



PROTECTING DEVELOPER ENVIRONMENTS WITH THE VM-SERIES ON AWS

Table of Contents

Introduction	3
Security Approaches for Protecting Developer Environments	3
Protecting Developer Environments With the VM-Series	3
Phase 1: Reducing Your Threat Exposure With Application Control and URL Filtering	3
Phase 2: Preventing Threats and Blocking Outbound Data Theft	4
URL Filtering and App-ID: Controlling Web Resource Access	5
Augmenting Web Access Control Policies With Third-Party Data Sources	5
Automating Policy Updates and Actions	5
Quarantine Infected Workloads Automatically	5
Automating Policy Updates as Workloads Are Added/Removed	6
Prevent Threats With Amazon GuardDuty and Dynamic Address Groups	7
Scaling to Protect Many VPCs	7
VM-Series Next-Generation Firewall Licensing Options	8
Conclusion	8

Introduction

The move to Amazon® Web Services, or AWS®, is enabling many organizations to adopt a more agile, iterative application development methodology. In so doing, developers and their workloads need rapid, oftentimes automated, yet secure access to web-based resources, such as GitHub®, Yum and Apt-Get, as well as OS update mechanisms for Windows® or Linux.

The need for additional security, over and above native security to protect developer environments, is well-documented. Developers are regularly accessing web-based executables, how-to guides and workbooks of all types to get their jobs done more quickly and effectively. Attackers commonly use these same web resources to inject malware into unsuspecting networks. In 2017, security incidents involving two commonly used development apps – MongoDB® and Elasticsearch® – were discovered. In the case of MongoDB, older versions with known security flaws were unpatched and subsequently compromised, exposing users' personal data. In the Elasticsearch incident, it was found that 4,000 or so instances were distributing malware. More recent examples include:

- Critical remote code execution vulnerabilities, such as the Apache Struts remote code execution vulnerabilities, [CVE-2018-11776](#); and the Drupal Core remote code execution, [CVE-2018-7600](#).
- Malware found in Arch Linux packages available on [Arch User Repository, or AUR, the official repository of user-submitted packages for Arch Linux](#).
- Cryptocurrency miner found hidden in the source code of an [Ubuntu snap package hosted on the official Ubuntu Snap Store](#).
- A joint report by threat intelligence firms Digital Shadows and Onapsis, warning of [increased attacks on Enterprise Resource Planning, or ERP systems, such as those from SAP and Oracle](#).

One of the challenges when moving application development environments to AWS is ensuring security can keep pace with development teams. Security over and above native security is often seen as injecting delays through industry-standard change control best practices while development relies heavily on automation. The ideal approach is one that protects developer environments – and the corporate data within – while operating at the speed of the cloud.

Security Approaches for Protecting Developer Environments

Currently, there are several approaches to protecting application development environments on AWS. These approaches include:

- **Do nothing:** This approach carries the highest risk as there is zero visibility into application traffic and, more importantly, no protection from threats. Although outbound traffic may be more limited than a non-development environment, this traffic may include system updates and payment processors, as well as services such as storage, databases, analytics, SaaS applications and on-premises resources. Each of these examples represents exposure to potential risks.
- **Use native, course-grained controls:** This includes manual white- and blacklisting of specific web sites. Although this provides some level of access control, it provides no visibility into actual traffic where malicious activity can be determined and stopped, nor is it capable of preventing threats or related command-and-control, or C2, traffic.
- **Backhaul development traffic through the corporate firewall:** In a large development environment, this places a heavy, costly performance burden on the corporate firewall, slowing down development processes, increasing bandwidth costs, and introducing friction between development and security.
- **Use open source, third-party tools, such as proxies:** Although open source products are generally inexpensive, their operational costs and complexity are much higher due to a lack of enterprise-level support and in-house expertise for these business-critical applications. They require their own set of policies to administer and are incapable of sharing potential threat intelligence or coordinating action with other parts of the security architecture. In other words, they offer single-purpose security functions while requiring an inordinate amount of overhead. In some scenarios, the ability to create highly available and resilient architectures is limited, hindering deployment in business-critical environments.

