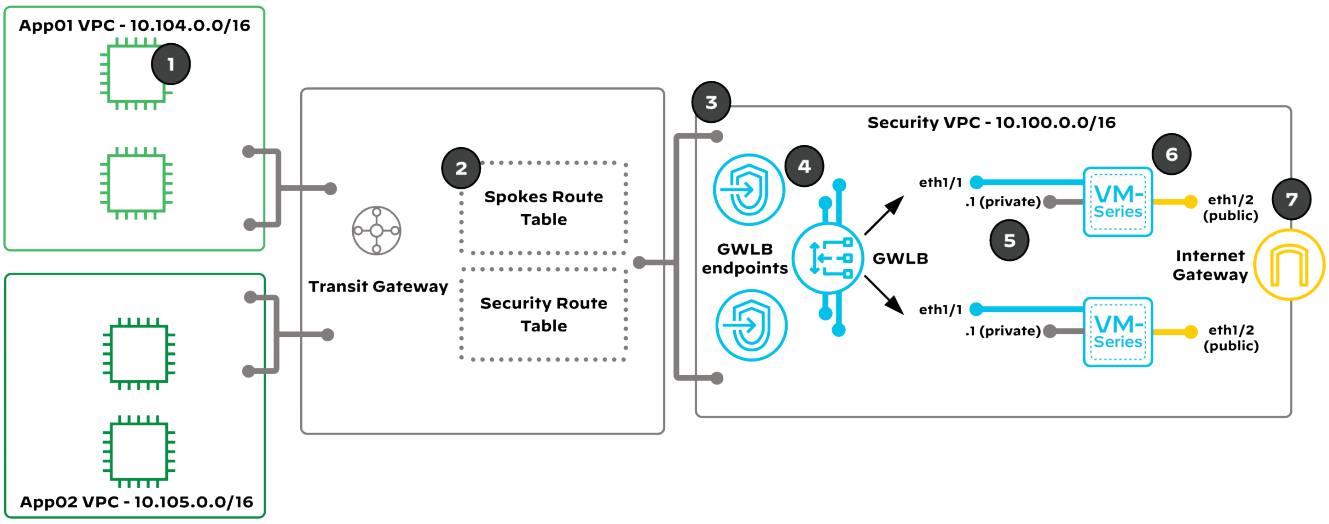
Figure 30 Firewall endpoint mapping and routing



Outbound Traffic Flow

1. In the application VPC, the instance in the application VPC sends traffic to a service on the internet. The route table applied to the instance's subnet directs traffic to the TGW attachment in its availability zone.
2. On the TGW, the application VPC is associated with the spokes route table. The TGW uses the static route in the that route table to forward the traffic to the security VPC.
3. In the security VPC, the route table applied to the subnet that contains the TGW attachment has a default route that forwards traffic to the GWLB endpoint in the attachment's availability zone.
4. In the security VPC, the GWLB endpoint forwards the traffic to the GWLB which chooses a VM-Series firewall from its target group. Because cross-zone load balancing is enabled, it could be any available firewall.
5. The firewall receives the traffic from the GWLB and associates it with the private zone, which is configured on the subinterface that is mapped to the security VPC GWLB endpoints.
6. The firewall applies the decryption, NAT, and security policies before forwarding it out its public dataplane interface based on the default route learned from DHCP on that interface. The outbound traffic's source IP address is translated to the IP address of the firewall's public interface.
7. In the security VPC, the route table applied to the subnet that contains the firewall's public interface has a default route that forwards traffic to the IGW. The IGW translates the source IP address of the outbound traffic to the EIP associated with the firewall's public interface and forwards the traffic to the internet.

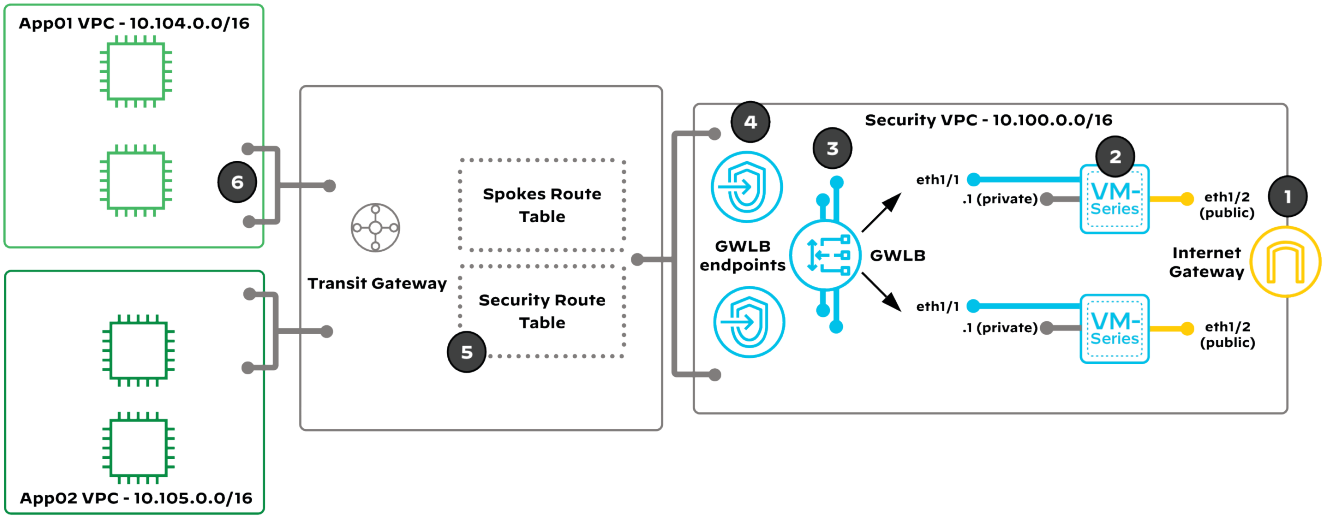
Figure 31 Outbound traffic flow



Return Traffic Flow

1. In the security VPC, the IGW translates the destination IP address of the return traffic to the firewall's public interface's IP address.
2. The firewall receives the traffic and associates the return traffic with an active session on the firewall. The firewall's route table directs the traffic destined to the application VPC out the private interface. Because this is the same interface that received the traffic from the GWLB, the firewall uses the session information, including the TLVs associated with the Geneve encapsulation, to return the traffic to the GWLB.
3. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic.
4. In the security VPC. The route table associated with the subnet that contains the GWLB endpoint has a route that forwards traffic destined to the application VPCs address space to the TGW attachment in its availability zone.
5. On the TGW, the security VPC is associated with the security route table. The TGW uses that route table to forward traffic to the application VPC.
6. In the application VPC, the TGW attachment forwards the traffic to the instance.

Figure 32 Return traffic flow



Security Policy

You use VM-Series firewall security policies to limit what applications and resources the private instances can reach. In most designs, the VM-Series firewall does not need to translate the destination IP address.

The VM-Series firewall security policy allows appropriate application traffic from private instances to the internet. Using the zones of the private subinterface and the public dataplane interface to define the policy. You should implement the outbound security policy by using positive security policies (*allow-listing*). Security profiles prevent known malware and vulnerabilities from entering the network in return traffic allowed by the security policy. URL filtering, data loss prevention, file blocking, and data filtering protect against data exfiltration.

Inbound Traffic

Inbound traffic originates outside the VPC and is destined to applications or services hosted within your VPCs, such as web servers. You have two options for implementing inbound security resiliency:

- **Combined**—You can use the VM-Series and GWLB in the security VPC, with distributed GWLB endpoints in the application VPCs. Unlike with outbound traffic, this design option does not use the transit gateway for traffic forwarding between the security VPC and the application VPCs.
- **Centralized**—You can use an AWS load-balancer sandwich design. This design uses a resilient, public-facing load balancer in the security VPC and a second resilient load balancer on the private side of the firewalls in the application VPCs. This design option uses the transit gateway for traffic forwarding between the security VPC and application VPCs.

Note

The combined option uses the GWLB to apply security for inbound traffic transparently and does not require address translation. The centralized option does not use the GWLB and requires that the firewall translate the traffic source IP address to ensure traffic returns to the correct firewall in the security VPC.

Combined Option

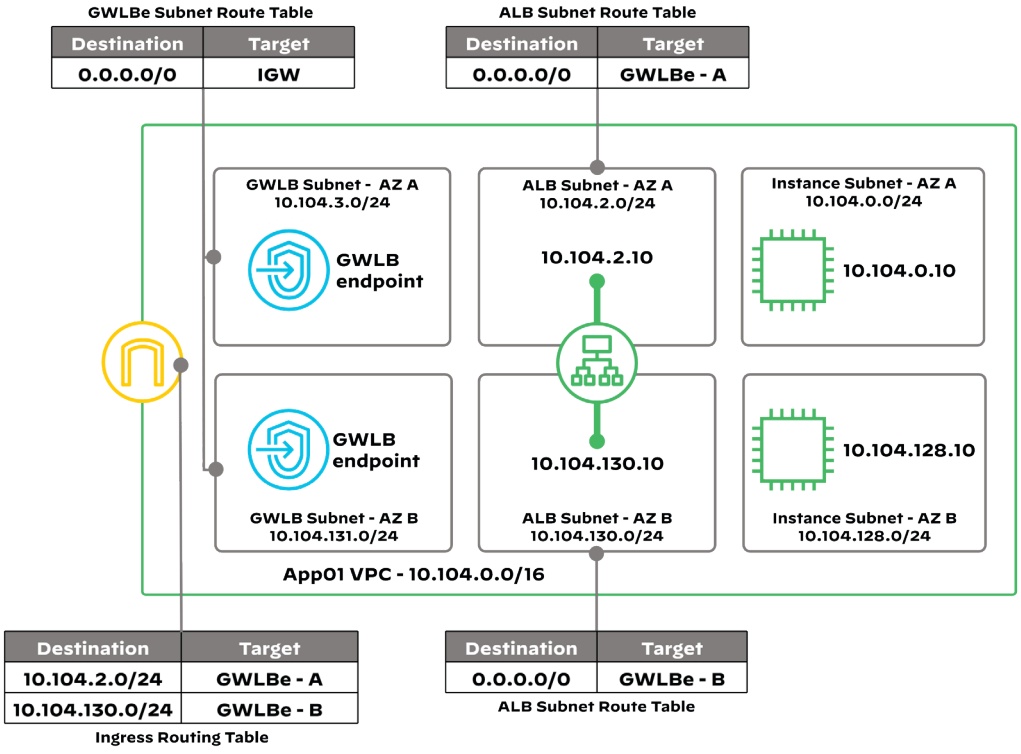
This design option adds distributed inbound security to the existing VM-Series firewalls in the security VPC and builds upon the previously discussed configuration in order to support outbound traffic flows. If you have regulatory or other compliance concerns, a separate set of firewalls dedicated to inbound traffic flows might be more appropriate.

This design uses the GWLB and VM-Series firewalls in the security VPC, with GWLB endpoints in the application VPCs for the transparent inspection of inbound traffic.

