paloalto NETWORKS

Our Single-Pass Architecture provides:

- No additional performance overhead when enabling additional features.
- Easy management of all threat prevention aspects of security policy.
- Simplified management through fewer consoles and functional gaps for more effective security coverage.
- Significantly lower total cost of ownership.

Single-Pass Architecture

Integrated, Prevention-Oriented Security

For many years, security professionals have pursued the goal of integrating threat prevention services into the firewall to alleviate the need for additional devices for intrusion prevention systems (IPS), network antivirus, user behavior analysis, data loss prevention (DLP), device classification, and other functions. This integration makes perfect sense because the firewall is the cornerstone of security infrastructure.

Traditional Integration vs. Single-Pass Architecture

Numerous integration approaches have come about, including unified threat management (UTM), deep packet inspection, and others. These approaches share a common problem, however: they lack consistent and predictable performance when security services are enabled. Specifically, the base firewall functions can perform at high throughput and low latency, but when added security functions are enabled, performance decreases while latency increases. More importantly, these traditional approaches to integration limit security capabilities because a "sequence of functions" approach is inherently less flexible than one in which all functions share information and enforcement mechanisms.

Palo Alto Networks Single-Pass Architecture addresses these performance and flexibility challenges with a unique single-pass approach to packet processing, delivering better performance and security.

Better Performance Through a Single Scan

Single-Pass Architecture eliminates many redundant functions that plague traditional integration. As packets are processed, networking, policy lookup, application and decoding, and signature matching for all threats and content are performed only once. This significantly reduces the processing overhead required to perform multiple functions in one device. For content inspection and threat prevention, Single-Pass Architecture uses a stream-based, uniform signature-matching engine. Instead of using separate engines and signature sets (requiring multiple passes) or proxies (requiring download prior to scanning), it scans traffic for all signatures once, avoiding the introduction of latency.

Better Security Outcomes Through Context Sharing

Single-Pass Architecture supports superior security posture, relative to traditional integration, because the architecture performs full-stack inspection up front, and then makes all resulting context available to all security enforcement options (including threat prevention). This stands in contrast to traditional integration approaches in which full context is not shared between all enforcement options. Implemented in a variety of physical and software-based form factors, Palo Alto Networks Next-Generation Firewalls (NGFWs) based on Single-Pass Architecture are the high-performance foundation of a security platform that stops known and unknown modern threats.

Key Benefits of Integrated Security

Integrating key security functions into the firewall is not merely integration for integration's sake. Integration with our Single-Pass Architecture approach offers many benefits to any organization.

Reduced Network Complexity

Traditionally, every new security need meant deploying a new security device to solve it. As the number of security requirements increased, the number of devices deployed at key network junction points became unmanageable. Organizations no longer have enough data ports, port mirrors, network taps, rack space, or power to easily accept more devices in their networks. Moreover, in the past, this "device sprawl" happened in the internal network. Today, this has shifted to the cloud, and organizations are running into greater network complexity as their networks now span software as a service (SaaS), public and private clouds, containers, and more. Integration—if done well—starts to simplify the network.

Better Network Performance

Every new device introduces additional latency, throughput chokepoints, routing issues, and more. Well-done integration can reduce network latency and the number of chokepoints traffic must pass through.

Fewer Functional Holes

There are several basic pieces of information that are useful for setting security policy, irrespective of the function: source user or IP address, application, device, application function, URL category, port, protocol, and traffic destination. However, each separate device or scanning process acquires this information in a unique way or, in many cases, cannot acquire some of it at all. Such gaps and inconsistencies significantly impact security effectiveness. Well-done integration allows this information to be collected once and applied in a single, flexible set of security policies.

Simpler Operational Management

Managing a loosely interconnected set of devices is no simple task. Separate management systems, functional holes, unknown functional overlaps, and network complexity all contribute to higher costs and potentially ineffective network security. Well-done integration simplifies security management through fewer consoles and functional gaps, helping to provide more effective security coverage.

