Securing LANs at Wire-Speed
An Architectural Approach

Table of Contents

Executive Summary

Enterprise LANs have become more complex and vulnerable in recent years. The increasingly distributed enterprise and growing mobile workforce has accelerated a blurring of the traditional LAN perimeter and made the LAN vulnerable to attacks from within, such as malicious user activities and the propagation of worms. In addition, client-server application vulnerabilities continue to introduce exposures to the integrity of data and network availability. These conditions have created the need for per-port, inline, wire-speed, stateful LAN security solutions that can be deployed inside the LAN to protect it from new threats and isolate existing threats at the source.

Protecting the LAN from inside threats requires both proactive and reactive security measures. Proactive measures are applied before an endpoint or user is granted access to the network, while reactive measures are used to detect and isolate threats. An inline LAN security solution must provide both proactive and reactive security measures without sacrificing wire-speed performance.

Threats have become blended and increasingly sophisticated. Security at the Internet-LAN perimeter is no longer adequate protection with a mobile workforce, wireless access and 3rd parties accessing the network. These developments have driven the need for a layered approach to security that employs multiple threat detection and prevention techniques before, during and after any user attempts access to the networks. This includes endpoint integrity verification, network authentication and access control, application-layer firewall, signature-based intrusion prevention, anomaly-based threat detection, and others. LAN security solutions need to be capable of performing deep packet inspection on 100 percent of LAN traffic to deliver comprehensive LAN security in this sophisticated threat environment.

There has been tremendous growth in business critical real-time applications, such as voice-over-IP (VoIP) and streaming video which cannot tolerate latency. Considering this growth, LAN security solutions now need to be very low latency—flexible and scalable to protect against evolving threats that impact application performance.

In summary, today's LAN security solutions will need to provide multiple security techniques, each capable of deep packet inspection at full wire-speed for 100 percent of traffic. These solutions will either be deployed as inline transparent appliances between LAN switches, or they will be absorbed into optimized LAN switches to offer per-port LAN security. The inline appliance approach can be easily incorporated into existing LANs and provides a path for integrating security into the network fabric.

A New Class of Silicon to Enable Wire-Speed LAN Security

Traditionally, LAN switches have been built using Layer 2/3 switching ASICs. These ASICs implement standard Layer 2/3 packet processing and traffic management functionality in hardware state machines and pipelines. While they afford some amount of configurability, these ASICs seldom provide programmability. Their fixed-function nature enables them to provide deterministic wire-speed throughput, irrespective of packet size. But the lack of programmability and stateful awareness makes them too inflexible for today's LAN security solution.

General-purpose processors offer the required programmability, but they have other limitations, such as inadequate instructions-per-second and DRAM bandwidth. More importantly, these generic processors are simply not designed to move packets at multigigabit speeds.

In recent years, network processors have emerged as the silicon architecture of choice for networking systems that require programmable packet processing. Network processors typically employ multiple processors or micro-engines to exploit the inherent parallelism in packet processing. They also provide hardware acceleration for packet movement at multigigabit speeds, Layer 2/3 table lookups, queuing, quality of service (QoS), and so on—but they usually lack hardware acceleration for security processing.

Specialized security coprocessors have become available from merchant silicon vendors. These coprocessors typically accelerate one type of security functionality, such as cryptography or regular expression matching. However, they rely on another type of silicon, typically a network processor, to provide them with the packet stream upon which to operate.

None of these options—ASICs, general-purpose processors, network processors, security coprocessors—can by itself address the peculiar challenges of comprehensively securing the LAN. One could conceivably build a LAN security solution by cobbling together a mix of these products, but it would be nearly impossible for such a solution to meet the stringent performance requirements—wirespeed 10-Gbps throughput and switch-like latencies—that are required for today's secure LAN.