To support inbound security with this design option, modify each application VPC that requires inbound security as follows:

- To supply inbound access, deploy an IGW to the application VPC.
- Add GWLB endpoints in each of the availability zones and associate them with the GWLB in the security VPC.
- Add a gateway route table that directs traffic to the subnets in an availability zone to the GWLB endpoint in that availability zone.
- Add an internet-facing load balancer with the application instances in the VPC as the targets.
- Add a route table for the GWLB endpoint subnets. The route table must direct traffic that is destined to the internet to the IGW.
- Add a route table for each of the subnets that contain the load balancer. The route tables must direct traffic that is destined to the internet to the GWLB endpoint in its availability zone.

Figure 33 Combined inbound security—application VPC



To support inbound security with this design option, modify the security VPC as follows:

- Add a subinterface on the private dataplane interface for each application VPC.
- Map all GWLB endpoints in the application's VPC to the subinterface.

Note

Because inbound traffic flows with the combined design option are intra-zone, you should modify the default intra-zone policy to block traffic.

In this design, the VM-Series firewalls inspect and transparently secure traffic before the internet-facing load balancer receives it. Because this design option uses GWLB for resiliency, the firewalls do not modify the traffic source or destination IP address.

Because the firewall is in the first device in the traffic path, if you want to filter traffic before the firewall receives it, you must use a network ACL on the GWLB endpoint subnets. If you are using an ALB, the firewall decrypts and re-encrypts the traffic before the ALB sees it.

This design option supports one or more application VPCs. Each application VPC that requires inbound security requires an IGW and locally deployed GWLB endpoints. The VM-Series firewalls in the security VPC inspect and secure the traffic, relying on the AWS PrivateLink connection between the GWLB endpoint and the GWLB to transmit traffic directly between the application VPC and the security VPC.

The VM-Series firewall security policy allows application traffic to the internet-facing application load balancer. VM-Series firewall security profiles prevent known malware and vulnerabilities from entering the network in traffic allowed by the security policy.

Inbound Traffic Flow

1. The URL request from the end-user is directed toward **example.lb.aws.com**. This request is sent to a DNS. DNS returns an IP address for each of the load balancer's enabled availability zones. The client OS picks an IP address and sends the traffic. In the application VPC, the IGW translates the destination public IP address to the IP address of the load balancer.
2. In the application VPC, the gateway route table associated with the IGW directs the traffic to the GWLB endpoint in the destination's availability zone. The GWLB endpoint forwards the traffic to the GWLB in the security VPC.
3. In the security VPC, the GWLB chooses a VM-Series firewall from its target group. Because cross-zone load balancing is enabled, it could be any available firewall.
4. The firewall receives the traffic from the GWLB on its private dataplane interface and associates it with the application's security zone, which is configured on the subinterface that is mapped to the GWLB endpoints in the application VPC.

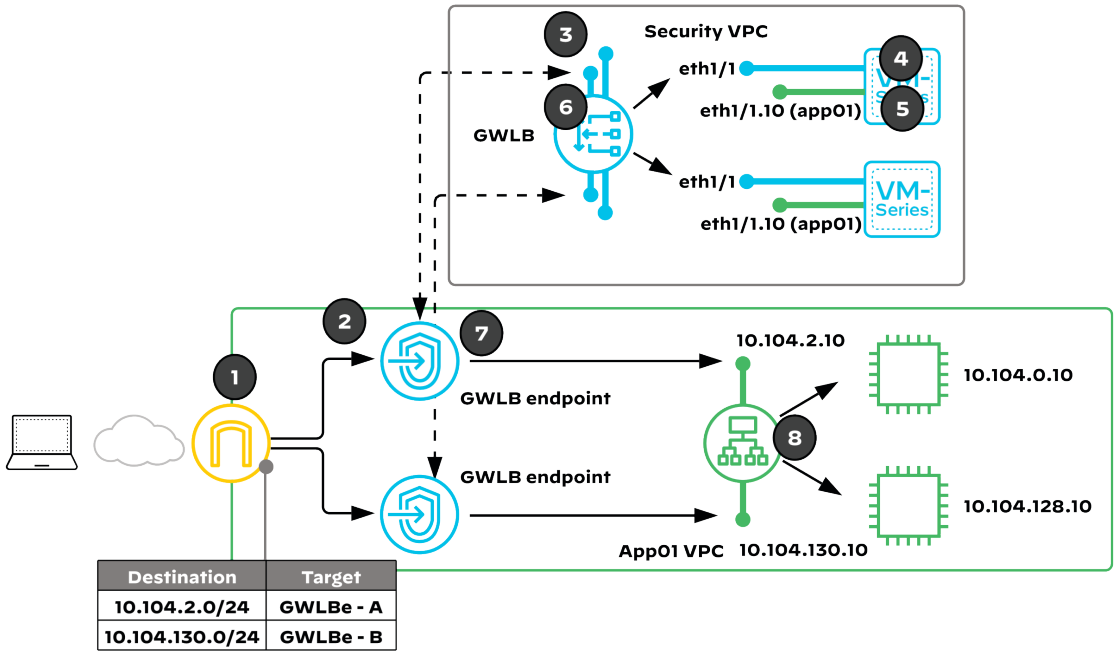
5. The firewall's route table directs the traffic destined to the application VPC out the private interface. The firewall applies security policies before forwarding. The security policy is an intra-zone policy. Because this is the same interface that received the traffic from the GWLB, the firewall uses the session information, including the TLVs associated with the Geneve encapsulation, to return the traffic to the GWLB.

6. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint in the application VPC that received the original traffic.

7. In the application VPC, the GWLB endpoint forwards the traffic to the internet-facing application load balancer.

8. In the application VPC, the internet-facing application load balancer receives the traffic and picks a target from the target group. The targets are the application instances. The application load balancer translates the packet's destination address to the selected instance interface IP address and translates the source IP address to the private IP address of the load balancer so that traffic returns to it on the return flow.

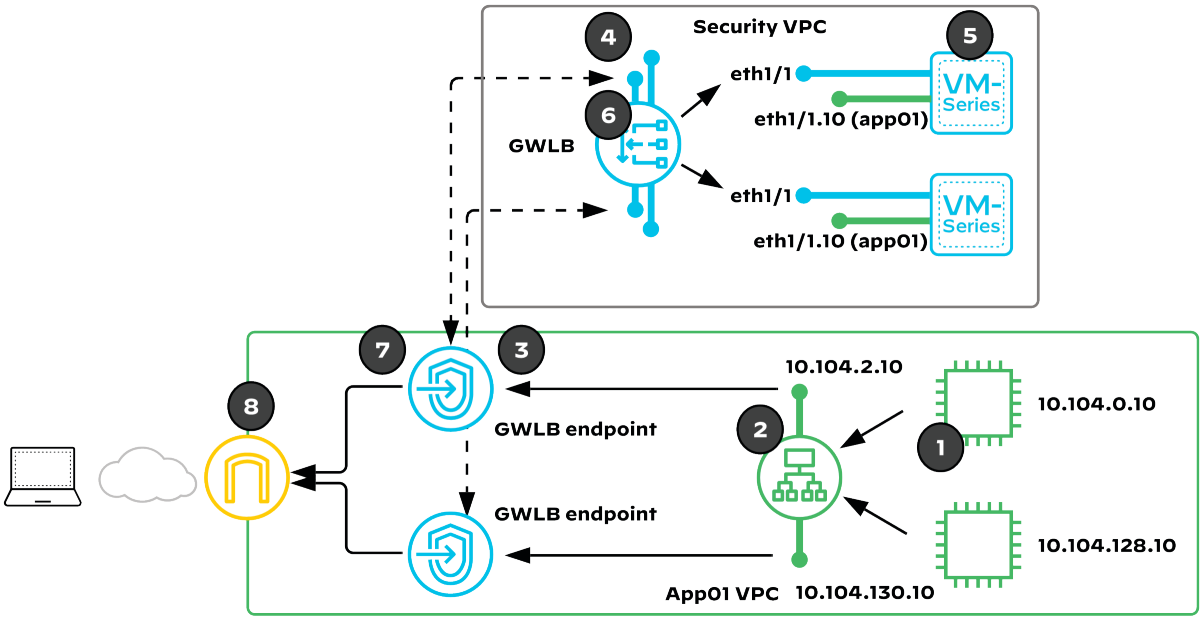
Figure 34 Combined inbound security—inbound traffic flow



Return Traffic Flow

1. In the application VPC, the instance sends return traffic to the load-balancer.
2. In the application VPC, the load balancer modifies the source and destination IP addresses. The route table applied to the subnet that contains the load balancer has a default route that forwards traffic to the GWLB endpoint in its availability zone.
3. The GWLB endpoint forwards the traffic to the GWLB in the security VPC.
4. In the security VPC, the GWLB chooses the same VM-Series firewall from its target group that was used for the inbound traffic flow.
5. The firewall receives the traffic from the GWLB on its private dataplane interface. The firewall receives the traffic and associates the return traffic to an active session on the firewall. The firewall uses the session information, including the TLVs associated with the Geneve encapsulation to return the traffic to the GWLB.
6. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic.
7. In the application VPC, the GWLB endpoint forwards the traffic to the IGW.
8. The IGW translates the source IP address of the outbound traffic to the EIP associated with the load balancer and forwards the traffic to the internet.

