Integrated Acceleration Techniques for Security Appliance Software

Srinivasa Rao Addepalli, Chief Software Architect
Software Products, Freescale Networking and Multimedia Group

April 8, 2009 - Linley Tech Seminar: Embedded Network Security Design
Outline

► Security Appliance Software Components
► Need for Higher Processing Requirements
► Offload Requirements
► Offload Functions
► Freescale Multicore Product Family
### Typical Software Components of a Security Appliance

<table>
<thead>
<tr>
<th>Software Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stateful Firewall with NAT</td>
<td>▶ Controlled access to network resources. Network address translation</td>
</tr>
<tr>
<td>IPSec VPN</td>
<td>▶ Confidentiality, Authentication and Integrity for traffic between networks. Secure Remote Access</td>
</tr>
<tr>
<td>SSLVPN</td>
<td>▶ Secure Remote Access through a browser</td>
</tr>
<tr>
<td>IDS and IPS</td>
<td>▶ Detect and prevent Intrusions at L4-L7 and application level</td>
</tr>
<tr>
<td>Application Traffic Throttling</td>
<td>▶ Detect and throttle less-priority application traffic (e.g. P2P, IM)</td>
</tr>
<tr>
<td>Network Anti-Virus</td>
<td>▶ Stop virus infected payloads and malware from crossing the perimeter (e.g. emails, HTTP, FTP)</td>
</tr>
<tr>
<td>Application Firewall (HTTP/SIP)</td>
<td>▶ Stop attacks/intrusions using deep data inspection of HTTP/SSL/compressed payloads</td>
</tr>
<tr>
<td>L4-L7 Load Balancer (ADC)</td>
<td>▶ Distribute load across multiple servers.</td>
</tr>
<tr>
<td>Traffic Policing &amp; Shaping</td>
<td>▶ Enforce QoS policies on network/application traffic</td>
</tr>
<tr>
<td>Virtualization (Data Center)</td>
<td>▶ Support multiple virtual security appliances within single hardware. Instances mapped to customers</td>
</tr>
</tbody>
</table>
Need for Enhanced Hardware Acceleration

**Higher Throughput Requirements**

- Increased deployment of security appliances in Enterprise Core
- Increased deployment of security functionality by Carriers
- Security applications making way into Data Center Application Delivery Controllers

<table>
<thead>
<tr>
<th>Deep Packet &amp; Data Inspection (DPI/DDI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Need for Protocol Analysis</td>
</tr>
<tr>
<td>▶ HTTP, FTP, SMTP, SIP, SNMP etc.</td>
</tr>
<tr>
<td>Content Format Complexity</td>
</tr>
<tr>
<td>▶ Various file formats, HTML/Javascript analysis, XML analysis</td>
</tr>
<tr>
<td>Patterns Number &amp; Complexity</td>
</tr>
<tr>
<td>▶ 7000 IPS patterns. 500,000+ ClamAV patterns. Increasing number of regular expressions.</td>
</tr>
</tbody>
</table>

**Multi-function Security Appliances**

- Unified Threat Management (UTM)

**Solutions**

- Multicore Processors
- Hardware Offload Engines
Offload Requirements

► Capability to offload routine, but expensive jobs
► Offloading of jobs to hardware engines should not require major rework of software
► IO overhead of using offload hardware engines should be as small as possible
► Packet ordering should not be changed by look-aside offload functions in multicore environments
► Offloading Areas
  ▪ Offload processing before incoming packet is given to software (Ingress Offload)
  ▪ Offload processing during packet processing in the software (Look-Aside Offload)
  ▪ Offload processing just before packet is sent out (Egress Offload)
## Offload Functions - Ingress

<table>
<thead>
<tr>
<th>Offload Functions</th>
<th>Description</th>
<th>Required by</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Offload Distribution Across Cores</strong></td>
<td>► 5 Tuple-based (For clear traffic)</td>
<td>► All security functions</td>
</tr>
<tr>
<td></td>
<td>► IPSec SPI-based (For incoming secured traffic)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>► VLAN-based (Virtualized Gateways)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>► Flexibility of custom selection of protocol header fields</td>
<td></td>
</tr>
<tr>
<td><strong>Offload Checksum Verification</strong></td>
<td>► Avoid checksum verification in software (IP, TCP, UDP, ICMP)</td>
<td>► IPS, Proxy based security (AV, AS, HTTP/SIP Firewall)</td>
</tr>
<tr>
<td></td>
<td>► TCP, UDP checksum offload particularly saves large number of CPU cycles</td>
<td></td>
</tr>
<tr>
<td><strong>Offload Protocol Field Extraction</strong></td>
<td>► Layer2 Protocol Headers (all kinds of Ethernet standards)</td>
<td>► Firewall</td>
</tr>
<tr>
<td></td>
<td>► IPv4/IPv6, TCP, UDP, ICMP, IPSec, etc.</td>
<td>► IPS</td>
</tr>
<tr>
<td></td>
<td>► Flexibility to extract custom protocol fields or protocols defined in future</td>
<td>► TP and TS</td>
</tr>
</tbody>
</table>
### Offload Functions - Ingress