Palo Alto Networks® enables you to protect your AWS deployment from threats through first reducing your threat footprint by controlling applications and web traffic, and then by preventing threats within the allowed traffic. To ensure security does not inject delays into the development process, you can employ automation to create completely touchless deployments.

Protecting Developer Environments With the VM-Series

The VM-Series virtualized next generation firewall allows you to take a two phased approach to protect your developer environments on AWS. First, application control and URL Filtering work together seamlessly to tightly control which web resources your developers and their workloads can access. The result is a smaller overall threat footprint to which you can apply Threat Prevention policies to block inbound threats while also stopping outbound C2 or data exfiltration.

Phase 1: Reducing Your Threat Exposure With Application Control and URL Filtering

- **Application control:** The VM-Series complements port-based filtering provided by AWS Security Groups by using Palo Alto Networks App-ID™ technology to classify traffic based on application. App-ID uses multiple identification techniques to determine the exact identity of applications traversing your network, including those that try to evade

detection by masquerading as legitimate traffic, hopping ports or using encryption. The application identity – whether client-server or web-based – is then used as a security policy element. Positive security model firewall rules allow applications, specific application functions or web resources to enable the business while implicitly denying all other traffic. The result is a dramatic reduction in your threat footprint.

- **URL Filtering:** Application control is augmented with secure web browsing and URL access by allowing administrators to block dangerous sites that deliver malware, attempt to circumvent security controls or attempt to steal user credentials through phishing. Palo Alto Networks URL Filtering service extends existing application-based policies to web browsing activities, using predefined or custom categories, such as Dev Tools, to control web traffic through a single policy table.

Phase 2: Preventing Threats and Blocking Outbound Data Theft

Controlling access to web resources for developer environments narrows your threat exposure, but as shown in the examples in the introduction, web resources that are “allowed by policy” may still provide access to compromised images or documents. Threat Prevention policies can help eliminate those risks by inspecting traffic both inbound from allowed resources and outbound, blocking C2 and data theft.

- **Full cyberattack lifecycle prevention** is achieved through Threat Prevention policies that combine IPS, anti-malware, C2 and data loss protection, or DLP, to protect the allowed application and web traffic from threats across the entire attack lifecycle.
 - **Network IPS functionality** prevents the exploitation of vulnerabilities such as the Apache Struts and Drupal examples. These server-side vulnerabilities in tools commonly used in public cloud deployments prove the need for vulnerability protection in cloud development environments.
 - **Anti-malware policies** prevent web attacks, including credential brute force, XSS and SQL injection attacks.
 - **C2 protections** created by Palo Alto Networks Unit 42 threat research team detect outbound malware post-infection in the event that a system is compromised with malware, coin miners or web shells.
- **File blocking and identification** provides another layer of protection for your developer environment. Allow only the inbound or outbound file types you would like to enforce by policy. This protection can be layered together with URL Filtering to only allow certain file types from certain categories of URLs. Outbound file blocking can stop the theft of data from your AWS instance.
- **Prevention of unknown threats**, often called zero-day threats, is addressed through WildFire® malware prevention service, which provides malware analysis and sandboxing. The service analyzes potentially malicious files based on several hundred behaviors, and if it deems a file malicious, WildFire delivers protections to you and all other WildFire users in as few as five minutes. Threat intelligence WildFire collects then serves to improve our overall prevention capabilities: the PAN-DB URL Filtering database is updated with newfound malicious URLs, new DNS signatures are added, and existing signatures are fine-tuned.

The content filtering and Threat Prevention capabilities of the VM-Series are integral enforcement and intelligence gathering points of the Palo Alto Networks Security Operating Platform. First and foremost, the VM-Series protects your AWS deployments from threats, data loss and business disruption. The information the VM-Series gathers as it protects your network is shared across Security Operating Platform components and with all other users to collectively and continually improve prevention capabilities.