Figure 35 Combined inbound security—return traffic



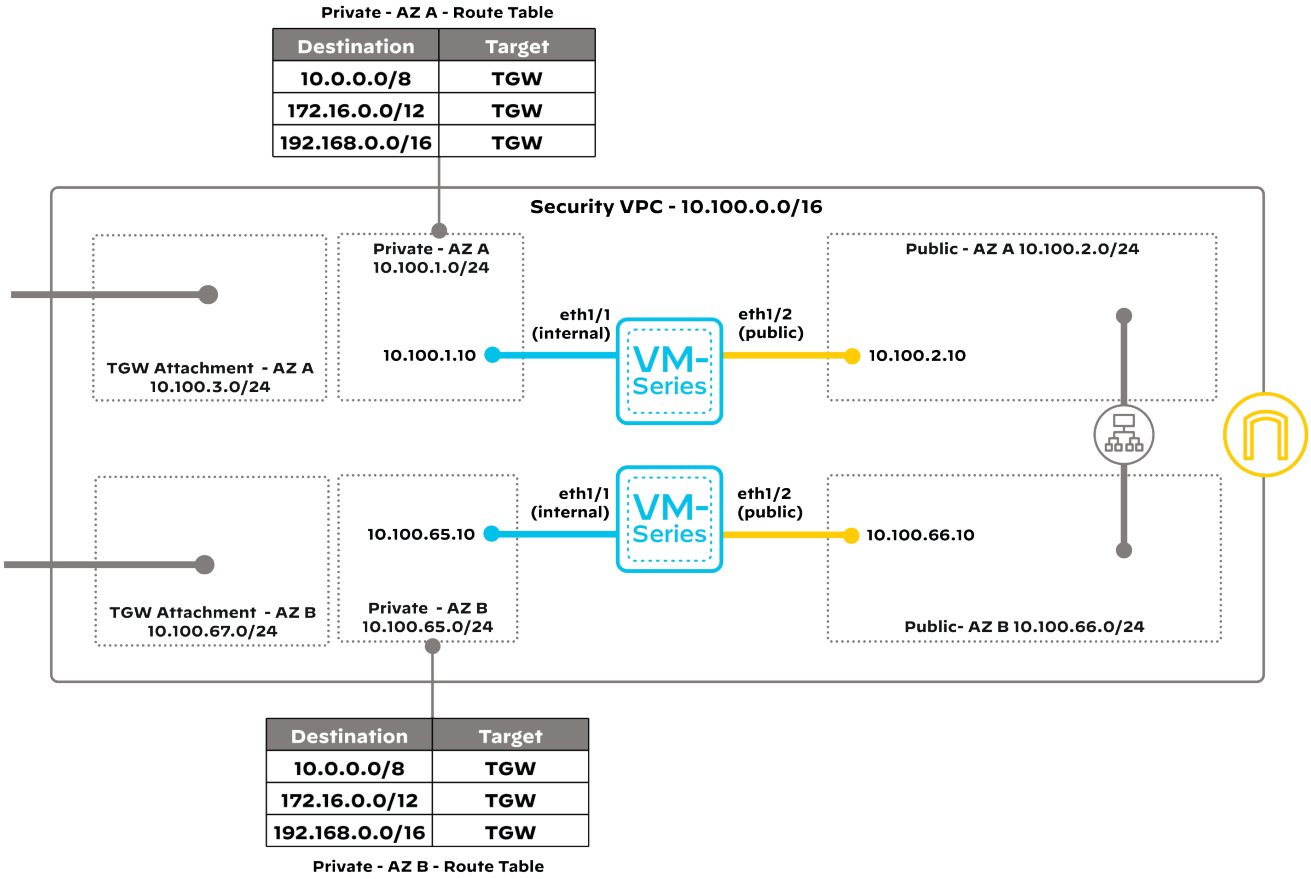
Centralized Option

This design option adds centralized inbound security to the existing VM-Series firewalls in the security VPC and builds upon the previously discussed configuration to support outbound traffic flows. A separate set of firewalls dedicated to inbound traffic flows might be more appropriate if you have regulatory or other compliance concerns. This design option does not use the GWLB for inbound traffic. Instead, it relies on a public ALB/NLB for resiliency and NAT to ensure traffic returns to the correct firewall.

To support centralized inbound security, modify the security VPC as follows:

- Add an internet-facing application or network load balancer.
- Add a route table for the subnets that contain firewall private interfaces. The route table must direct traffic that is destined to the spoke VPCs address space to the TGW. In most cases, routes to the RFC 1918 address space are the most straightforward method for a configuration, because you should not need to modify it as you add spoke VPCs to the TGW.
- Modify the security group on the firewall's private dataplane interface. Adjust the security group to allow return traffic from the application VPCs and health checks from the GWLB subnets. Use an ACL to deny UDP traffic on port 6081 from any network other than the GWLB subnets in the VPC.

Figure 36 Centralized inbound security—ALB and route tables



For inbound traffic, the Application Load Balancer terminates incoming connections to its listener and initiates corresponding new connections to the VM-Series firewalls in the target group. The ALB is associated with the availability zones that contain VM-Series firewalls. If you configure the ALB for multiple web applications behind the same VM-Series firewalls, you must define unique target groups for each application. Each target group contains the same VM-Series firewalls but has unique TCP ports assigned.

AWS sources all new connections from the Application Load Balancer interfaces in the public subnets. The destination IP address is the private IP address of the VM-Series firewall's public interface. Health checks monitor back-end availability on all specified HTTP and HTTPS ports.

Destination IP address translation rules on the VM-Series firewalls map incoming traffic from the ALB listener to the private instance or internal load balancer. The VM-Series firewall also applies a source IP address translation to inbound traffic. The firewall translates the source IP address to the IP address of the private interface of the firewall, ensuring return traffic flows symmetrically.

The VM-Series firewall security policy allows HTTP and HTTPS application traffic from the load balancer to the private instances. VM-Series firewall security profiles prevent known malware and vulnerabilities from entering the network in traffic allowed by the security policy. If you want to support the use of HTTP and HTTPS back ends on ports other than 80 or 443, you should configure the security policy rules' services to include the specific service ports in use instead of *application-default*.

Traffic routing services such as domain name servers and firewall next-hop configurations use FQDN to resolve load-balancer IP addresses versus hard-coded IP addresses. Using FQDN allows the load balancers to be dynamic and scale up or down in size and remain resilient; one availability zone might go down, and the other can continue feeding sessions to the application instances.

Inbound Traffic Flow

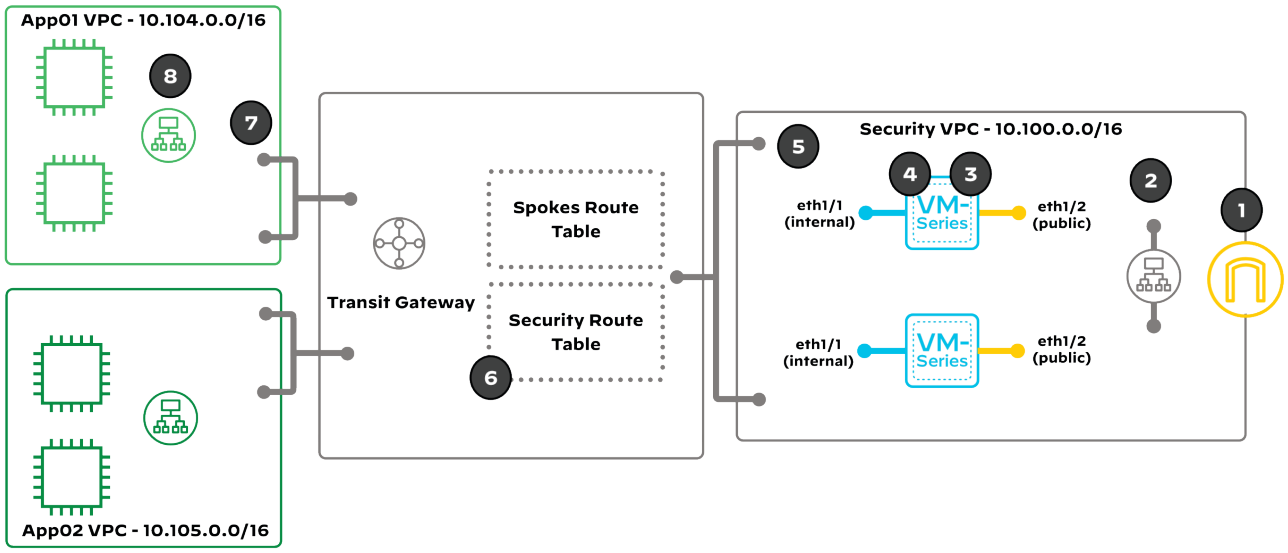
1. The URL request from the end-user is directed toward **example.lb.aws.com**. This request is sent to a DNS. DNS returns an IP address for each of the load balancer's enabled availability zones. The client OS picks an IP address and sends the traffic. The IGW translates the destination public IP address to the IP address of the load balancer.

2. In the security VPC, the internet-facing load balancer receives the traffic and picks a target from the target group. The targets are the private IP addresses for the public-facing interface on each of the VM-Series firewalls. The load balancer translates the packet's destination address to the selected VM-Series firewall's public interface IP address and translates the source IP address to the private IP address of the load balancer so that traffic returns to it on the return flow.

3. The firewall translates the IP addresses on the incoming traffic. The firewall changes the source IP address to the IP address of the firewall's private interface so that the return traffic from the instance travels back through the same firewall in order to maintain state and translation tables. The firewall changes the destination IP address to the IP address of the internal load balancer. The firewall learns the IP addresses for the internal load balancer by sending a DNS request for the FQDN assigned to the load balancers. This returns multiple IP addresses, one for each of the subnets or availability zones into which you deployed the load balancer.

4. The firewall's route table directs the traffic destined to the application VPC out the private interface.
5. In the security VPC, the route table applied to the subnet that contains the firewall's private interface has a route that forwards traffic destined to the application VPCs address space to the TGW attachment in its availability zone.
6. On the TGW, the security VPC is associated with the security route table. The TGW uses that route table to forward traffic to the application VPC.
7. In the application VPC, the TGW attachment forwards the traffic to the internal load balancer.
8. The internal load balancers have instances in both availability zones and do a round-robin load balancing to the active instances in the target list, translating the source IP address to its IP address.

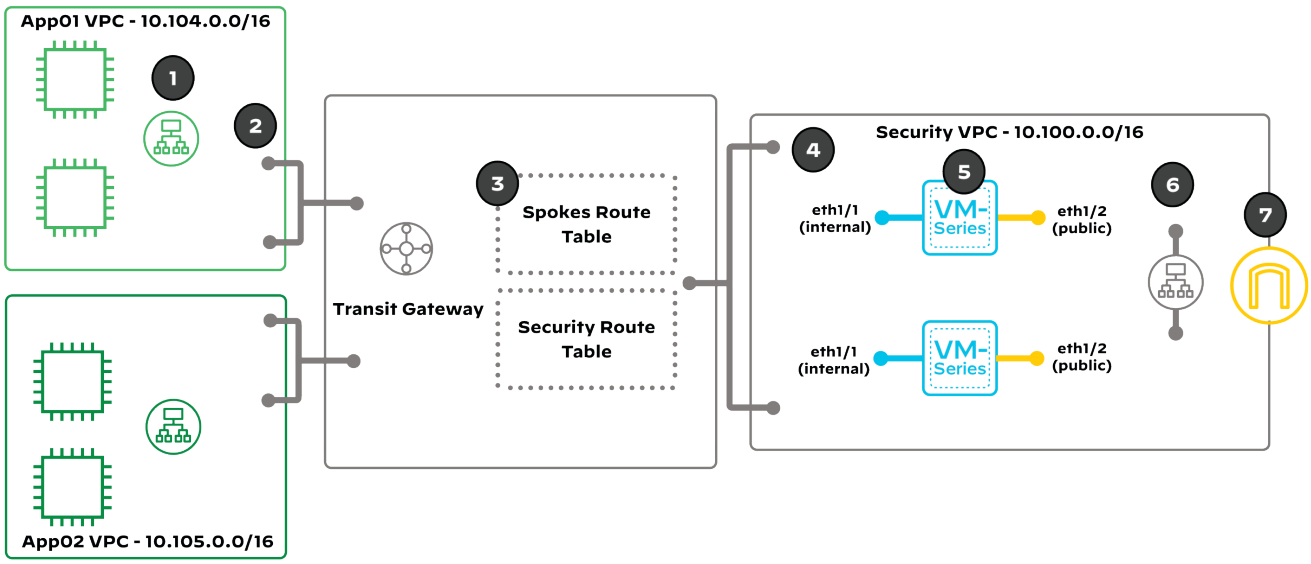
Figure 37 Centralized inbound security—inbound traffic flow



Return Traffic Flow

1. In the application VPC, the instance in the application VPC sends return traffic to the load balancer.
2. In the application VPC, the load balancer sends the return traffic to the firewall. The route table applied to the load balancer's subnet directs the traffic to the TGW attachment in its availability zone.
3. On the TGW, the application VPC is associated with the spokes route table. The TGW uses the static route in the route table to forward the traffic to the security VPC.
4. In the security VPC, the route table applied to the subnet that contains the TGW attachment uses local routing to direct traffic to the VM-Series firewall interface.
5. The firewall receives the traffic and associates the return traffic to an active session on the firewall. The firewall translates the source IP address to the firewall's public interface and destination IP address to the IP address of the internet-facing load balancer.
6. In the security VPC, the route table applied to the subnet that contains the load balancer has a default route that forwards traffic to the IGW.
7. The IGW translates the source IP address of the outbound traffic to the EIP associated with the load balancer and forwards the traffic to the internet.

