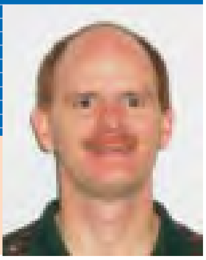
SOFTWARE CORNER

Software, IP Security, and Silicon Acceleration: Part I

By Curtis A. Schwaderer

CompactPCI Systems



Networking equipment solutions are evolving from single-purpose devices into systems that incorporate multiple functions into a single box or blade. Ever-increasing computing power coupled with declining component costs have been a trademark of the technology industry since its inception and the security space seems to be no exception. A two part discussion focusing on examining security software architecture, discuss and identify possible performance bottlenecks, and then look at how one security silicon company is accelerating in these areas to achieve wire-speed solutions for gigabit bandwidths and beyond. Part one of this series will examine comprehensive traditional security software architecture and discuss the "hot spot" areas where performance bottlenecks potentially occur. The second part of the series will focus on security silicon acceleration and how these components interface with security software and accelerate potential performance bottleneck areas.

Security software architecture

An example of IP security software architecture with associated data, control, and management paths is shown in Figure 1. The three management paths in the figure are labeled with a number in a circle (circle 1, circle 2, and circle 3).

The management and configuration path is shown in Figure 1 as path 1. This path sets up the initial security policies that are stored in the Security Policy Database (SPD). Indexed by selector fields, a security policy database (SPD) consists of the following:

- IP source and destination addresses, name (user name or system ID)
- Data sensitivity level (not typically used)
- Transport layer protocol
- Source ports
- Destination ports

Also, wildcarding and ranges for the IP addresses are acceptable as well; the IP addresses do not have to be specific. This non-specificity makes it convenient to identify a single policy that covers a large number of clients, making the searching and matching of the policies against the incoming packet a bit more challenging (this feeds into the performance issue described later in this column). Similar to the IP addresses, the layer 4 port selectors can also be wildcards or ranges.

The control plane of the software architecture is represented as path 2. The control plane provides automatic key exchange between security gateways or secured clients. A negotiated set of keys for a given secure connection, called a Security Association (SA), describes a unidirectional flow of a single secure connection. A bidirectional connection therefore will have at least two SAs associated with it. Once the keys are negotiated, the key exchange protocol works with the policy management block to store the security associations in the Security Association Database (SAD). Like the SPD, security association entries have the same kind of selectors associated with them and most SAs can also contain wildcards or ranges. When packets matching a par-