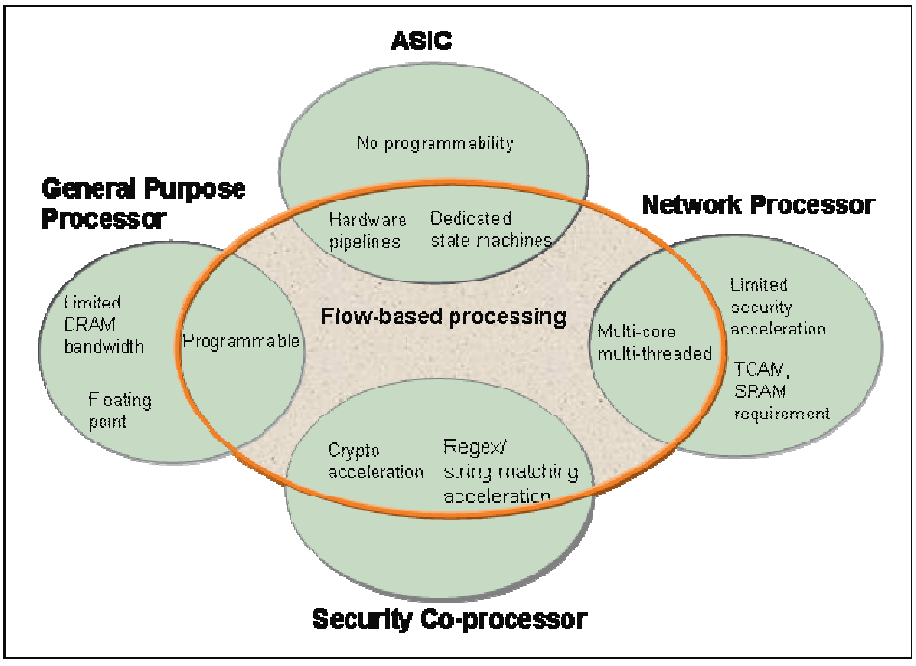

Figure 1: New Class of Silicon Required for 10-Gbps LAN Security

Clearly, what is needed is a new class of silicon architecture, as illustrated in Figure 1. This new class must incorporate the deterministic throughput and latency benefits of an ASIC, the programmability of a generic processor, the programmable packet

processing capabilities of a multicore, multithreaded network processor, and the security acceleration of security coprocessors. This new class of silicon needs to be much more than a mechanical integration of different architectures, because its real advantage will come from the architectural integration of the multiple security services running on this silicon, and the correlation between these services.

Flow-Based Packet Processing Architecture

A silicon architecture that can meet the functionality and performance requirements for multigigabit LAN security is shown in Figure 2. The architecture works in this way:

Step 1: Incoming packets go through ingress processing at 10-Gbps.

Step 2: Packets are sprayed across all idle processor threads.

Step 3: Multibank Layer 2 cache and interleaved DRAM allow all processor threads to process packets concurrently.

Step 4: Threads offload compute-intensive tasks to hardware accelerators (counters, queues, free lists, packet DMA, and so on) and security accelerators.

Step 5: Outgoing packets pass through egress processing at 10 Gbps.