Lower Total Cost of Ownership

Purchasing separate devices for each functional security requirement, equipment maintenance, and operational expenditure all add significantly to your total cost of ownership (TCO). Well-done integration can significantly reduce these costs.

Problems with Traditional Approaches to Integration

Given the significant benefits of a well-integrated security ecosystem, the obvious question is: why have traditional integration attempts failed? Such attempts are largely flawed for two reasons. These glaring issues must be addressed for the benefits of integration to be achieved.

Flawed Traffic Classification

The traditional approach to security integration is to add functions on top of a foundational firewall. Many cloud providers offer firewall services based only on port/protocol (e.g., TCP/80), and this is essentially meaningless for today's applications, which often use nonstandard, non-unique, and/or dynamically selected ports. All further security functionality is then based on this flawed initial traffic classification.

Flawed Integration Methodology

Traditional integration attempts are based on collapsing multiple functions into one operating system and chassis. This isn't integration; it is consolidation, and the difference is critical. Consolidation simply takes multiple products and stuffs them into one device. In many cases, management and hardware are still separate, but there is an illusion of integration because the functions are performed in one device. In other cases, the functions all run on the same general-purpose CPU, draining system resources with each additional function activated.

Palo Alto Networks Single-Pass Architecture

It may be a seemingly obvious approach, but security software that looks at traffic in a single pass is unique to Palo Alto Networks NGFWs. This approach to traffic processing ensures that each task is performed only once on a set of traffic. Key processing functions are as follows:

- **Networking and management functionality** at the core of all traffic processing form a common networking foundation with a common management structure.
- **User-ID™ technology** maps IP addresses to (e.g., Active Directory) users and users to groups (roles) to enable visibility and policy enforcement by user and group.
- **Device-ID™ technology** maps IP addresses to specific devices to enable visibility and policy based on device identity.
- **App-ID™ technology** identifies applications with a combination of application signatures, protocol detection and decryption, protocol decoding, and heuristics. This is carried through to Content-ID to scan and inspect applications appropriate to their use, as well as to the policy engine and other services.
- **Content-ID™ technology** uses a uniform signature format to scan traffic for threats (exploits, viruses, spyware and malware communications) and sensitive data patterns (e.g., Social Security and credit card numbers, custom patterns). In addition to a signature-matching engine, the firewall dataplane applies machine learning on PowerShell scripts, portable executable (PE) files, JavaScript, and executable and linked format (ELF) files in real time to prevent file- and web-based attacks.
- **Policy engine** enables threat inspection and the decision to enforce forward/no forward to be done in a single policy. User-ID, Device-ID, App-ID, threat prevention, URL filtering, malware analysis, data loss prevention, and more can be written in a single policy.