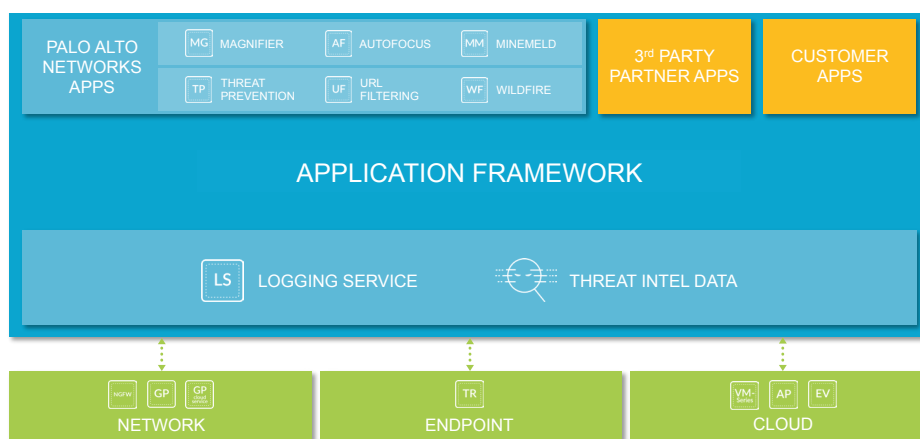


Figure 1: Palo Alto Networks Security Operating Platform

Automation allows you to accomplish two critical, prevention-focused tasks. First, it lets you embed our security into your application development workflow, ensuring your security keeps pace with development. Secondly, it lets you create policies that are dynamically updated as workloads are added or removed from your VPC, or as new, potentially malicious threats are discovered.

URL Filtering and App-ID: Controlling Web Resource Access

You can use App-ID, the VM-Series traffic classification engine, to control applications and access to web resources by manually white- and blacklisting specific known URLs, reducing your threat exposure. This binary and somewhat manual approach allows a limited set of connections and blocks everything else. A drawback to this approach is that none of the state information or background activity is logged, resulting in a lack of context and visibility into what is being blocked, which can be gleaned from traffic logs.

To exert more fine-grained control and gain visibility into your web traffic while controlling the development environment's access to web resources, you can enable the URL Filtering subscription in conjunction with App-ID application-based control. URL Filtering identifies and prevents access to dangerous websites through a unique combination of analytics and machine learning, protecting users by automatically preventing web-based attacks. You can set URL Filtering policies by simply extending firewall policies, giving you one set of rules to manage, thereby reducing complexity and operational overhead.

This takes whitelists and blacklists to another level by blocking whole categories of sites, rather than just specific URLs. For example, you can block all URL categories that have zero business need, such as Shopping, Auctions, Religion or Dynamic DNS. Even more importantly, you can block all connections to malicious categories, such as Malware, Phishing and C2. Following this approach limits your exposure to inbound threats and helps block outbound data exfiltration – often the top security concern as organizations move to the public cloud.

Beyond preventing access to malicious websites, another critical aspect of the App-ID and URL Filtering combination is the visibility it provides you into what's happening in your AWS environment. Observing attempted connections from within your AWS development environment to malicious or unsanctioned URLs in the log files or via alerts is a strong indication that something is amiss. Perhaps a server is compromised or an employee is unwittingly accessing malicious sites, putting the business at risk. Preventing known risks by blocking website categories reduces your security exposure, but adding visibility into web traffic empowers you to prioritize and quickly respond to immediate threats.

Augmenting Web Access Control Policies With Third-Party Data Sources

Another powerful way to reduce the attack surface exposed to known malicious sources is by using External Dynamic Lists to pull in third-party threat intelligence and block potential attacks automatically without making firewall rule changes. External Dynamic Lists allow you to automatically import threat indicators from third-party feeds, such as the [SANS DShield Internet Storm Center Top 20 IP list](#) or the Brute Force Blocker that sees SSH brute force attacks around the globe, and automatically prevent attacks against your AWS application deployments.