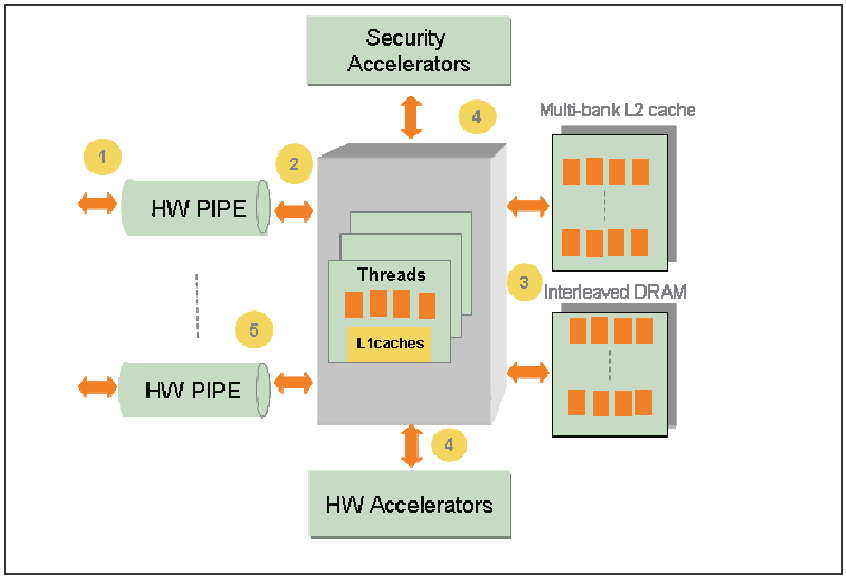

Figure 2: *New Silicon Architecture*

At the heart of this silicon is a multicore, multithreaded processor complex that provides the required programmable horsepower. This processor complex uses hardware accelerator engines to offload compute-intensive tasks. The Layer 2 cache and external DRAM are multibanked to allow high throughput at low latencies.

All packet movement and traffic management is executed in hardware pipelines that provide deterministic throughput. Security processing also is hardware-accelerated to allow the architecture to provide 10-Gbps of throughput while inspecting 100 percent of the packets.

Delivering Deterministic 10-Gbps Performance

The inline LAN security solution needs to provide 10-Gbps throughput and end-to-end latencies of less than 100-microseconds, while deeply inspecting each packet. Several factors affect throughput and end-to-end latency.

Foremost among these factors is the amount of raw or peak processing capacity, which is determined by the number of processors and the frequency at which the processors run. This capacity is measured in terms of billions of instructions per second (BIPS).

A second factor is the amount of effective processing capacity, which is a measure of the utilization at which the processor pipeline operates while processing packets at the full load of 10 Gbps. The utilization of the processor pipeline is determined by the pipeline's basic micro-architecture—including the pipeline stages, latencies of Layer 1 instructions and data cache accesses, multiported register files, and branch prediction techniques—and by the latencies to shared resources such as Layer 2 cache and external DRAM. Multithreaded processors allow the processor pipelines to be used at very high utilizations by effectively hiding the latencies to such shared resources.

The multicore, multithreaded architecture is well suited for packet processing applications because of the inherent parallelism that allows multiple packets to be processed concurrently. While the same is true for most network processors when processing at Layers 2, 3, and 4 on a packet-by-packet basis, it is not the case for flow-based stateful LAN security processing. One approach to solving this problem is to bind all packets in a given flow to a specific processor core. Another approach—this one offering higher-performance—is to allow the packets in a particular flow to be concurrently processed by multiple cores, while using granular semaphores to maintain flow state consistency.

Multithreading attempts to overlap the shared resource latencies of one thread with the execution in the processor pipeline of other threads. To achieve high utilization in the processor pipeline—and thus an effective processing BIPS that is close to peak—shared resource latencies must be optimized. As shown in Figure 3, there are numerous shared resources that the processors need to access when doing deep packet inspection, including packet headers and buffers in on-chip cache or external DRAM, flow state and other data-structures in on-chip Layer 2 cache or external DRAM, on-chip accelerators such as DMA engines, and security acceleration cores. Any of these shared resources can become a performance bottleneck if it is not carefully architected.