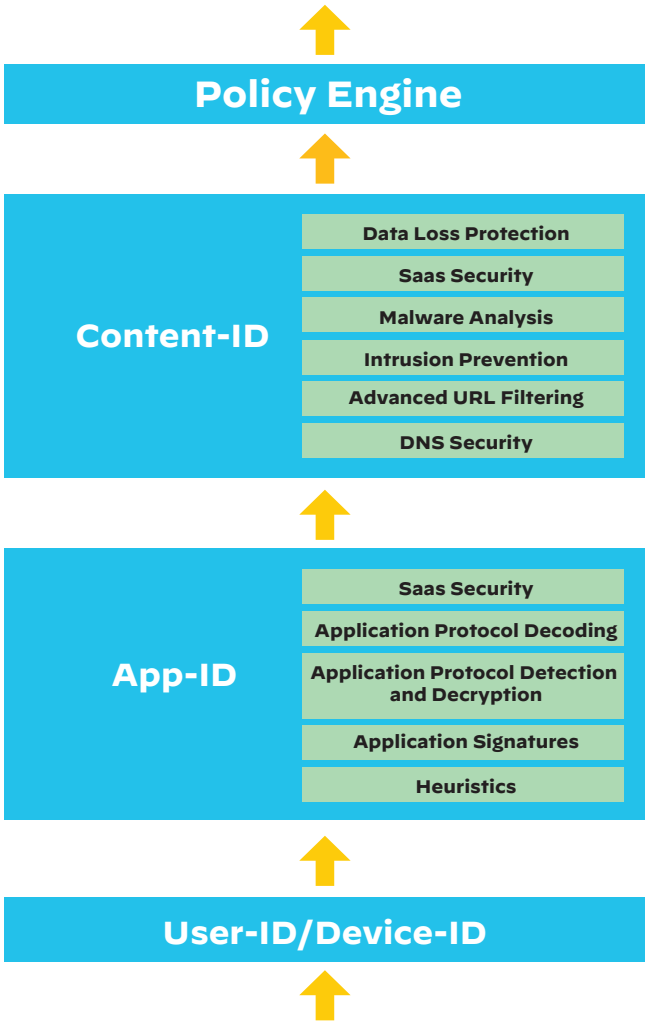

Figure 1: Key processes of our Single-Pass Architecture

Scan It All, Scan It Once

You can sum up our Single-Pass Architecture in a short phrase: "Scan it all, scan it once."

Common Protocol Decoding Engine

A key component of our Single-Pass Architecture is the use of a common protocol decoding engine for all traffic. The engine is used to pick apart an application stream to determine what the different pieces are (e.g., where a file transfer starts and stops, what the file type is, when the user is posting data vs. downloading, when a command is being executed). This information is then used as the basis for scanning the content for files, data, threats, and URLs. Performing the content scanning task only once saves significant processing power, as this is one of the most processing-intensive tasks for a security device to perform.

Visibility into All Ports, Protocols, and Applications Anywhere

Stream-based inspection does not rely on protocol or application-specific proxy services, which will be bypassed by unsupported protocols. As long as the firewall is inline, it will inspect all passing traffic, regardless of form factor. All Palo Alto Networks NGFWs, comprising the PA-Series (hardware), VM-Series (virtual), CN-Series (container), and Prisma® Access (cloud-delivered), leverage the same inspection engine powered by PAN-OS® and provide consistent security across the whole network real estate, whether in the data center, the branch, or private or public clouds.

Stream-Based Signature Engine

The use of a stream-based engine replaces several components commonly used in other solutions: a file proxy for data, virus, and spyware; a signature engine for vulnerability exploits; and an HTTP decoder for URL filtering. Using one common engine offers two key benefits:

1. Unlike file proxies that need to download the entire file before they can scan the traffic, a stream-based engine scans traffic in real time, reassembling packets only as needed and only in very small amounts.
2. Unlike traditional approaches, all traffic can be scanned with a single engine, instead of multiple scanning engines.

Pros and Cons of Stream-Based Scanning

It's important to understand the advantages and disadvantages of a stream-based scanning engine compared to a file proxy engine. The benefits of a stream-based engine are straightforward:

- **Scalability**: A stream-based engine requires significantly less memory and processing power since it doesn't need to store the entire file while it's downloading prior to scanning. Think of 5,000 users simultaneously downloading 5,000 different files and a file proxy trying to manage all of them—it just doesn't work. A stream-based engine scans the file downloads as they pass through, which is a much more feasible approach to scanning large amounts of data.
- **Low latency**: The stream-based engine processes and forwards the file as it receives it, scanning it with sub-millisecond latency unnoticed by the end user. File proxies, on the other hand, can introduce latency in the tens of seconds.
- **Common processing**: Using a stream-based engine enables one processing engine for all traffic, whereas a file proxy cannot scan for vulnerabilities and must therefore be part of a multi-pass approach.

Some key trade-offs with a stream-based engine should still be considered:

- **SMTP/POP3/IMAP limitations**: Stream-based engines work very well for most applications, but not for blocking viruses, spyware, or data over traditional email protocols, such as SMTP. While alerting works well, without actually proxying the connection, blocking such attachments within an email message will often cause a continuous retransmission of the attachment over SMTP. In addition, it is not possible to quarantine the email message. Usually, this is not a problem, as the email server is already surrounded by one or more layers of antivirus. This does not apply for SaaS services like Microsoft 365™ or Gmail®, as SaaS Security can help protect these environments.
- **Scannable compressed formats limited to ZIP and GZIP (without password encryption)**: These are the only compression formats that compress in blocks of data, instead of compressing the entire file as one block. This is typically not a problem, as these are the most common compression algorithms, and this is supplemented with file type scanning and alerting so that other file types can be monitored and potentially blocked from traversing certain network segments or applications.