Automating Policy Updates and Actions

Increased visibility and context from traffic and threat logs is critical because it allows your business to be alerted to risks and respond before damage can be done. In the time it takes for an analyst to discover an issue and respond by checking logs and querying a security information and event management system, or SIEM, the threat may have already moved laterally or begun data exfiltration. Using automation, the visibility generated by URL Filtering can help accelerate your threat response by using information from the URL logs to automatically and progressively exert greater control, to the point of quarantining infected endpoints or workloads. Firewall policies can be automatically updated using information generated by AWS Virtual Private Cloud – VPC – tags, logs or other sources using Dynamic Address Groups.

DAGs allow you to use an abstraction layer to automate the collection of constantly changing source and destination IP addresses needed to define and enforce security policies. Using AWS tags, such as Operating System and Workload OS, as selection criteria, the VM-Series can learn the associated IP addresses as workloads are added or removed and automatically feed them to the policy as updates. You can also use DAGs to update security policies using third-party intelligence sources, such as Amazon GuardDuty™.

Quarantine Infected Workloads Automatically

For developers who may be using web-based development resources, URL Filtering can automatically provide the VM-Series with visibility into suspicious URL connections. That information can in turn be used within a DAG to modify the policy on the fly. For example, if a developer inadvertently accesses a malicious URL, the policy can be automatically updated based on the log information collected. The source IP is automatically tagged, adding it to a DAG. In this case, the policy may consist of limiting the network access of that system so it can no longer communicate with network resources containing highly confidential assets. This not only blocks access to the destination IP, but also effectively quarantines the infected system to prevent damage.

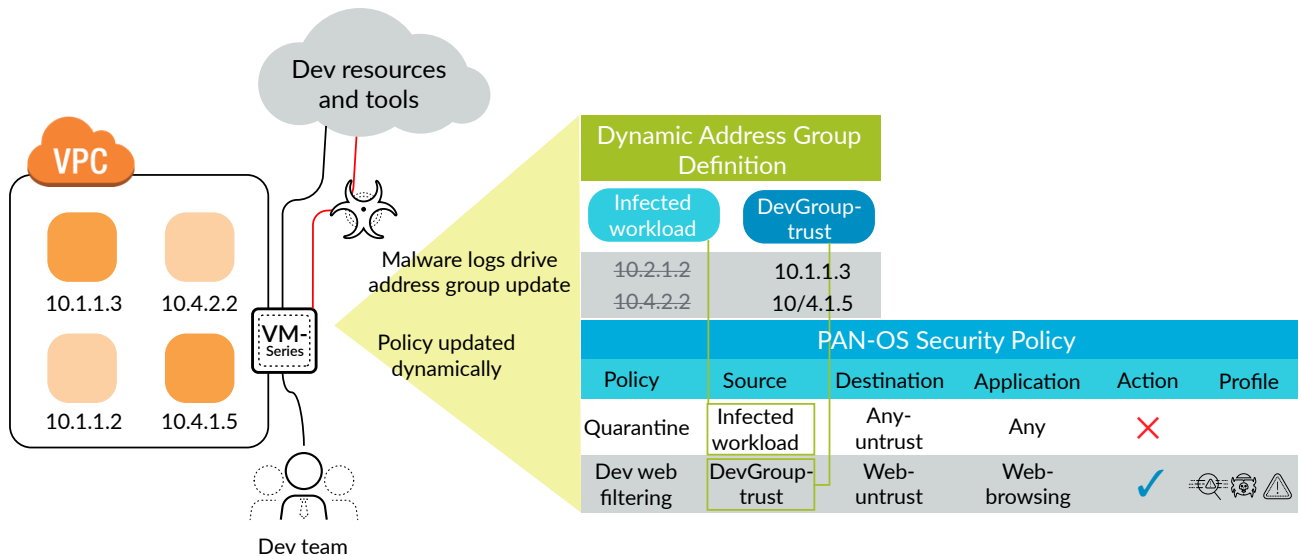


Figure 2: Using DAGs to quarantine workloads based on threat activity

InfoSec maintains remote access to the infected system, so remediation can begin immediately. The human work required here is in setting up the policies, but once completed, DAGs are automatically populated and orchestrated, addressing potential threats at machine speed without human intervention. DAGs allow for the creation of policies without knowing specifically what will trigger the Threat Prevention policies or what workloads are being added or removed, making them incredibly flexible.