<table>
<thead>
<tr>
<th>Offload Functions</th>
<th>Description</th>
<th>Required by</th>
</tr>
</thead>
</table>
| **Offload Traffic Prioritization**        | - Prioritize Management Traffic over Data Traffic to make UI accessible always even in DDoS situations  
                                         - Traffic Prioritization within data traffic. Example:  
                                           • Conversational (Voice Data)  
                                           • Streaming (Video)  
                                           • Interactive (SSH, Chat, HTTP, etc.)  
                                           • Background (SMTP, File Sync, etc.)  | Generic requirement                |
| **Offload Traffic Policing**              | - To control traffic across multiple networks/protocols, etc.  
                                         - Classify packets to multiple policers. Example: VLAN-based classification  
                                         - Mark the traffic using dual-token, three-color method. (Green, Yellow, Red)  
                                         - Queue the packets until read by software  
                                         - Use RED, WRED or taildrop to manage Queues  
                                         - Flow control by sending Pause frames  
                                         - Control (rate limit) traffic that consume lot of CPU cycles | Traffic Policing, DDOS Protection   |
| **Offload TCP Receive Consolidation**     | - Lower number of packets better the performance  
                                         - Consolidate multiple consecutive packets of a session | All security functions             |
## Look Aside Offload Functions (During Software Packet Process)

<table>
<thead>
<tr>
<th>Offload Functions</th>
<th>Description</th>
<th>Security App</th>
</tr>
</thead>
</table>
| Offload Crypto and Protocol Encap/Decap       | ▶ Symmetric Cipher & Hash Algorithms  
▶ Public Key Crypto Algorithms (RSA, DSA and DH)  
▶ Random Number Generator  
▶ IPSec Protocol Intelligence to offload protocol encapsulation & decapsulation  
▶ SSL Record Layer  
▶ DTLS Data Layer  
▶ SRTP Data                                                                 | ▶ IPSec  
▶ SSLVPN  
▶ VOIP |
| Offload Pattern Matching                      | ▶ Regular Expression Search  
▶ Stateful Rule Match (return result only when combination of patterns match)                                                                 | ▶ IDP, AV, App Detection  
▶ Application firewalls |
| Offload XML Validation, HTML, Java script Tokenization | ▶ XML traffic increased dramatically in recent years.  
▶ XML Validation with XSD.  
▶ XML/HTML Parsing  
▶ XPath Evaluation                                                                 | ▶ IDP  
▶ Web Application firewall |
## Offload Functions During Software Packet Processing

<table>
<thead>
<tr>
<th>Offload Functions</th>
<th>Description</th>
<th>Security App</th>
</tr>
</thead>
</table>
| Offload Buffer Pool Management    | ► Avoid overheads of  
• Maintaining core-based memory pools  
• Memory-pool management (Replenishing, Alloc, Free)  
► Almost all modules require memory blocks | All          |
| Offload Queue Management          | ► Avoid Overheads associated with Pipeline Model  
• Queue Management overheads  
  (Enqueue/Dequeue, dropping events/packets)  
► Scheduling high-priority event over low-priority events (Prioritization) | All          |
| Offload Timer Management          | ► Each firewall session, Ipsec SA requires one timer at minimum. Hence millions of timers.  
► Creation, deletion and expiry of millions of timers | All          |
## Offload Functions – Egress

<table>
<thead>
<tr>
<th>Offload Functions</th>
<th>Description</th>
<th>Security App</th>
</tr>
</thead>
</table>
| **Offload Egress Traffic Shaping** | ▶ Shape the traffic based on uplink Routers’ Policing SLA to reduce chance of dropping important packet by uplink router.  
▶ Shape based on SLA bandwidth (Effective bandwidth)  
▶ Queue traffic based in priority to smoothen burst traffic  
▶ Schedule the packet based on priority (DSCP value) | Traffic Shaping |
| **Offload TCP Stateless Segmentation** | ▶ Reduce number of packets going through security functions.  
▶ TCP Segmentation based on MSS | All            |
# QorIQ™ Platform Levels

<table>
<thead>
<tr>
<th>QorIQ P5</th>
<th>Highest-performing embedded processors</th>
<th>Service Provider Routers</th>
<th>Network Admission Control</th>
<th>Storage Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>QorIQ P4</strong></td>
<td>Tap the full potential of multicore with this “many-core” platform</td>
<td>Metro Edge Router</td>
<td>IMS Controller</td>
<td>Radio Network Control</td>
</tr>
<tr>
<td><strong>PRODUCTS:</strong></td>
<td>P4080</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QorIQ P3</td>
<td>Your first step into true multicore performance</td>
<td>Converged Media Gateway</td>
<td>SSL, IPSec, Firewall</td>
<td>Access Gateway</td>
</tr>
<tr>
<td>QorIQ P2</td>
<td>Unprecedented performance per watt in this highly integrated platform</td>
<td>Unified Threat Management</td>
<td>VoIP Carrier-Class Media Gateway</td>
<td>Wireless Media Gateway</td>
</tr>
<tr>
<td><strong>PRODUCTS:</strong></td>
<td>P2020</td>
<td>P2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QorIQ P1</td>
<td>A highly integrated, cost-effective, low-power platform</td>
<td>Integrated Services Router</td>
<td>Network Attached Storage</td>
<td>Home Media Hub</td>
</tr>
<tr>
<td><strong>PRODUCTS:</strong></td>
<td>P1020</td>
<td>P1011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unprecedented performance per watt in this highly integrated platform