Figure 38 Centralized inbound security—return traffic flow



Health checks ensure that each path is operational and that instances are operational. The checks also monitor the return path is operational as the public load balancer probes through the firewall to the internal load balancer. The internal load balancers probe the instances in their target group to make sure they are operational.

East-West Traffic

East-west traffic, or traffic between VPCs, flows through the VM-Series firewalls in the security VPC. This traffic follows the same path as outbound traffic and does not require additional configuration in the AWS VPCs beyond what is required for outbound traffic flows.

As with the outbound traffic, you use VM-Series firewall security policies to limit what applications and resources the instances in each VPC can reach. To differentiate east-west traffic flows from outbound traffic flows on the VM-Series firewall, use zone information. Outbound traffic flows are inter-zone flows using the zones applied to the private subinterface and the public dataplane interface. East-west traffic flows are intra-zone flows and use only the security zone applied to the private subinterface.

Note

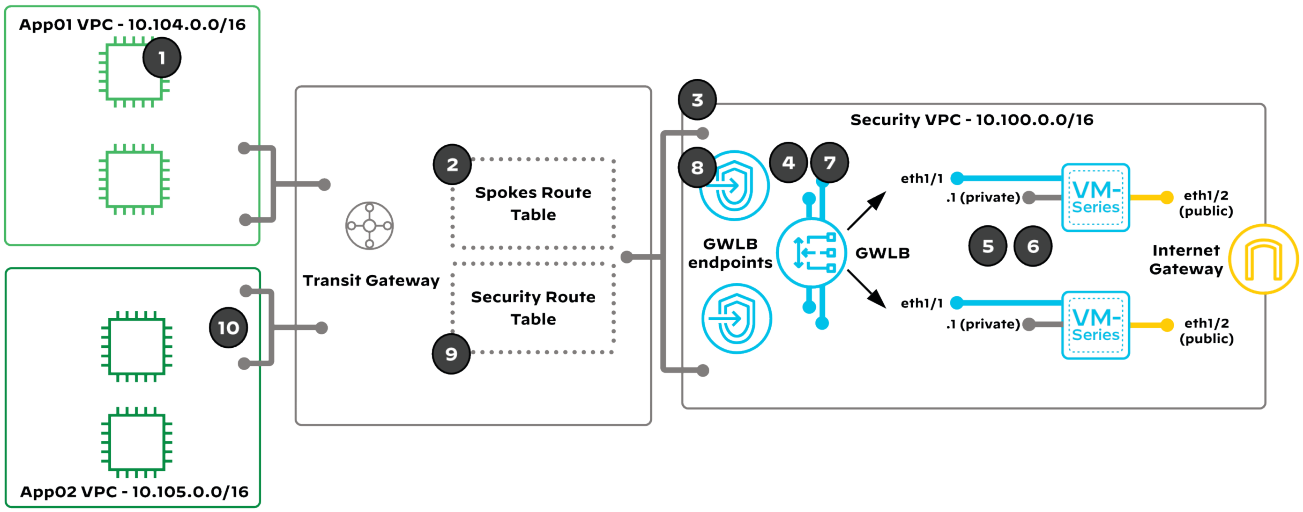
Because the east-west traffic flows are intra-zone, you should modify the default intra-zone policy to block traffic.

East-West Traffic Flow

1. In the first application VPC, an instance sends traffic to an instance in the second application VPC. The route table applied to the instance's subnet directs the traffic to the TGW attachment in its availability zone.
2. On the TGW, the application VPCs are associated with the spokes route table. The TGW uses the static route in the route table to forward the traffic to the security VPC.
3. In the security VPC, the route table applied to the subnet that contains the TGW attachment has a default route that forwards traffic to the GWLB endpoint in the attachment's availability zone.
4. In the security VPC, the GWLB endpoint forwards the traffic to the GWLB, which chooses a VM-Series firewall from its target group. Because cross-zone load balancing is enabled, it could be any available firewall.
5. The firewall receives the traffic from the GWLB and associates it with the private zone, which is configured on the subinterface that is mapped to the GWLB endpoints in the security VPC.
6. The firewall applies security policies before forwarding it out its private dataplane interface. The firewall's route table directs the traffic destined to the application VPCs out of the private interface. Because this is the same interface that received the traffic from the GWLB, the firewall uses the session information, including the TLVs associated with the Geneve encapsulation to return the traffic to the GWLB.
7. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic.

8. In the security VPC, the route table associated with the subnet that contains the GWLB endpoint has a route that forwards traffic destined for the second application VPC's address space to the TGW attachment in its availability zone.
9. On the TGW, the security VPC is associated with the security route table. The TGW uses that route table to forward traffic to the second application VPC.
10. In the second application VPC, the TGW attachment forwards the traffic to the instance.

Figure 39 East-west traffic flow

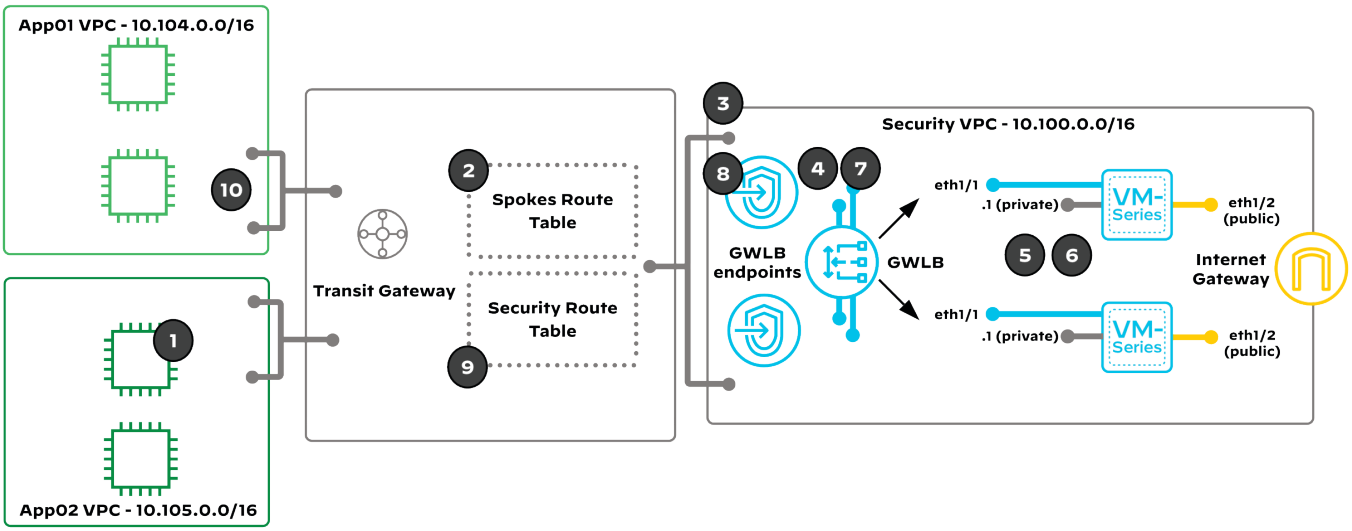


Return Traffic Flow

1. In the second application VPC, an instance sends return traffic to an instance in the first application VPC. The route table applied to the instance's subnet directs the traffic to the TGW attachment in its availability zone.
2. On the TGW, the application VPCs are associated with the spokes route table. The TGW uses the static route in the route table to forward the traffic to the security VPC.
3. In the security VPC, the route table applied to the subnet that contains the TGW attachment has a default route that forwards traffic to the GWLB endpoint in the attachment's availability zone.
4. In the security VPC, the GWLB endpoint forwards the traffic to the GWLB, which chooses the same VM-Series firewall from its target group..
5. The firewall receives the traffic from the GWLB and associates it with the private zone, which is configured on the subinterface that is mapped to the GWLB endpoints in the security VPC.

6. The firewall receives the traffic and associates the return traffic to an active session on the firewall. The firewall's route table directs the traffic destined to the application VPCs out of the private interface. Because this is the same interface that received the traffic from the GWLB, the firewall uses the session information, including the TLVs associated with the Geneve encapsulation, to return the traffic to the GWLB.
7. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic.
8. In the security VPC. The route table associated with the subnet that contains the GWLB endpoint has a route that forwards traffic destined to the first application VPCs address space to the TGW attachment in its availability zone.
9. On the TGW, the security VPC is associated with the security route table. The TGW uses that route table to forward traffic to the second application VPC.
10. In the first application VPC, the TGW attachment forwards the traffic to the instance.

Figure 40 Return traffic flow



Backhaul to On-Premises Networks

To get traffic from on-premises resources to private instances, you can use VPN connections or AWS Direct Connect. VPN connections from on-premises gateways connect to the TGW as a VPN attachment. Multiple tunnels and ECMP provide resiliency. The default route in the spokes route table provides the path that allows traffic from instances in the spoke VPCs to reach on-premises resources.

You can backhaul to the TGW with Direct Connect either directly in a colocation facility or from on-premises as a service through a WAN provider. Dual connectivity is recommended for resiliency.