For example, source IPs may exhibit indirect signs of compromise, such as connecting to Network Time Protocol, or NTP, or trying to connect to Windows Update® in an all-Linux environment. These may be suspicious indicators that do not necessarily warrant quarantine on their own. These types of indicators could be used to create a DAG that would institute a “block-continue” page to access a corporate resource. A legitimate human user will simply click “continue” and move on without disrupting the developers or the business. However, automated malware will not know how to interpret the “block-continue” page, nor will it know how to click on the continue option. This would effectively disrupt the potential exfiltration of data.

Automating Policy Updates as Workloads Are Added/Removed

In an agile and iterative development environment, new workloads are added and removed from a VPC as needed, and they will periodically need operating system updates. Rather than require development teams to submit change requests to security teams for policy updates, DAGs can use AWS tags to “learn” the IP address of a workload that is then automatically added to the policy as an update. This means no change request, no delay, developer teams work at their pace, and security keeps up automatically.

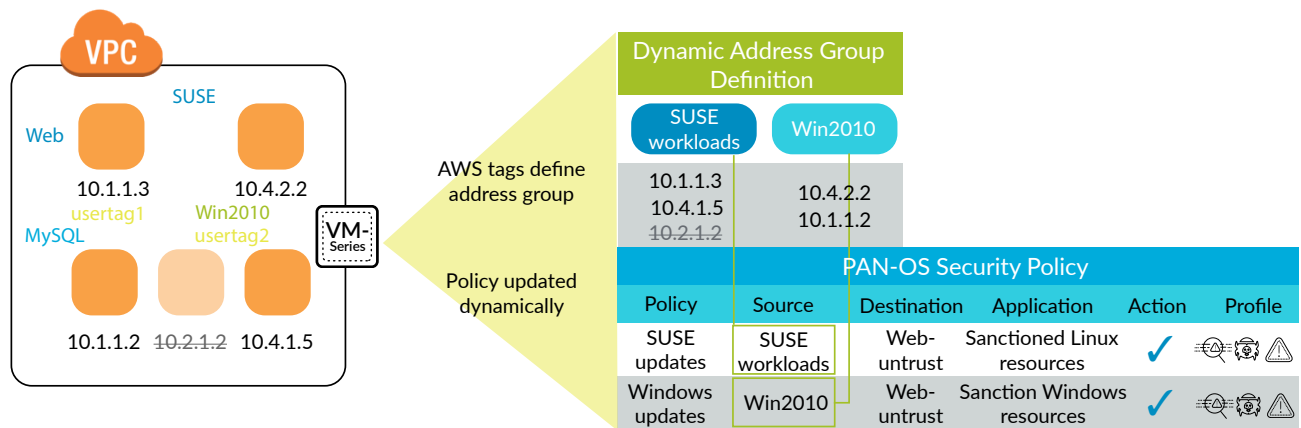


Figure 3: Using DAGs to update policies as workloads are added/removed

In the example shown, AWS tags for operating system are used to define an address group for Windows and SUSE Linux. The security policy can be unique based on the respective update resource security posture; the Windows policy may be more restrictive than the Linux policy. As new workloads are added or removed from the AWS VPC, the Operating System tag will automatically update the security policy with the new or removed IP address information.

Prevent Threats With Amazon GuardDuty and Dynamic Address Groups

DAGs can also be used to ingest external data from third-party sources, such as Amazon GuardDuty, as a means of preventing threats. When Amazon GuardDuty surfaces a potentially malicious IP address, it sends the address to AWS CloudWatch, where an AWS Lambda function feeds the IP address information to a DAG to automatically update the policy to drop the session, blocking potential attacks.