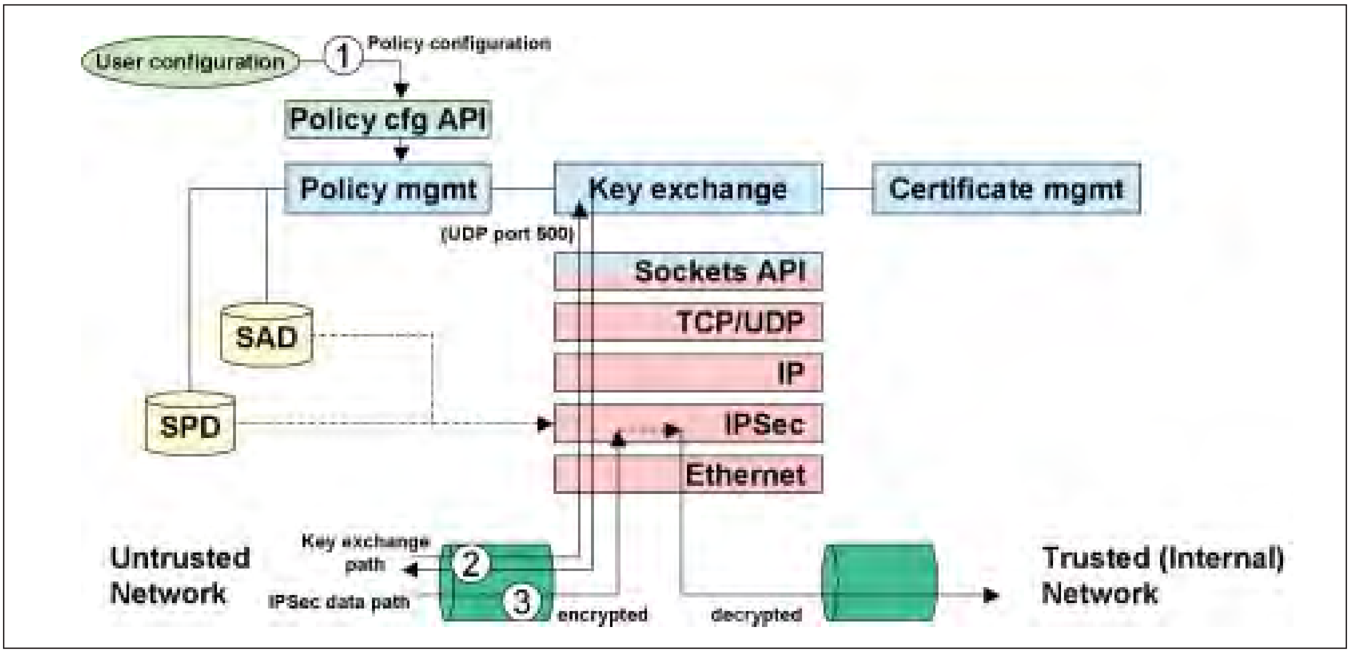


Figure 1

©Copyright 2003

ticular security policy invoke a number of security associations, an *SA bundle* develops. A packet may provide authentication using Authentication Header (AH) encapsulation and encryption services are performed using an Encryption Security Payload (ESP). This scenario would result in a packet wrapped with an AH further wrapped in an ESP header. The *SA bundle* for each direction of this connection would consist of an SA referencing AH information, and another SA referencing ESP information. An SA can contain sequence number counters and anti-replay windows (to prevent replay attacks), sequence counter overflow, lifetime and lifebytes, the crypto keys, and other miscellaneous values.

The data plane path, path 3, consists of multiple steps within the security architecture. First, the incoming packet is matched against a security policy. Now, the security policy can dictate that the packet is passed as is, dropped, or request further security processing be performed. Passing and dropping of the packet is part of access control, which is discussed later. If further security processing is needed, this will involve processing security headers, decrypting payloads for in-bound packets, and encrypting and adding security headers for outbound packets. For IPSec encapsulated packets, a Security Parameter Index (SPI) is found just after the IP header where the source and destination ports would normally be. This SPI is assigned when the key exchange occurs and the secure connection is established. The SPI, along with the destination IP address and transport protocol uniquely identify the security association belonging to the connection.

This security architecture is typically implemented in software using a general purpose CPU and, depending on the application, each of these three paths will include performance bottlenecks that will limit the speed of the solution. In addition to data, control, and management plane considerations, there are also a variety of layers at which security processing is performed.

Virtual Private Networks (VPNs) and firewalls represent two mainstream applications of this security architecture. Initially offered in two distinct single-purpose devices, these functions are now typically combined into single VPN/firewall devices. Intelligent line cards and routers are also displaying the functionality of VPNs and firewalls. This phenomenon is an example of the multi-function evolution in the networking world and also causes concern over how these functions can push out adequate performance running on a single-computer environment.

Encryption and authentication

There are a variety of Internet Engineering Task Force (IETF) RFCs that cover security. The most fundamental of these specifications is RFC 2401, Security Architecture for IP. This RFC covers encryption and authentication services at the IP and TCP/UDP layers. The encryption and decryption services occur from the IP or TCP/UDP header above, and the resulting packet is further encapsulated with an additional IP header resulting in security services at all layers of the stack. There are two main security protocols that provide security services at the IP layer, Authentication Header, specified by RFC 2402, and ESP, covered by RFC 2406.