Figure 41 Backhaul with Direct Connect gateway

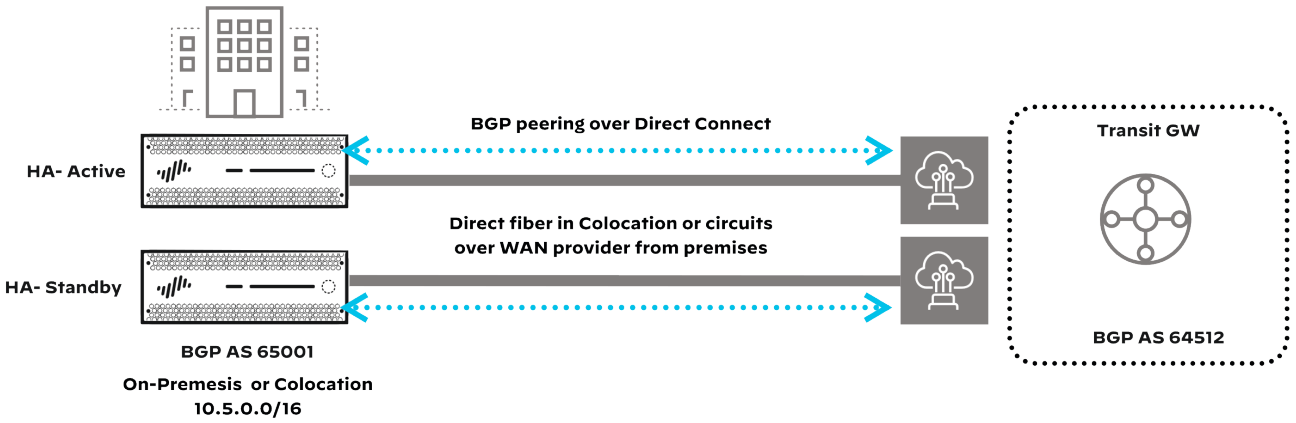
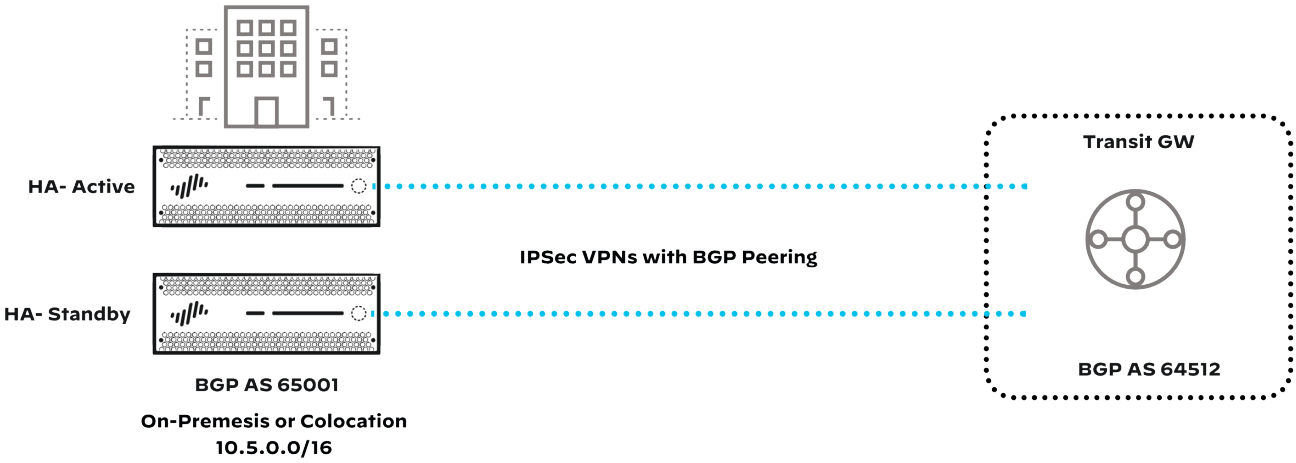


Figure 42 shows VPN connectivity from on-premises gateways to the TGW via a VPN attachment. There is a VPN attachment for each CGW, and each attachment is made of two tunnels.

Figure 42 Backhaul with VPN



Management Traffic

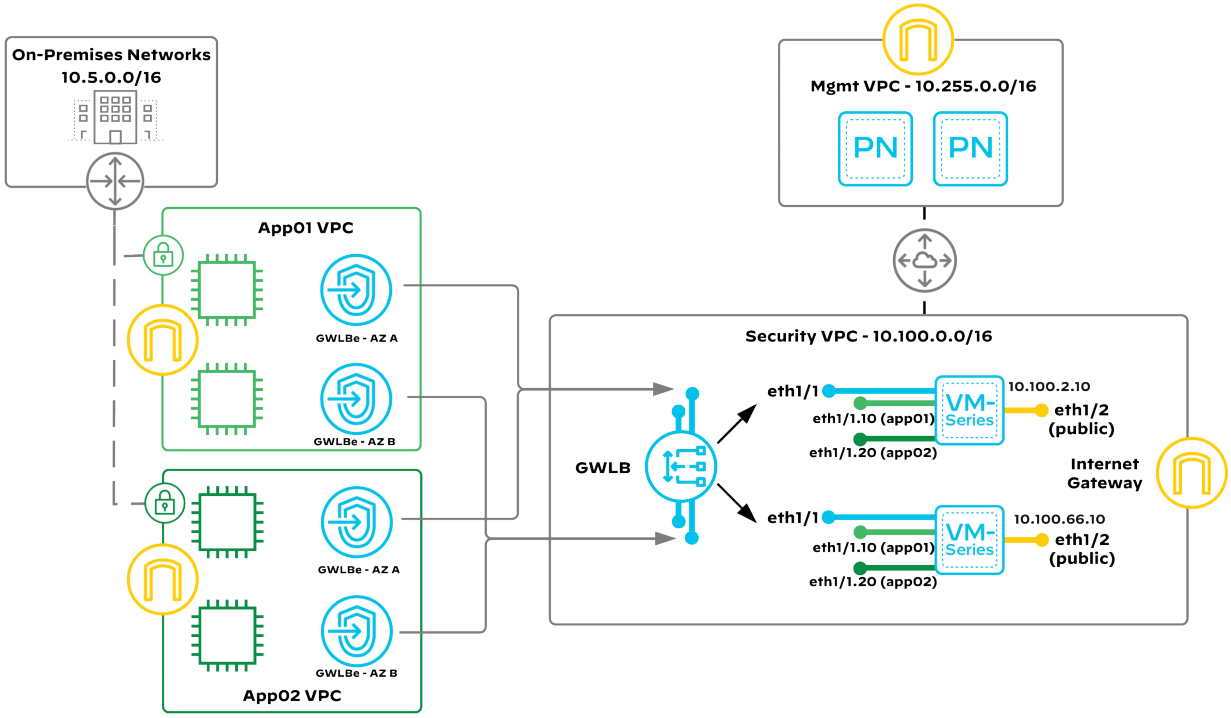
This design uses Panorama for the management of the firewalls and uses Cortex Data Lake for logging. You deploy Panorama in an active/standby configuration in a separate, dedicated VPC. You deploy the firewalls with a management interface that routes to Panorama and the internet for software and content updates. The firewalls also need connectivity to subscription services and Cortex Data Lake for logging.

This design connects the management VPC to the firewalls via the Transit gateway.

ISOLATED DESIGN MODEL

The Isolated Design model centralizes the security instances in a dedicated security VPC, providing inbound and outbound security services for one or more isolated VPCs.

Figure 43 Isolated design

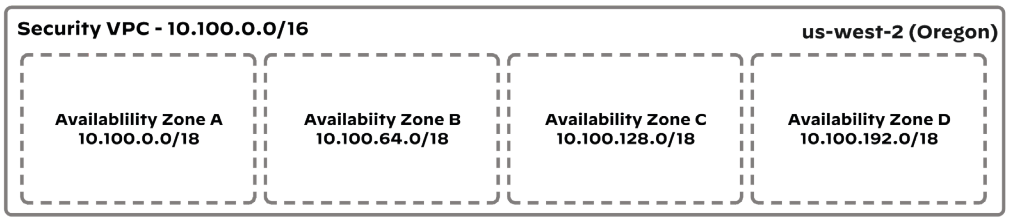


Security VPC

This design deploys a VPC dedicated to security. In the security VPC, you deploy the VM-Series firewalls in separate availability zones and deploy a GWLB to distribute traffic to the firewalls.

When deploying the security VPC, configure it with a non-overlapping IP address block that is appropriately sized for your organization's needs. For resiliency, use multiple availability zones and consider breaking your subnet ranges into availability zone-based blocks that you can summarize in security groups and routing tables.

Figure 44 VPC availability zone-based IP blocks



To supply outbound internet access, you must deploy an IGW into the VPC. The IGW performs network address translation of the management interface's private IP address to its associated public IP address and allows outbound traffic from the firewall to reach the internet.

VM-Series Firewalls

Deploy two or more VM-Series firewalls in separate availability zones.

Although the VM-Series firewall supports multiple interface deployment configurations such as virtual wire, Layer 2, and tap mode, VM-Series firewall interfaces on AWS are always Layer 3 interfaces because of AWS networking requirements. In AWS, you should always configure VM-Series firewall interfaces to obtain their IP address through DHCP.

Because this design model uses overlay routing, the VM-Series firewall requires three Ethernet interfaces: a management interface, a private dataplane interface for traffic from the GWLB, and a public dataplane interface for outbound traffic.

When you deploy a VM-Series instance from the AWS Marketplace, it has a single interface by default. You have to create the additional interfaces and associate them with the VM-Series instance. You also need to create two Elastic IP addresses. Assign one to the management interface so you can manage the firewall and the other to the public interface so the VM-Series firewall can support outbound traffic flows.

Although the dataplane interfaces both get their IP addresses from DHCP, only the public interface should accept the default route.

Attach each firewall interface to separate subnets. The management and public subnets have separate route tables, as follows:

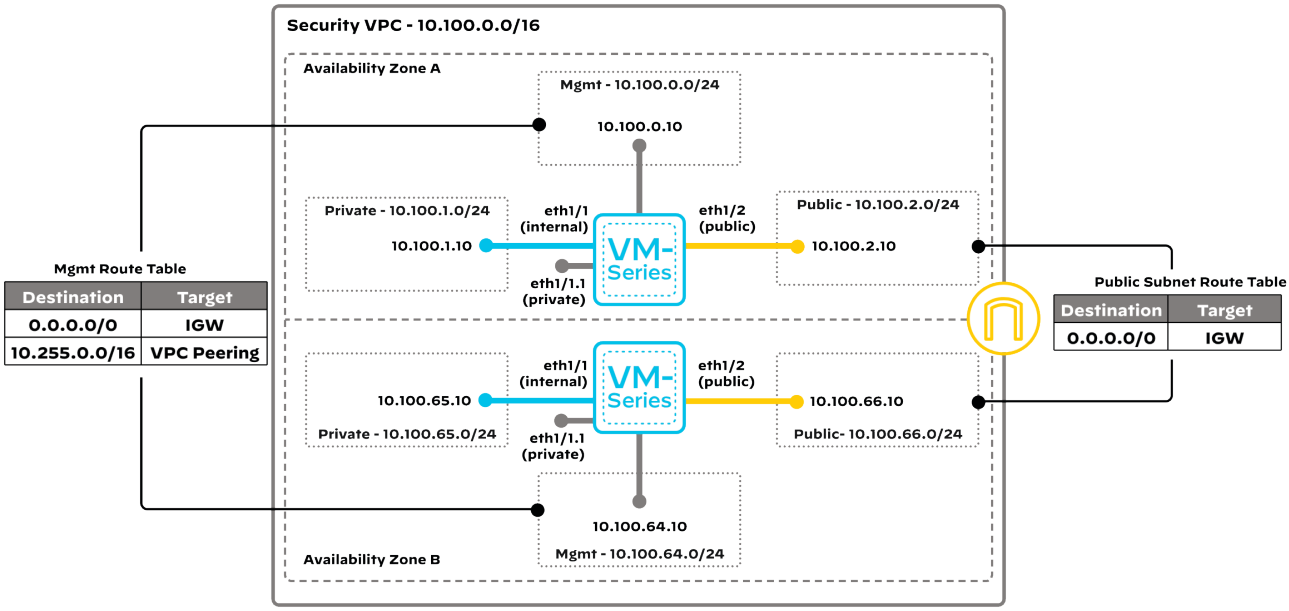
- The management route table has all management subnets assigned to it, a default route to the IGW for internet access, and a route to the VPC peering connection for access to Panorama.
- The public route table has all public subnets assigned to it and a default route to the IGW for internet access.