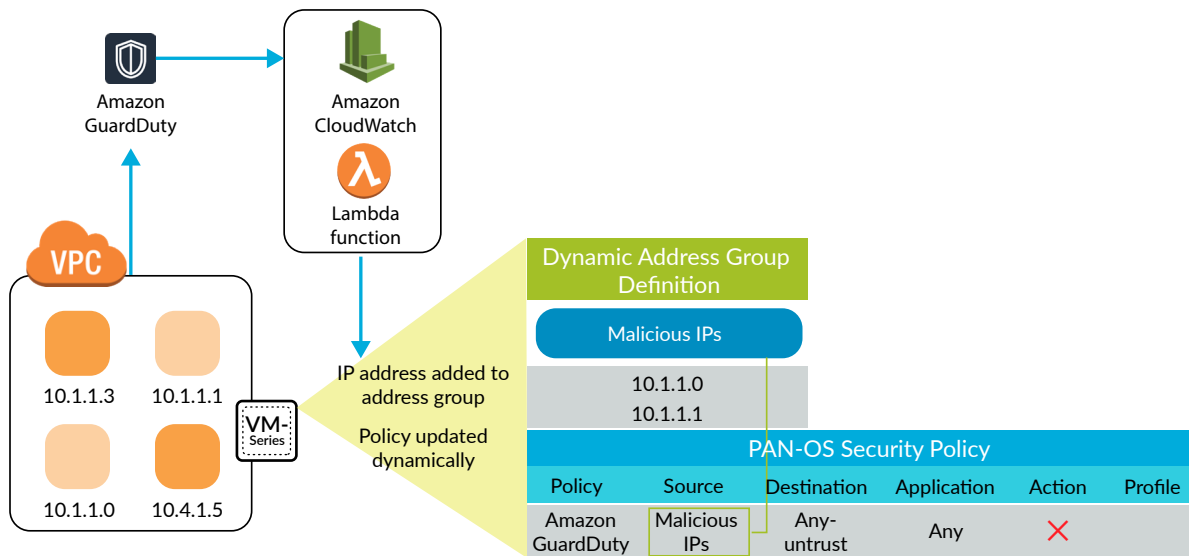


Figure 4: Feeding threat intelligence from Amazon GuardDuty into policy via DAGs

Automation allows security teams to balance protecting the business with allowing developers to operate freely and rapidly without running into security roadblocks. These orchestration examples show ways to proactively update policies without slowing down the development process.

Scaling to Protect Many VPCs

AWS architectures vary widely, but in many cases, organizations are choosing to consider resilience and scalability as key deployment elements. To scale outbound security for environments with many VPCs, a common approach is a Transit VPC, which uses a “hub-and-spoke” architecture with the VM-Series firewall deployed in the “hub” to perform secure connectivity and threat prevention functions for the subscribing VPCs, or “spokes.” Connections between VPC spokes and corporate locations or web resources will “transit” the hub VPC, sharing VM-Series next-generation security services.