A typical security topology illustrating the two different kinds of IPSec crypto services available is shown in Figure 2. The transport mode security occurs at layers four and above, and the IP header of the client and server travel the network unencrypted. Now, while the services being used and the payload are encrypted, a potential hacker still may have insight into the members of the communication session. Using tunnel mode, multiple host and client sessions are encrypted and muxed into one large encrypted "tunnel" between two devices called security gateways. The encryption occurs at the IP layer and above, resulting in the hacker only having insight into the two communicating security gateways, but not the clients and servers involved.

Encryption and authentication represent the two primary features of a VPN. A typical application of a VPN would be a remote client who desires access to corporate networks without worrying about observers on the network grabbing the information as it travels the greater Internet.

Access control

Access control involves the creation of *access control lists* (ACLs) between two interfaces in the network and dictating whether each of those entries is allowed to pass through or be dropped. Access control lists can be thousands of entries long and describe rules for using information at the IP or TCP/UDP layers. A common attack is to identify a client IP address within the trusted network, attempt to use that IP address from the outside to gain access to an internal machine. This kind of threat is called *spoofing*. To counter *spoofing*, a user might write an ACL rule that drops a packet if an internal network IP address comes in as the source IP address from the external network.

Access control is a primary feature of firewalls. Firewalls can cause a bottleneck in the system since a given ACL can have thousands of entries to cycle through while packets are arriving at the firewall at wire speeds on a gigabit link. This implies the firewall has to be able to apply the ACL rules to the packet extremely fast or risk a large disruption in packet flow.

Securing parts of the payload

Secured Sockets Layer (SSL) is another related technology used to secure portions of a session that are critical to being secure. SSL provides consumers with the ability to casually browse an on-line store using clear packets that provide high performance and response. Then, when consumers add items to their shopping cart and proceed to the purchasing screen, the SSL technology kicks in and encrypts sensitive information such as credit card numbers and personal information.

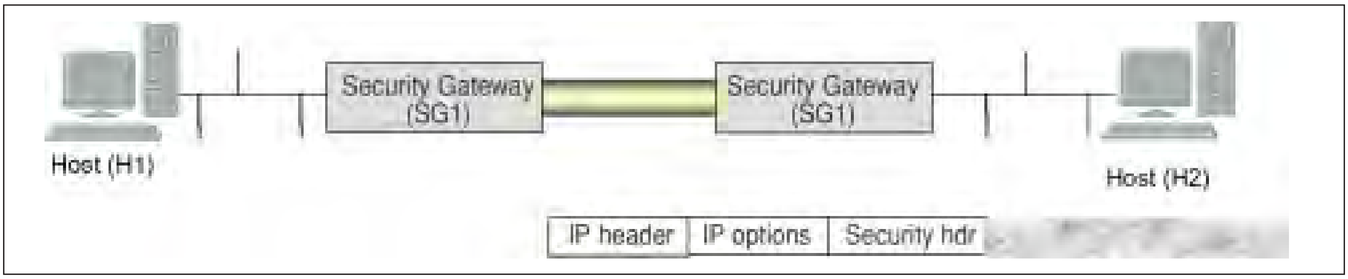


Figure 2

©Copyright 2003

Intrusion detection and mitigation, virus detection, and removal

Another form of a layer 7-security threat involves making various attempts at gaining access to unauthorized locations by embedding information or executables into the payload of services that are allowed access into the internal network. Once a trusted user provides a thread of execution, the payload has all the permissions needed to perform malicious acts. Commonly, virus detection software scans payloads for specific strings that identify viruses or intrusion attempts. However, the intrusion signatures can be broken between packets or contain embedded control characters to make intrusion signatures harder to detect. Security services using virus detection and intrusion detection must scan the entire packet, and even across packet boundaries, making this kind of security service a liability to system performance.

Controlling security processing

As access control, encryption, and authentication converge into multi-function VPN/firewall devices, the security architecture for IP RFC helped define the concept of a security policy and security associations. Recall that security policies include the ability to pass, drop, or provide IPSec services for a given packet matching a rule in the SPD, where the SPD policies come from corporate system administrators or network service providers. These policies may be very broad based, and literally thousands of IP addresses may be in the list to provide security services. In order to provide security services, the device needs to have crypto keys to perform encryption and decryption functionality. These crypto keys can be manually provisioned or automatically obtained through another set of RFCs relating to Internet Key Exchange (IKE). Two notable RFCs relating to IKE are RFC 2408, Internet Security Association and Management Protocol (ISAKMP) and RFC 2409, the Internet Key Exchange Protocol (IKE).