You do not need to modify the default routing of the subnets dedicated to the private dataplane interface. However, you should configure a security group on the firewall's private dataplane interface that allows health checks and UDP traffic destined to port 6081 from all the GWLB subnets. The security group should deny all other traffic.

The private dataplane interface should have at least one subinterface. The GWLB endpoints should map to subinterfaces instead of the dataplane interface. Mapping all endpoints to a subinterface allows you to have a restrictive security policy on the dataplane interface that allows health checks only from the GWLB subnets. This security policy provides a layer of protection if an endpoint is misconfigured or newly added to the GWLB.

> **Note**
>
> Although the firewall does not use the subinterfaces for traffic forwarding, you must configure them to receive a DHCP address and to have a unique interface name. AWS does not assign the subinterface an IP address, which is expected and does not affect traffic forwarding with a GWLB.

Figure 45 VM-Series firewall interfaces and AWS route tables



You must also enable Geneve inspection and overlay routing on the firewall.

> **Note**
>
> With Geneve inspection enabled, the firewall is able to look inside the Geneve encapsulation to the actual traffic. If you don't enable Geneve inspection, the firewall sees traffic destined to the dataplane interface IP address with a destination port of 6081.

You can automate the VM-Series configuration management by using Panorama in the management VPC to provide a secure, scalable, and automated architecture. Bootstrapping speeds up the process of configuring and licensing the firewall and making it operational on the network. This process allows deployment of the firewall with only a basic configuration so that it can connect to Panorama and obtain the complete operational configuration.

Although you can enable aws-gwlb-inspect and map the GWLB endpoints to subinterfaces after deployment, you must do so directly on the firewall and you cannot configure it in Panorama. Because a manual per-device configuration can be challenging as you scale out devices, you should configure Geneve inspection and overlay routing during bootstrap instead.

GWLB

A GWLB in the security VPC transparently distributes traffic across the VM-Series firewalls. The GWLB should be deployed in all the security VPC's availability zones, with a single endpoint service.

The GWLB targets the VM-Series firewalls instances and uses HTTPS heath checks to monitor their availability.

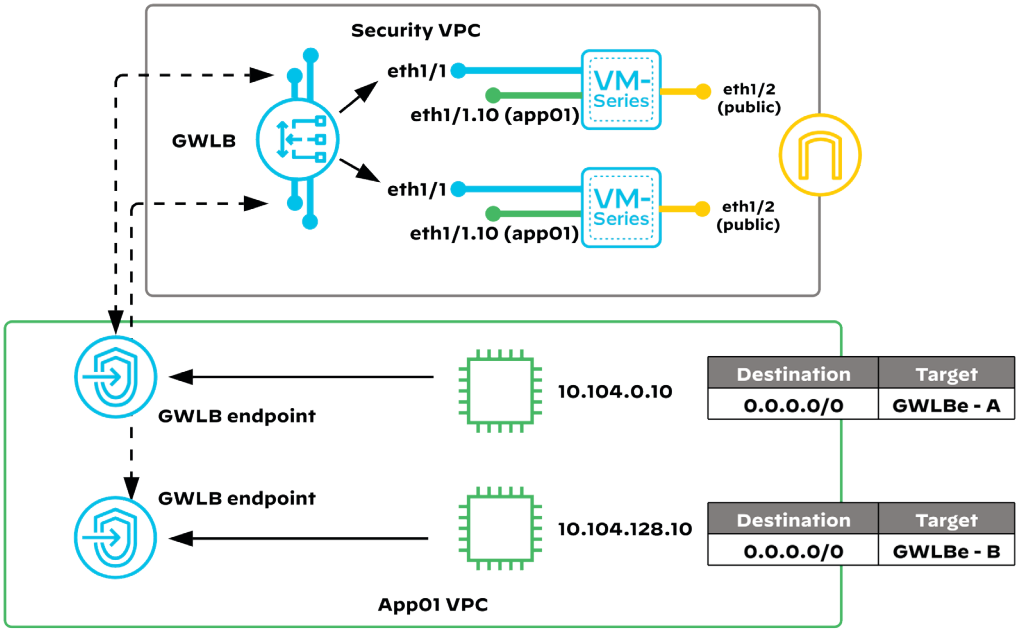
Outbound Traffic

This design uses overlay routing for outbound security on the VM-Series firewalls. Outbound traffic from instances in the isolated VPCs uses the PrivateLink connections from GWLB endpoints in the applications VPCs to the GWLB in the security VPC to egress the AWS environment through the VM-Series firewalls.

To support outbound traffic flows, in each of the isolated application VPCs, deploy GWLB endpoints in each of the availability zones that the instances are deployed into. The GWLB endpoints are associated with dedicated subnets, which allows them to have a unique routing table.

A routing table for each of the instance subnets direct all outbound traffic to the GWLB endpoint in the instance's availability zone. You do not need to modify the default routing of the subnets dedicated to the GWLB. By default, they can reach all the IP addresses within the isolated application VPC.

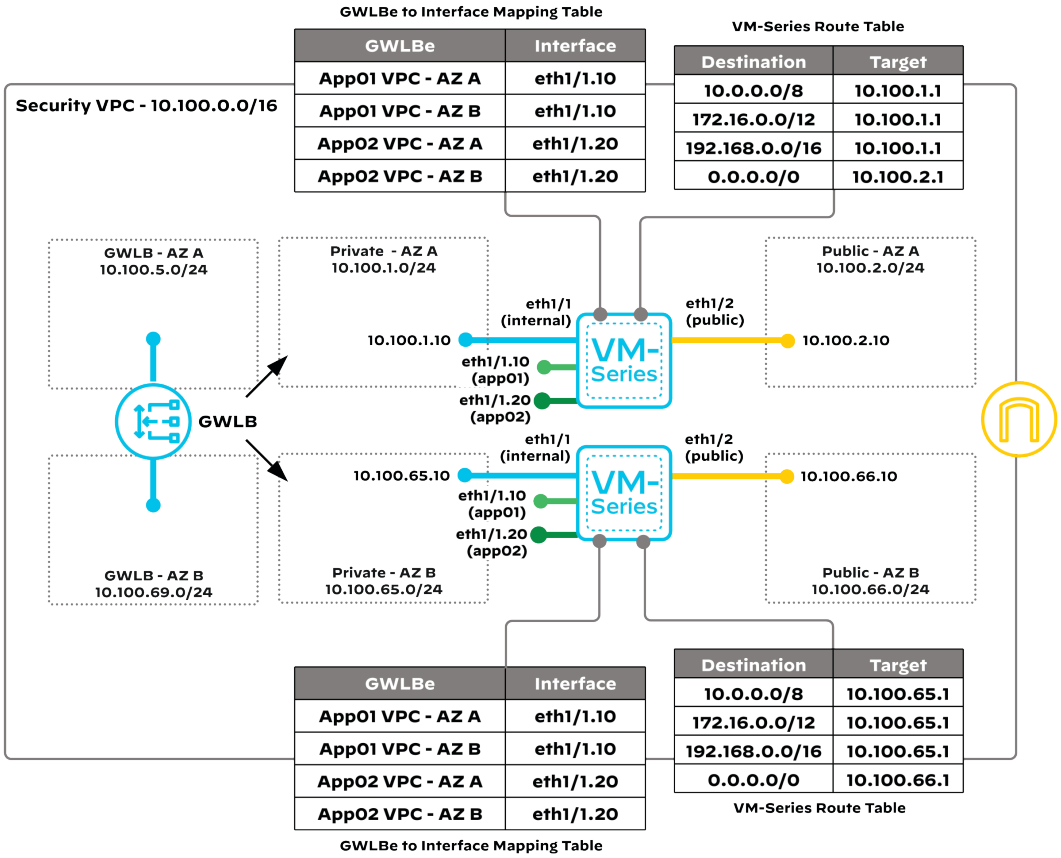
Figure 46 Outbound network and route table details



On the VM-Series firewalls in the security VPC, create a subinterface on the private dataplane interface for each isolated application VPC. Map the GWLB endpoints in the VPC to its subinterface. The subinterface must have a unique security zone and interface name. The zone is used in security policy, but the subinterface itself is not used in traffic forwarding.

The firewall also has routes to the isolated application VPC address space—or for simplicity, routes to all the RFC 1918 address space—that point to the private dataplane interface's default gateway.

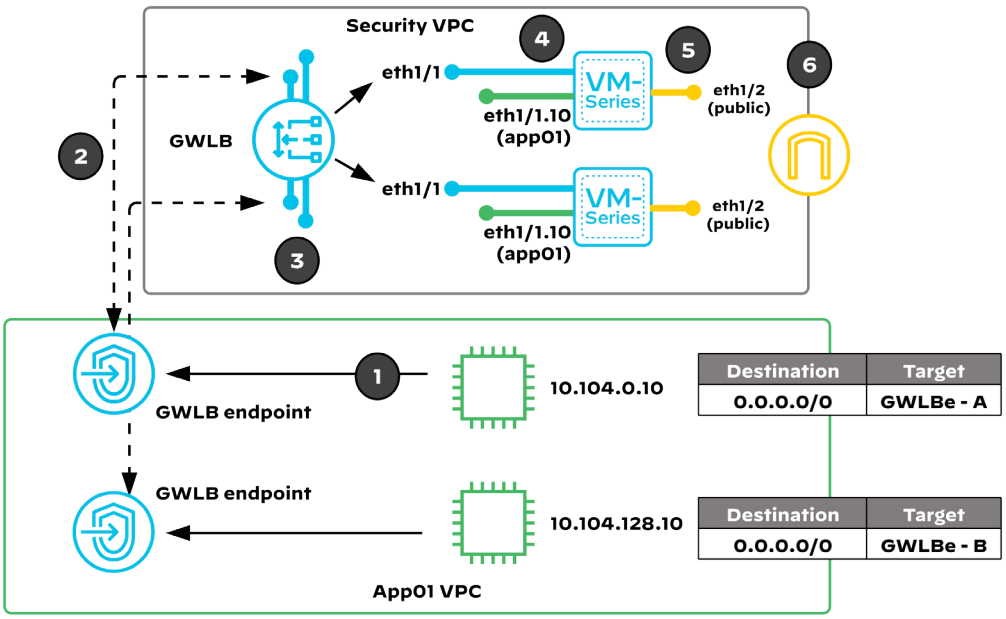
Figure 47 Firewall endpoint mapping and routing



Outbound Traffic Flow

1. In the application VPC, the instance in the application VPC sends traffic to a service on the internet. The route table applied to the instance's subnet forwards traffic to the GWLB endpoint in its availability zone.
2. In the application VPC, the GWLB endpoint forwards the traffic over PrivateLink to the GWLB in the security VPC.
3. In the security VPC, the GWLB chooses a VM-Series firewall from its target group. Because cross-zone load balancing is enabled, it could be any available firewall.
4. The firewall receives the traffic from the GWLB and associates it with the application VPC's zone, which is configured on the subinterface that is mapped to the GWLB endpoints in the application's VPC.
5. The firewall applies the Decryption, NAT, and Security policies before forwarding it out its public dataplane interface based on the default route learned from DHCP on that interface. The outbound traffic's source IP address is translated to the firewall's public interface's IP address.
6. In the security VPC, the route table applied to the subnet that contains the firewall's public interface has a default route that forwards traffic to the IGW. The IGW translates the source IP address of the outbound traffic to the EIP associated with the firewall's public interface and forwards the traffic to the internet.