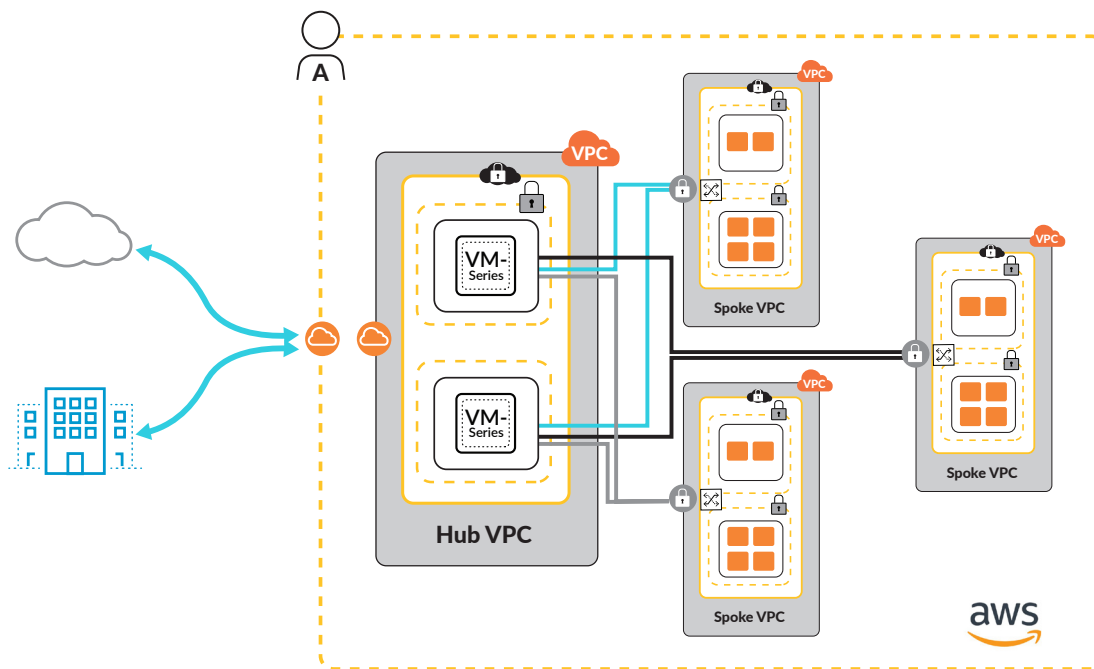


Figure 5: Transit VPC architecture used to secure many VPCs

Additional pairs of VM-Series firewalls are deployed automatically based on the defined ratio of VPC spokes per VM-Series firewall pair, either within an individual account or across multiple accounts. Using Panorama™ network security management, your security team can deploy a range of security policies to protect your AWS deployment from threats. The Transit VPC with the VM-Series is a cost-effective, easy-to-manage alternative to backhauling traffic to a corporate firewall or deploying a VM-Series per VPC. A transit architecture allows you to add and remove VPCs, applications, or accounts as spokes in a fully automated or semi-automated manner. High availability is accomplished by deploying a VM-Series in separate Availability Zones, allowing the AWS fabric to ensure uptime.

VM-Series Next-Generation Firewall Licensing Options

Deployed as an Amazon EC2® instance in your VPC, the VM-Series supports several licensing options, including consumption-based licensing via the AWS Marketplace, bring your own license and the VM-Series Enterprise Licensing Agreement, or ELA.

- **Consumption-based licensing:** Use your AWS Management Console to purchase and deploy hourly or annual VM-Series bundles directly from the AWS Marketplace.
 - **Bundle 1 contents:** VM-300 firewall license, Threat Prevention (inclusive of IPS, AV and malware prevention) subscription and Premium Support (written and spoken English only).
 - **Bundle 2 contents:** VM-300 firewall license, Threat Prevention (inclusive of IPS, AV and malware prevention), WildFire, URL Filtering and GlobalProtect™ network security for endpoints subscriptions, and Premium Support (written and spoken English only).
- **Bring your own license:** You can purchase any one of the VM-Series models, along with the associated subscriptions and support, via normal Palo Alto Networks channels and then deploy via a license authorization code through your AWS Management Console.
- **VM-Series ELA:** For large-scale deployments on AWS or across multiple virtualization environments, the VM-Series ELA allows you to forecast, and purchase upfront, VM-Series firewalls to be deployed over a one- or three-year period. The VM-Series ELA provides predictable firewall expense over the life of the term and simplifies licensing by establishing a single start and end date for all VM-Series licenses and subscriptions. Each VM-Series ELA includes a VM-Series firewall, subscriptions for Threat Prevention, URL Filtering, WildFire and GlobalProtect Gateway, plus unlimited Panorama virtual machine licenses and support.

Conclusion

Application developers are tasked with writing code as efficiently and effectively as possible. Attackers are constantly finding ways to steal data, intellectual property and computing resources located in the cloud or the data center. Using the VM-Series deployed on AWS, security teams can take advantage of automation to embed security into the application development lifecycle, allowing development teams to work at the speed of the cloud with the assurance that their work and the company data are protected.



3000 Tannery Way
Santa Clara, CA 95054
Main: +1.408.753.4000
Sales: +1.866.320.4788
Support: +1.866.898.9087
www.paloaltonetworks.com

© 2018 Palo Alto Networks, Inc. Palo Alto Networks is a registered trademark of Palo Alto Networks. A list of our trademarks can be found at <https://www.paloaltonetworks.com/company/trademarks.html>. All other marks mentioned herein may be trademarks of their respective companies. protecting-developer-environments-with-the-vm-series-on-aws-wp-100818