Your first step into true multicore performance

Tap the full potential of multicore with this “many-core” platform

Highest-performing embedded processors
QorIQ™ P4 Series P4080 Block Diagram

Power Architecture™
- e500-mc Core
  - 128 KB Backside L2 Cache
  - 32 KB D-Cache
  - 32 KB I-Cache
  - 1024 KB Frontside L3 Cache

CoreNet™
- Coherency Fabric

Frame Manager
- Parse, Classify, Distribute
  - Buffer
  - Mgr.

RapidIO™ Message Unit (RMU)
- PCIe
  - 2x DMA
- PCIe
  - sRIO
- Aurora

18-Lane 5 GHz SerDes

Additional Components:
- eOpenPIC
- PreBoot Loader
- Security Monitor
- Internal BootROM
- Power Mgmt
- SD/MMC
- SPI
- 2x DUART
- 4x I2C
- 2x USB 2.0/ULPI
- Clocks/Reset
- GPIO
- CCSR

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2009. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.
Frame Manager Accelerator functionality

► Offload Distribution Across Cores
► Offload Checksum Verification
► Offload Protocol Field Extraction
► Offload Traffic Prioritization
► Offload Traffic Policing
► Offload Egress Traffic Shaping
Security & Pattern Matching Engines
Accelerator functionality

- Offload Crypto & Protocol Encap/Decap
- Offload Pattern Matching
Buffer Manager & Queue Manager Accelerator functionality

- Offload Queue Management (Pipelining Queues)
- Offload Buffer Pool Management
- Offloading Timer Management (Partial Support)
Summary

Security Appliances require high computing power
• To satisfy growing demands of bandwidth
• To do deep-packet and data inspection to detect & prevent sophisticated attacks.

Solution: Multicore processors with integrated acceleration engines

QorIQ™ P4080 multicore processor
• Designed for networking and security related appliances and markets
• Combines 8 cores running each at 1.5Ghz with acceleration Engines SEC, PME, FMAN, QMAN and BMAN
  ▪ Provides acceleration engine at Ingress, Look Aside and at Egress level.
• 2 Mbytes of L3 Cache in addition to L1 and L2 Caches with facility to position the code
QorIQ™ P4 Platform P4080

- Industry-leading performance in under 30-watts (max)
- Streamlined programming
  Through close partner collaboration, the P4080 is well-tooled –even before silicon availability. Leveraging the hybrid simulation environment, Simics® Virtual Platform for the QorIQ P4080 from Virtutech, developers can migrate code, work through code partitioning and even have fully debugged software early in the development cycle.
- Eight Power Architecture® e500mc cores
  Operating at frequencies up to 1.5 GHz with private L2 cache and embedded hypervisor technology, these are the most advanced cores available in a multicore architecture today. Who needs 16 when you can do it on eight?
- Advanced virtualization technology
  Each core is able to operate fully independent of the other cores –accesses to memories, datapath accelerators and network interfaces are completely contained; safe and autonomous operation of multiple individual operating systems is ensured.
- On-demand datapath acceleration
  Datapath acceleration IP works in concert with the cores to manage packet routing, security, quality-of-service (QoS) and deep packet inspection –freeing the cores to focus on value-added services and application processing.
- CoreNet™ coherency fabric
  Eliminates bus contention, bottlenecks and latency issues associated with scaling shared bus/shared memory architectures that are common in other multicore approaches.
Datapath Acceleration Architecture

Handles over-the-top traffic

- Bandwidth-intensive multimedia and mobile traffic affected by social patterns or new service creation (Facebook, Telepresence, Skype)
- Drives new demands for network architecture responsiveness in service creation and transport
- Freescale’s next-generation Datapath Acceleration Architecture (DPAA) provides the ability to meet such demands
- 18 Mpps parse and classify, load-steering, network accelerators and multi-level prioritized queuing

DPAA simultaneously enables a lower complexity software environment as well as very high networking performance

"Intelligence is the ability to avoid doing work, yet getting the work done."
- Linus Torvalds

QorIQ™ P4
Platform DPAA

Cores

Accelerators

Congestion Mgmt

Parse

Classify

Steer

Manage Work Q

Enqueue

BMan

FMan

QMan

Policing

Stash Context

Network Interfaces

"Intelligence is the ability to avoid doing work, yet getting the work done."
- Linus Torvalds