The Case for Ultra-High Performance Silicon Security Systems

Next-Generation Internet Security Solutions that Eliminate Processing Bottlenecks

Speed, Security and Sales 2
- Why Speed Is Important 2
- Incomplete Transactions 3

The Need for Robust, Universal Transparent Internet Security 3
- How Ecommerce Security Is Delivered. 3
- Gaining Trust 4

Security Then and Now 4

The Inability of Current "Evolutionary" Security Solutions to Enable 5
- Specialized hardware devices 5
- Services Strategies 6

General Design Requirements—A New Class of Semiconductors 7

About Layer N Systems 8

The Case for Ultra-High Performance Silicon Security Systems

Next-Generation Internet Security Solutions that Eliminate Processing Bottlenecks

Increasing ecommerce throughput without compromising security is a goal that many hardware, software and even services vendors seek. Currently, Internet security is rationed to only the most sensitive data transfers because it is far too resource-intensive. This paper addresses the shortcomings of current hardware and software solutions and proposes a silicon-based solution for optimum robust Internet security.

Speed, Security and Sales

Ecommerce has made tremendous inroads, yet many prospective users are still on the sidelines because of security concerns. The fundamental problem is that while the technology to enable secure transactions is readily available, it is cost-prohibitive to implement, and is therefore rationed to only the most crucial parts of ecommerce.

Americans are spending more money than ever online. In fact, the Web is the fastest-growing retail channel, according to Boston Consulting Group.[1] Even though online buying was less than 2 percent of total retail sales in North America in 2000, BCG expects retail revenues from online sales will be $65 billion in 2001.[2] Market analyst firm IDC, a subsidiary of International Data Group, the parent company of IDG News Service, estimates the worldwide ecommerce transactions — driven largely by B2B — totaled $354 billion in 2000.

As more people come online worldwide and the ease of executing online transactions increases, companies like IDC, Forrester Research and BCG estimate the number of transactions to range between $5-7 trillion by 2005.[3]

Definitions of ecommerce range from only those financial transactions executed via the Internet, to include those that are executed from email, those from extranets, and to those that are simply a verification of identity. Regardless, all analysts predict that the amount and value of transactions committed can go nowhere but up.

Fueling this projected growth are two trends: More people outside the U.S. coming online, and diminished fears associated with conducting ecommerce transactions. Both trends reflect expected increases in Internet reliability and security.

Why Speed Is Important

Because ecommerce revenues are boosted by increases in ease and speed, especially for financial transactions, system OEMs have been focused on these market opportunities.

Nothing is more frustrating to users—or more suspicious—than being caught in the middle of a hung financial transaction. Even transactions that eventually execute, but do so slowly, lead to user distrust—usually expressed in terms of an abandoned shopping cart.

Of those who come to Web sites and initiate a shopping cart event, over 90 percent leave before buying.[4] With so much traffic and commerce predicted for the Internet, it would be logical to predict that companies with an eye-appealing, functional Web site would be highly profitable.

---

[1] The Industry Standard Jun 11 2001.

[2] Industry Standard, May 10 2001..

[3] IDG Industry Standard, May 23 2001.

Yet, the immediacy and availability of the Internet also means that users easily wander off a site without completing a transaction.

Incomplete Transactions

One major reason for incomplete transactions is the time factor. When the transaction process is slowed, second thoughts—as well as distrust—are entertained by users. Waiting for a transaction to finish is not an inherent talent of the average online buyer. In fact, 4 seconds is frequently used as a rule of thumb when assessing site response rate. A transaction must be completed in under 4 to 8 seconds or the average user will terminate the transaction. The longest an average customer will wait is 8 seconds. That's not much. But in Internet terms, 4 to 8 seconds can be an eternity. And it can mean the difference between increasing revenues for your company or losing a customer. Then, too, a potential customer who aborts a transaction is probably lost forever.

The second reason a transaction is not completed is the issue of security—specifically, the lack of trust in the security of the event. Any question that a transaction is not secure—and a slow moving transactions does provoke this reaction, then it is only the strongest of heart or the most foolish who continues with completing the transaction.

Moving bits fast and reliably is a commodity. Security is the next value-add. And, because you can't see "into" the security process, providing security is also one of the great unknowns.