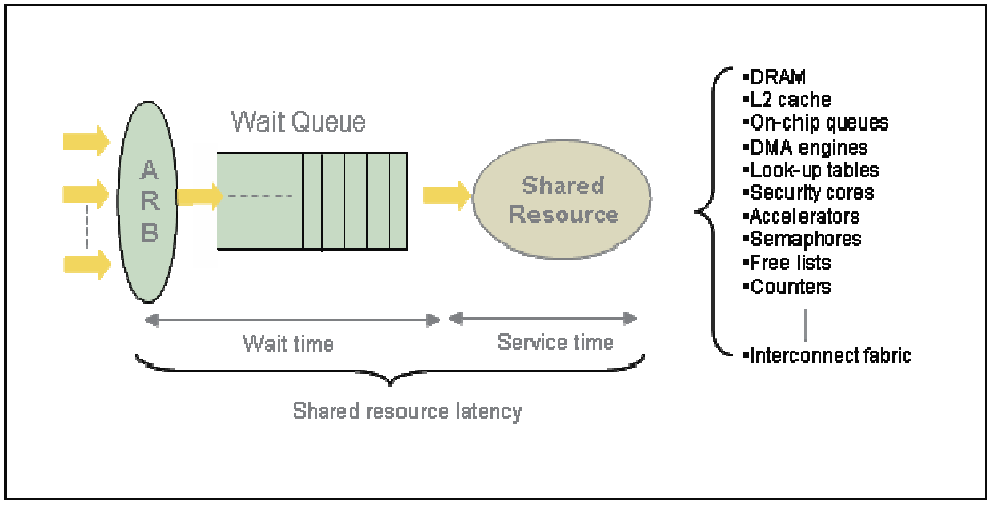

Figure 3: Shared Resources

The most important factor that determines the shared resource latency is the utilization level at which the shared resource is operating. Figure 4 shows that each shared resource needs to be operated at a level below 70 percent utilization to keep these latencies well bounded.

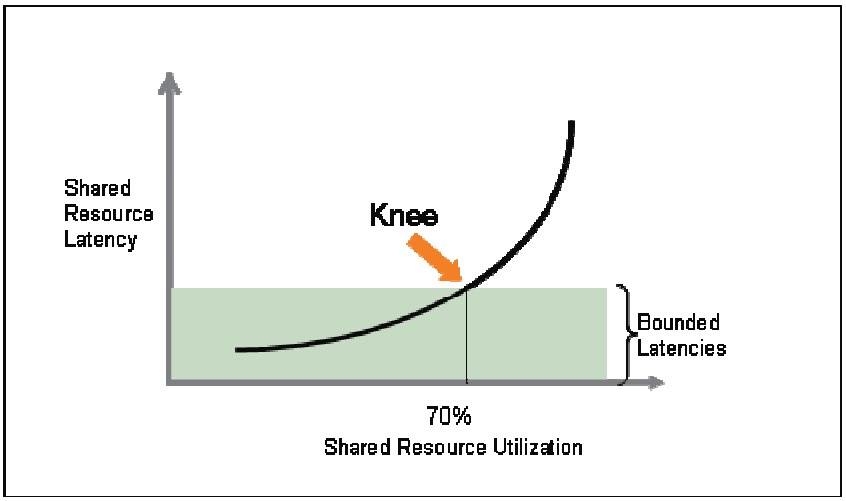

Figure 4: Shared Resource Latencies

One such shared resource is the interconnect fabric that provides the connectivity between the processor cores, the shared Layer 2 cache, the external DRAM, and all the hardware accelerators. Quite often, to optimize the silicon area taken up by the interconnect and to reduce complexity, the interconnect is implemented as a shared bus, or as a variation on this idea, such as a shared ring bus. Unfortunately this approach becomes a huge performance bottleneck, because it affects the latencies for all shared resources.

An alternative interconnect architecture, as shown in Figure 5, uses point-to-point and multi-point links to alleviate this bottleneck. By providing bandwidth in excess of 1 terabit per second from processors to shared resources, it ensures that interconnect will always operate at very low utilizations, keeping the latencies well-bounded.