The algorithm for automatic key exchange (for example, the circle 2 path in Figure 1) can be described as follows:

- A packet arrives on an interface, is looked up in the security policy database, and a matching rule is found that says the packet should have IPSec services applied to it.
- The VPN/firewall looks up the packet in the SAD where the encryption keys are stored for active sessions and no matching security association is found.
- The Internet Key Exchange (IKE) entity is notified and using the information in the security policy, initiates a session with an IKE peer to exchange keys for the session.
- Eventually the keys are passed between the devices and stored in the SAD. The packet that started the key exchange now gets encrypted and passed to the outbound interface for transmission.

Bottlenecks in the security solution

With the functionality of each component of the security system taken into consideration, analysis of the potential problems can be done. As security functions find their way into devices performing other functions, these combined functions begin to compete for limited computing and memory bandwidth.

The potential bottlenecks and where they occur can be seen in Figure 3 and are labeled A, B, and C.

- Bottleneck A, the policy management bottleneck, is typically considered an *out-of-band* kind of activity that involves the creation of policies and rules that are performed during initial provisioning of the box. Since provisioning of boxes is a relatively infrequent activity occurring at the beginning of deployment, it is not considered to be time critical. This being considered, further detail on performance acceleration options for provisioning of security policies is not necessary. However, there is certainly a complexity issue that arises. Thousands of rules and pass/drop/IPSec decisions can be complex to manage and maintain, and organizing these rules, using graphical interfaces and rules languages is helpful. Furthermore, control languages that help reduce provisioning