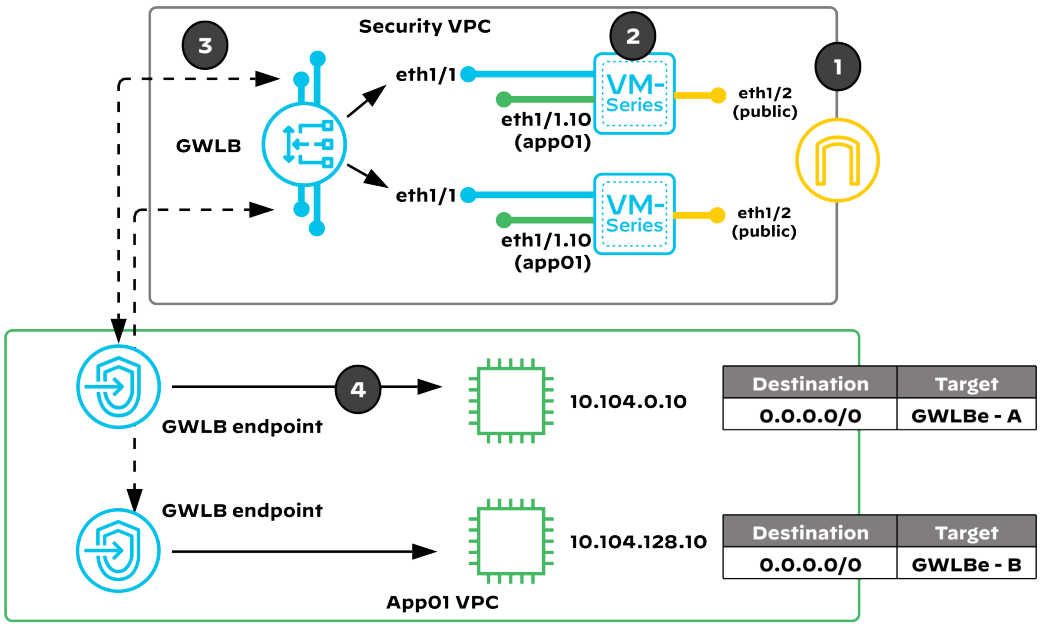
Figure 48 Outbound traffic flow



Return Traffic Flow

1. In the security VPC, the IGW translates the destination IP address of the return traffic to the firewall's public interface's IP address.
2. The firewall receives the traffic and associates the return traffic with an active session on the firewall. The firewall's route table directs the traffic destined to the application VPC out the private interface. Because this is the same interface that received the traffic from the GWLB, the firewall uses the session information, including the TLVs associated with the Geneve encapsulation, to return the traffic to the GWLB.
3. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic in the application VPC.
4. In the application VPC, the GWLB endpoint forwards the traffic to the instance.

Figure 49 Return traffic flow



Security Policy

You use VM-Series firewall security policies to limit what applications and resources the private instances can reach. In most designs, the VM-Series firewall does not need to translate the destination IP address.

The VM-Series firewall security policy allows appropriate application traffic from private instances to the internet. Using the zones of the private subinterface and the public dataplane interface to define the policy. You should implement the outbound security policy by using positive security policies (*allow listing*). Security profiles prevent known malware and vulnerabilities from entering the network in return traffic allowed by the security policy. URL filtering, Data Loss Prevention, file blocking, and data filtering protect against data exfiltration.

Inbound Traffic

Inbound traffic originates outside the VPC and is destined to applications or services hosted within your VPCs, such as web servers. This design uses the GWLB and VM-Series firewalls in the security VPC, with GWLB endpoints in the application VPCs for the transparent inspection of inbound traffic.

This design uses the existing VM-Series firewalls in the security VPC and builds upon the previously discussed configuration to support outbound traffic flows. A separate set of firewalls dedicated to inbound traffic flows might be more appropriate if you have regulatory or other compliance concerns.

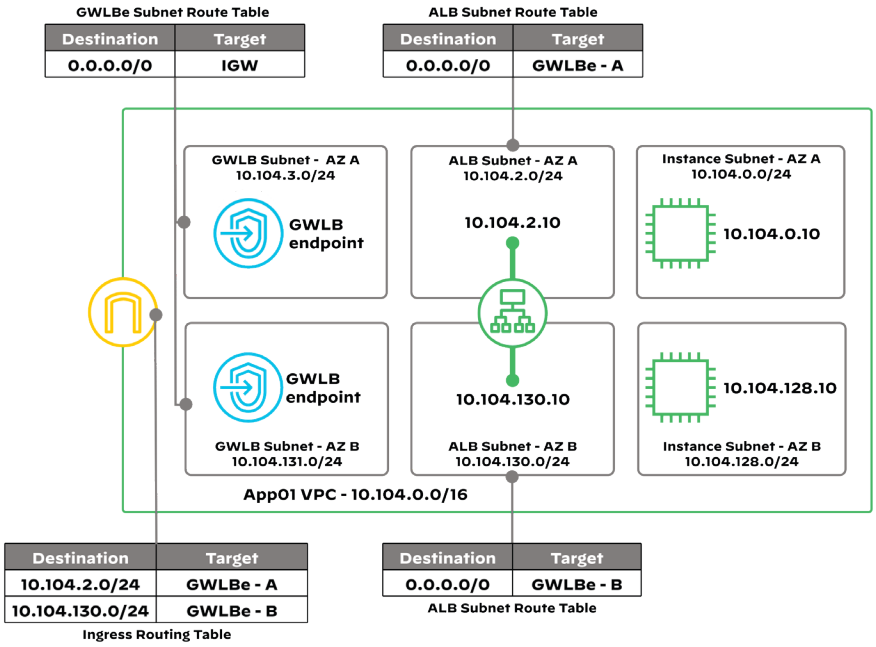
To support inbound security with this design option, modify each application VPC that requires inbound security as follows:

- To supply inbound access, deploy an IGW to the application VPC.
- Add a gateway route table that directs traffic to the subnets in an availability zone to the GWLB endpoint in that availability zone.
- Add an internet-facing load balancer with the application instances in the VPC as the targets. Individual instances with associated EIPs are also supported.
- Add a route table for the GWLB endpoint subnets. The route table must direct traffic that is destined to the internet to the IGW.
- Add a route table for each of the subnets that contain the load balancer. The route tables must direct traffic that is destined to the internet to the GWLB endpoint in its availability zone.

Note

Because inbound traffic flows are intra-zone, you should modify the default intra-zone policy to block traffic.

Figure 50 Isolated inbound security—application VPC



In this design, the VM-Series firewalls inspect and transparently secure traffic before the internet-facing load balancer receives it. Because this design option uses GWLB for resiliency, the firewalls do not modify the traffic source or destination IP address.

Because the firewall is in the first device in the traffic path, if you want to filter traffic before the firewall receives it, you must use a network ACL on the GWLB endpoint subnets. If you are using an ALB, the firewall decrypts and re-encrypts the traffic before the ALB sees it.

The design option supports one or more application VPCs. Each application VPC that requires inbound security requires and IGW, and GWLB endpoints. The VM-Series firewalls in the security VPC inspect and secure the traffic, relying on the AWS PrivateLink connection between the GWLB endpoint and the GWLB to transmit traffic between the application VPC and the security VPC.

The VM-Series firewall security policy allows application traffic to the internet-facing load balancer, and VM-Series firewall security profiles prevent known malware and vulnerabilities from entering the network in traffic allowed by the security policy.

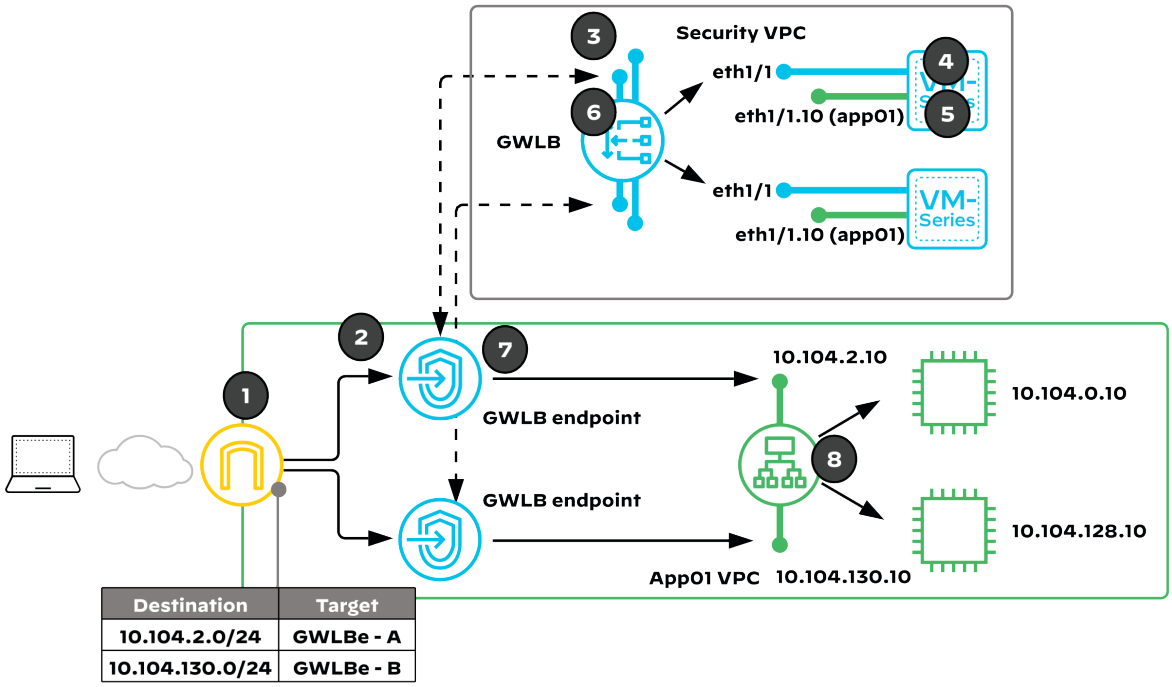
Inbound Traffic Flow

1. The URL request from the end-user is directed toward example.lb.aws.com. This request is sent to a DNS. DNS returns an IP address for each of the load balancer's enabled availability zones. The client OS picks an IP address and sends the traffic. In the application VPC, the IGW translates the destination public IP address to the IP address of the load balancer.