Keeping the goal of integration and performance in mind, Palo Alto Networks chose to implement a stream-based scanning engine.

Single-Pass vs. Multi-Pass Architecture

The initial comparison to providing multiple security functions in discrete devices is obvious—each of the described blocks in single-pass architecture will be performed by each device (assuming they can perform all the functions). The duplication of processing is staggering in this case. Additionally, existing attempts to integrate security functions into a single device are often only "sheet metal" integration, where the networking and management functions are integrated but elements of traffic classification, protocol decoding, file proxying, and signature matching are performed with separate software and sometimes separate hardware. Figure 2 shows a worst-case view of discrete devices with a multi-pass approach.

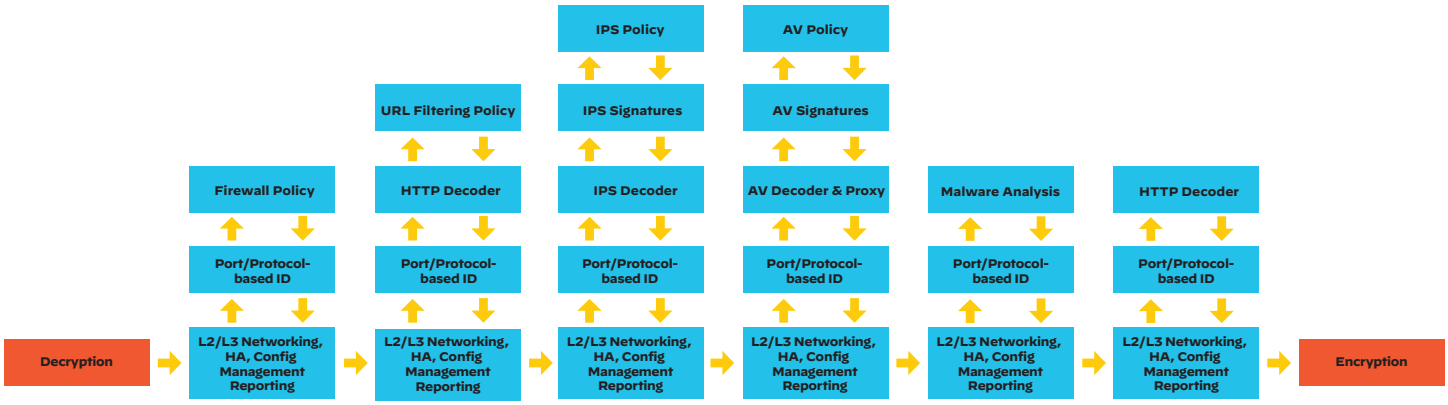

Figure 2: Processes of a multi-pass approach

Figure 2 assumes there are discrete devices performing each function, which results in multiple passes through the networking layer, traffic classification, decoders, signature engines, and policy tables. Additionally, with most of today's traffic being encrypted, effective security requires decryption services for each of the devices. Each pass generates processing overhead, latency, throughput degradation, and operational costs to keep it all functioning. Some basic cost saving is often achieved by collapsing the networking layer and port/protocol identification into a single pass, but most of the heavy lifting—including file proxies, application decoding, signature engines, and policy enforcement—is often still separate functions with overhead that compete for shared processing.

Conclusion

As the number of required security functions continues to increase, there are two options: add another security device or security service; or add a new function to an existing device. With our Single-Pass Architecture, Palo Alto Networks makes it possible to add a function to an NGFW instead of adding another security device. Our integrated approach offers benefits and advantages that discrete devices cannot. Discrete devices will still be necessary where highly specialized functionality is required, but in most cases, integrated security is now a viable option.

paloalto NETWORKS

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