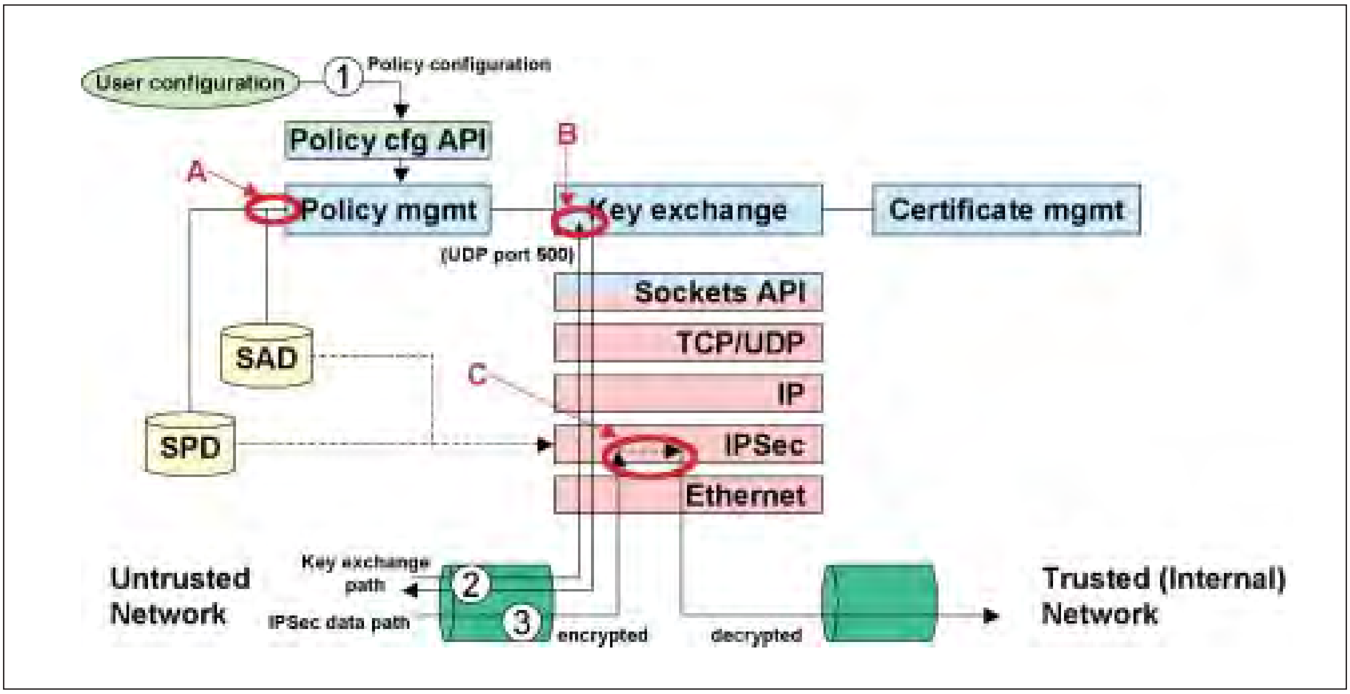


Figure 3

are going to be needed in the future, especially when combining features for a multi-function device.

- The key exchange bottleneck, B, represents a primary area of performance concern. Performance and a decrease in system capacity can be the result of the rapid exchange of multiple packets between security gateways, which is due to the large number of short-lived connections requiring key exchanges.
- The processing bottleneck, C, represents IPSec processing in the form of the initial security header processing, and the actual encryption/decryption of the packet itself. The processing of the security header has functions that analyze security headers, perform SA lookups, and ensure anti-replay attacks are not present. Other functions are cycled during this phase as well, but the aforementioned functions are the primary causes of performance concern. The actual encryption/decryption of the packet uses a variety of algorithms that involve multiple iterations/memory access calls, etc. As a result, encryption/decryption can be extremely slow for software running on a general purpose CPU to execute.

IKEs per second

IKEs per second represent another bottleneck in the security system architecture. IKEs represent, for short-lived connections, how many key exchanges per second can the system do, while keeping up with traffic. Key exchange tends to be time consuming, involving multiple messages passed back and forth before the security association is ready to be used. To improve upon this, some software algorithm acceleration has been completed that will pass keys for multiple sessions within a single key exchange session. For perspective, the *hundreds of megabits per second* class of requirements for IKE setup and teardown may be 1000 to 5000 key exchanges per second and for gigabit class performance, IKEs per second may reach over 10,000.

SSL acceleration

With SSL, a mixture of performance bottlenecks involving both sides of SSL technology. The switching between standard and SSL mode of operation and the encryption/decryption of the payload being sent securely represent significant performance issues. The encrypted payload for SSL is relatively small compared to the messaging that takes place to establish the SSL session, therefore SSL acceleration must focus on accelerating the SSL establishment, then have acceleration for encryption and decryption if needed.

IPSec processing

The encryption and decryption of the packet payload represent the primary performance bottleneck of the IPSec processing. The IPSec processing is probably the biggest concern within the entire security software architecture. Presently, silicon that performs gigabit Ethernet encryption and decryption is the focus of many security companies. In addition to the encryption/decryption functions of IPSec, ESP and AH header processing and looking up security associations can also affect performance. In wireless networks there may be up to a million users flowing through the security equipment, matching security policies and security association lookups can drop the connection bandwidth below acceptable limits, implying acceleration of these processes is also needed. Wireless network connections tend to be primarily voice traffic and consume little bandwidth even though there may be millions of users, implying a small issue of concern. However, as Internet over cable continues to grow, the cable network operators are faced with the same problem, only with higher bandwidth video and data applications. Therefore, acceleration in the header processing and SA lookup are two important areas of research for alleviating IPSec processing performance bottlenecks.

Conclusion

This part of the series has focused primarily on an introduction to the security software architecture, functionality of the components, and performance critical areas. The next part of the series will focus on a number of securities acceleration silicon solutions, how they interface with the software architecture, and how much acceleration can be expected with a silicon-assisted architecture.

The second part of this two part series, appearing next month, will focus on security silicon acceleration and how these components interface with the software and accelerate potential performance bottleneck areas.

©Copyright 2003