2. In the application VPC, the gateway route table associated with the IGW, directs the traffic to the GWLB endpoint in the destination's availability zone. The GWLB endpoint forwards the traffic to the GWLB in the security VPC.

3. In the security VPC, the GWLB chooses a VM-Series firewall from its target group. Because cross-zone load balancing is enabled it could be any available firewall.
4. The firewall receives the traffic from the GWLB on its private dataplane interface and associates it with the application's zone, which is configured on the subinterface that is mapped to the GWLB endpoints in the application's VPC.
5. The firewall's route table directs the traffic destined to the application VPC out the private interface. The firewall applies security policies before forwarding. The security policy is an intra-zone policy. Because this is the same interface that received the traffic from the GWLB, the firewall uses the session information, including the TLVs associated with the Geneve encapsulation, to return the traffic to the GWLB.
6. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic.
7. In the application VPC, the GWLB endpoint forwards the traffic to the internet-facing load balancer.
8. In the application VPC, the internet-facing load balancer receives the traffic and picks a target from the target group. The targets are the application instances. The load balancer translates the packet's destination address to the selected instance interface IP address and translates the source IP address to the private IP address of the load balancer so that traffic returns to it on the return flow.

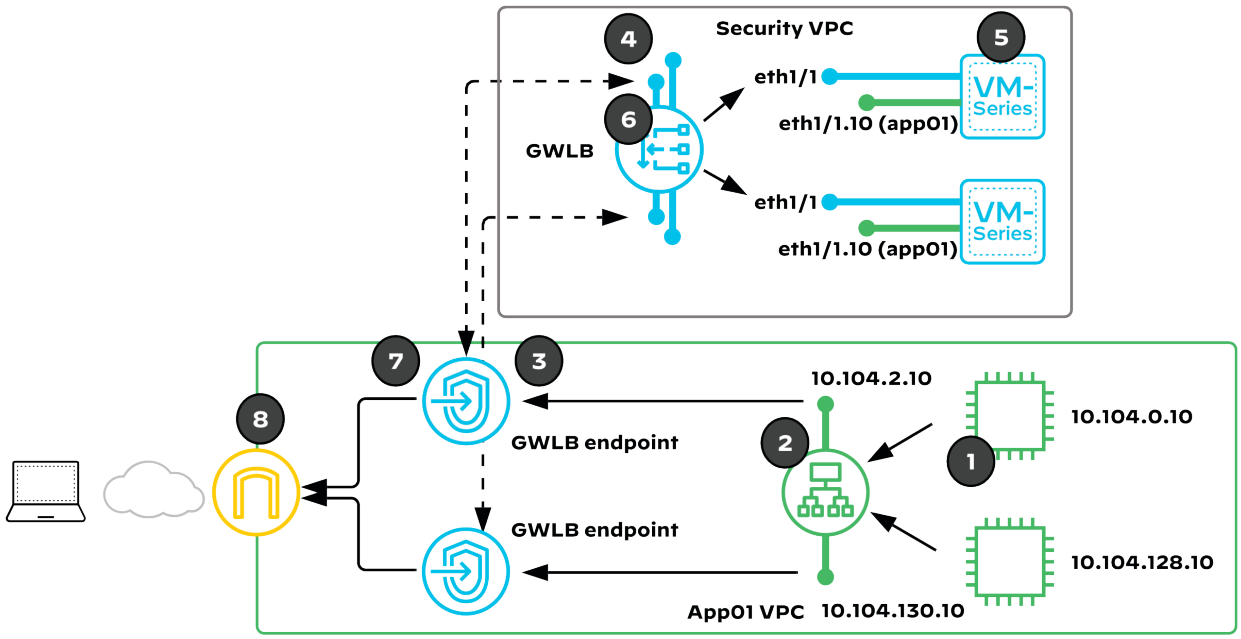
Figure 51 Inbound traffic flow



Return Traffic Flow

1. In the application VPC, the instance sends return traffic to the load-balancer.
2. In the application VPC, the load balancer modifies the source and destination IP addresses. The route table applied to the subnet that contains the load balancer has a default route that forwards traffic to the GWLB endpoint in its availability zone.
3. The GWLB endpoint forwards the traffic to the GWLB in the security VPC.
4. In the security VPC, the GWLB chooses the same VM-Series firewall from its target group that was used for the inbound traffic flow.
5. The firewall receives the traffic from the GWLB on its private dataplane interface. The firewall receives the traffic and associates the return traffic to an active session on the firewall. The firewall uses the session information, including the TLVs associated with the Geneve encapsulation to return the traffic to the GWLB.
6. In the security VPC, the GWLB uses the session information to forward the traffic to the GWLB endpoint that received the original traffic.
7. In the application VPC, the GWLB endpoint forwards the traffic to the IGW.
8. The IGW translates the source IP address of the outbound traffic to the EIP associated with the load balancer and forwards the traffic to the internet.

Figure 52 Return traffic



East-West Traffic

VPC networking provides direct reachability between all instances within a VPC, regardless of IP address and subnet allocation. All instances within a VPC can reach any other instance within the same VPC, regardless of AWS route tables. You can use AWS security groups and ACLs to permit or deny traffic into or out of an instance or group of instances.

East-west traffic between subnets within a VPC always goes directly between instances. AWS route tables cannot override this behavior, and a limitation of this design model is that the VM-Series firewall cannot have control over or visibility to east-west traffic. You can use network ACLs to restrict traffic between subnets. Still, they are not a replacement for the visibility and control provided by the VM-Series firewalls.

The Centralized design model provides a scalable design for inspection and control of east-west, inbound, and outbound traffic.

Backhaul to On-Premises Networks

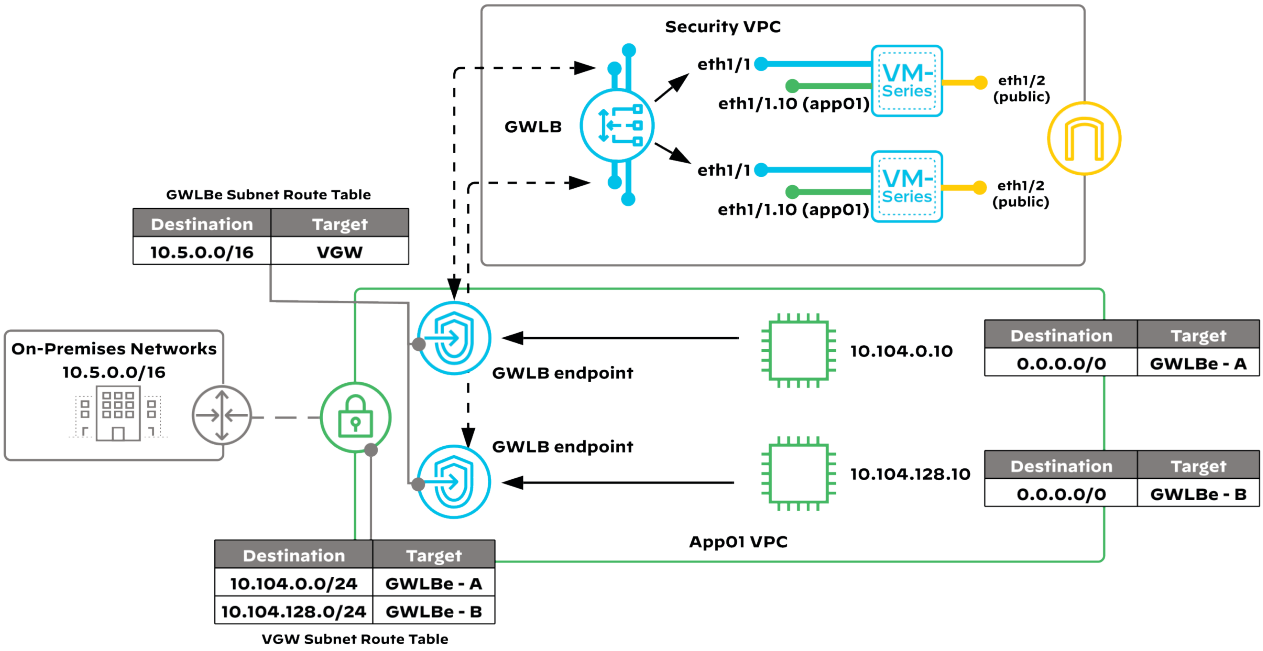
It can be convenient to have direct connectivity between the private IP addresses of hosts in your on-premises networks to the private IP addresses of your instances in AWS. Direct connectivity also helps administrators and developers to reach resources that do not have public IP address access. You should control access between the on-premises networks and the instances in AWS by protecting the connection with VM-Series or PA-Series firewalls.

To have the VM-Series firewall control traffic between the VPC and the on-premises networks, direct inbound packets that enter through the VGW to communicate through the firewall with a gateway route table and additional GWLB endpoints.

To support backhaul security with this design, modify each application VPC that requires backhaul security as follows:

- To supply backhaul access, deploy an VGW to the application VPC.
- Add GWLB endpoints in each of the availability zones and associate them with the GWLB in the security VPC. The GWLB endpoints should be in dedicated subnets so their traffic can be uniquely directed.
- Map the GWLB endpoints to a new subinterface and zone on the firewalls.
- Ensure the firewall has a route to the on-premises networks pointing out its private dataplane interface.
- Add a gateway route table that directs traffic to the subnets in an availability zone to the GWLB endpoint in that availability zone.
- Add a route table for the GWLB endpoint subnets. The route table must direct traffic that is destined to the on-premises networks to the VGW.
- Add a route table for each of the instance and load balancer subnets. The route tables must direct traffic that is destined to the on-premises networks to the GWLB endpoint in its availability zone.

Figure 53 VPN connections to the VGW bypass the VPC firewall



Management Traffic

This design uses Panorama for the management of the firewalls and uses Cortex Data Lake for logging. You deploy Panorama in an active/standby configuration in a separate, dedicated VPC. You deploy the firewalls with a management interface that routes to Panorama and the internet for software and content updates. The firewalls also need connectivity to subscription services and Cortex Data Lake for logging.

This design connects the management VPC to the security VPC via VPC peering.

Summary

Moving applications to the cloud requires the same enterprise-class security as your private network. The shared-security model in cloud deployments places the responsibility of protecting applications and data on your organization. Deploying Palo Alto Networks VM-Series firewalls in your AWS infrastructure provides a scalable infrastructure with protections from known and unknown threats, complete application visibility, a common security policy, and native cloud automation support. Your ability to move applications to the cloud securely helps you to meet challenging business requirements.

HEADQUARTERS

Palo Alto Networks
3000 Tannery Way
Santa Clara, CA 95054, USA
https://www.paloaltonetworks.com

Phone: +1 (408) 753-4000
Sales: +1 (866) 320-4788
Fax: +1 (408) 753-4001
info@paloaltonetworks.com

You can use the **feedback form** to send comments about this guide.

B-000110P-1-22a