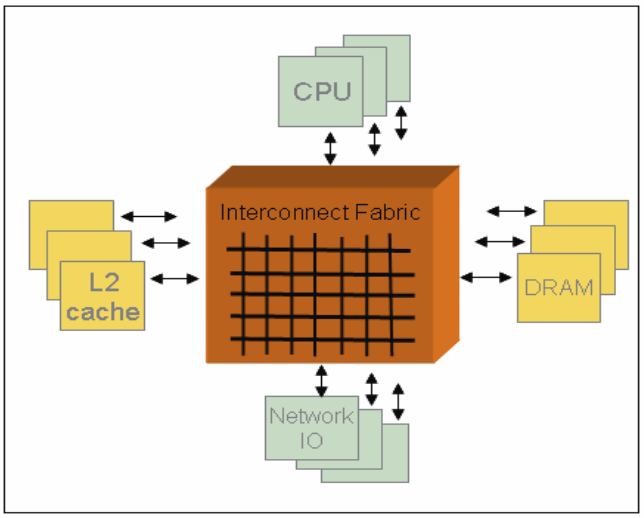

Figure 5: Interconnect Architecture for Deterministic Throughput at 10 Gbps

Another factor that can significantly affect performance is the instruction set efficiency. Packet and security processing applications have special requirements that can hinder performance. Proprietary Industry Standard Architecture (ISA) extensions, such as pre-fetching packet buffers and flow state, bit-wise operations, and offloading to special hardware coprocessors, are required to improve the efficiency of the instructions. Compute-and latency-intensive tasks, such as atomic counter updates, free list management, queue management, packet movement from on-chip to off-chip memory and vice-versa, and hash-table lookups, can all be offloaded to hardware engines, thus freeing up the processor BIPS to handle packet processing functionality that must be performed in a programmable way.

One of the key challenges in a programmable architecture is being able to move packets efficiently at the desired throughput. The internal movement of packet data on the silicon needs to be completely offloaded to hardware pipelines and state machines. This

offloading must be complemented with queuing, bandwidth and congestion management (such as Deficit Round Robin and weighted random early detection), and other traffic management algorithms. All of these techniques must be handled by hardware accelerators that can deliver deterministic throughput.

As a final challenge, 10 Gbps of encryption and authentication throughput for IP Security (IPSec) and Secure Socket Layer (SSL) applications requires employing parallel hardware accelerators that implement the standard cryptographic algorithms, such as Advanced Encryption Standard (AEA) and Secure Hash Algorithm (SHA-1). Some amount of programmability to control block-by-block movement in these accelerator cores is important to allow future adaptability—and their architecture and design should accordingly ensure that it can supply and drain 10 Gbps throughput to and from the accelerator cores. Similarly, 10 Gbps of signature-based threat containment can be provided only by employing regular expression or string-matching hardware accelerators. Given the instruction-intensive nature of doing pattern matching, implementing it in software running on the processor cores can result in nearly a 10X degradation of performance, as shown in Figure 6. It is important to note that even with 32 CPUs (128 threads), one cannot achieve secure switching throughput close to 10 Gbps (even at 750 MHz) unless the CPUs are complemented by purpose-built security hardware accelerators.

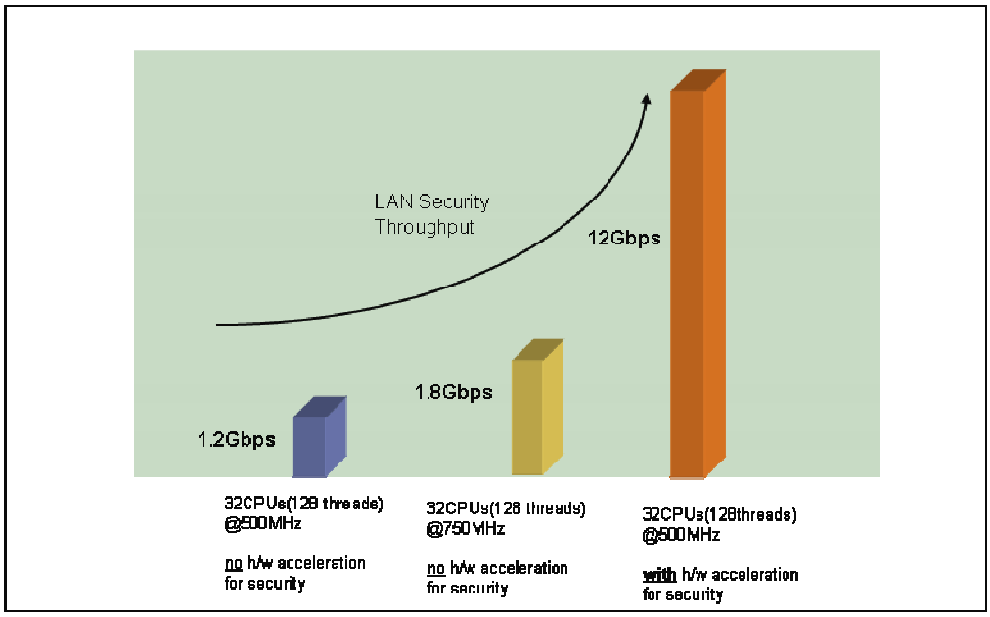

Figure 6: Security Processing—How Hardware Acceleration Impacts Throughput

Silicon Architecture Requirements Summary

In order to provide the necessary processing power for in-depth security functionality, silicon architecture in switches need to be designed to provide:

1. Hardware pipelines for deterministic packet movement throughput
2. Multiple multithreaded programmable processor cores
3. Hardware acceleration engines

Additionally, several factors affect the performance of systems built using such programmable silicon architecture. Specifically, these include:

- Raw processing power, measured in terms of the peak and effective BIPS of the processor cores (billion instructions per second), as determined by the number of processor cores, frequency of operation, single- or dual-issue pipeline architecture, utilization of the CPU pipelines, single vs. multithreaded processor cores, branch prediction, and so on.
- Instruction efficiency, as determined by the instruction set architecture and specialized extensions for specific networking and security processing.
- Raw throughput of acceleration engines used by the processor cores to offload compute-intensive tasks, such as hash-map table operations, checksums, atomic counter updates, free-lists, on-chip queues, and others.
- Memory architecture—the size and bandwidth of on-chip and off-chip memory, the efficiency of off-chip memory channels, and partitioning of key data structures across on-chip and off-chip memory.

The key to deterministic performance is to ensure that each resource on the silicon, including the on-chip interconnect that connects all the resources, has enough headroom to keep the access latencies well bounded and thus keep up with packet processing at wire speed.

Conclusion

Today's LAN security systems requires a new class of silicon that can deliver 10 Gbps throughput for multiple security services, such as application-aware stateful firewall, encryption and authentication, and blended threat containment to combat sophisticated attacks. Since no one technique will be sufficient to deliver the high level of performance—high throughput and low latencies—required in LAN deployments, this silicon will need to employ multiple techniques, including multithreading, smart hardware offload engines, and security-specific acceleration.

About the Author

Manish Muthal, Cofounder, Director of ASIC Development

Manish Muthal brings over 12 years of experience in packet processing, traffic management, switching fabrics and high-end processor design in ASIC and hardware architecture. Muthal joined Nevis Networks from Tenaya Networks, where he was Cofounder and Manager of ASIC Development. Preceding that role, Muthal was Senior ASIC Engineer at Juniper Networks, leading the design for the route-lookup module of Internet Processor3 ASIC. His other achievements include Manager of ASIC Development at Amber Networks (acquired by Nokia), where he helped develop a multimillion gate ASIC that integrated packet processing and traffic management for Internet edge routers. Muthal's career began at Intel, where he worked on platform architecture for high-end SMP servers and the design and management of core-logic chipsets for PC platforms. He graduated with a BS in Electronics & Telecommunications from VNIT, India and an MS in Computer Engineering, University of California, Santa Barbara.

About Nevis Networks

Nevis Networks provides innovative ASIC-based LAN security systems designed to help corporations protect information privacy and integrity, ensure network availability, and maintain regulatory compliance. With its patent-pending LANsecure™ architecture, the Nevis LANenforcer product family integrates NAC with the deepest threat containment at wirespeed to create a "Personal DMZ" around every user on the LAN. Nevis was founded in 2002 by seasoned executives with strong track records in security, semiconductor design, and networking technologies, and has raised over $40 million from veteran Silicon Valley investors New Enterprise Associates, BlueRun Ventures, and New Path Ventures. The company is headquartered in Mountain View, California, with additional R&D centers in Pune, India and Beijing, China.

Nevis Networks
500 N. Bernardo Avenue
Mountain View, CA 94043
T: 650-254-2500
F: 650-254-2555
www.nevisnetworks.com