The Need for Robust, Universal Transparent Internet Security

Contrast the need to avoid interrupts in completing a transaction with another statistic: About 75 percent of U.S. organizations have experienced a significant information-security breach in the past year, according to Meta Group research.[5]

As information systems become more complex and drive the business, the cost of security breaches has risen dramatically. Our world is now governed in large part by databases that sit on Web servers all around the world. Credit card information, names, addresses, and like information are often the target of hackers and the reason for firewalls and other preventative hardware and software tools.

Virtual private networks (VPNs) and intranets have a very defined perimeter that can be policed heavily without too much anxiety or overhead.

But since the Internet is by nature a public network, security is extremely hard to enforce. The large number of carriers and service providers used in transporting a particular traffic session from point to point exponentially multiply the dimensions of the problem of providing security over the Internet.

While the chances of a system being totally secure are nil, there are ways of minimizing the threat of theft. Such solutions entail a combination of hardware, software, and policy measures.

Traditionally, the most sensitive data is encrypted. The simplest way is by configuring a Web server to use the secure sockets layer, or SSL.

How Ecommerce Security Is Delivered.

The Internet is all about instant gratification, whether searching for information, making a winning bid for a treasured collectible, selling stock based on margin trading, or transmitting personal medical information.

Excluding the VPNs, most such transactions executed via the Internet are secured using SSL client-to-server encryption technology. Developed by Netscape, the scheme uses both public and private keys to

---

[4] The Boston Group reports that 97% of people leave before buying at more than 10,000 e-tailing sites. Sixty-five percent who start to fill up a cart abandon it before going through the checkout process, according to a Shop.org study by BCG. See Gary Andrew Poole's "The Riddle of the Abandoned Shopping Cart," The Industry Standard, by November 10, 2000.

[5] June 2001, Secure Computing

authenticate users. The RSA-key encryption/decryption process is the hallmark of the SSL transaction. [See **Secure Sockets Layer (SSL) explained**, **RSA explained**, and **The SSL exchange**]

The process of exchanging information under secure conditions to confirm identity and complete financial transactions is expensive in terms of CPU cycles and server space. Setting up a session, sending the credit card number and tearing down the session, for example, can take just seconds over high-speed connections. Power spikes actually result from the decryption side of the computation.

But the RSA public key exchange used with SSL can and does take computing power. Large encrypted keys that are 512- to 1024-bits long deter code-breaking algorithms if for no other reason than the sheer size of the computation required to decrypt is daunting and time-consuming. Even when this processing is offloaded to a Web server or an SSL accelerator, the decryption function multiplied thousands and thousands of times over can bring the system to a screeching halt.

According to a recent study by Networkshop, popular server/OS platforms experienced anywhere from 13 to 73 times fewer transactions per second when processing SSL sessions.

Gaining Trust

The other expense of ensuring security is gaining the trust of new users. A high degree of trust that transactions are secure is required if the investments already made in ecommerce are going to see positive returns and match the huge projections of every major analyst firm for the 'Net's ecommerce growth.

The case for a transparent security solution embedded in a chip increases proportionately with the size of the encryption key.

Security Then and Now

Complex problems lead to complex solutions. In the beginning, the Internet was not designed for security uses. Instead, it was initially designed for better communications among otherwise distant researchers.

The network technologies traditionally expand capacity by extending core capabilities. These have tended to provide network engineers with applications and enhancements that improve the reliability and performance of the network. As such, Internet security measures have been "smeared on" as an afterthought, rather than built in.

This reaction has led security to be deployed using several, predictable schemes.

- First, as an enduser application software is used for quick fixes. The responsibility for awareness, installation and use of the software is left to the enduser.

[diagram: General software solution / Enduser "fix"]

- Then, as security has been escalated as a desired feature, more purposeful software has been developed. Yet, because of the CPU "expense," true security solutions dependent on software are rationed, so that systems do not explode from extreme computational needs.

[diagram: Specific software solution / Rainbow nCipher]

- Next, hardware-based solutions find their way into the network to boost capacity. Yet, the first generation of hardware still treats security as an afterthought, adding in cards to servers that "assist" host CPUs with difficult math processing but do not offload the task. Obviously, such solutions provide only incremental gains.

[diagram: Hardware solution/no relief to CPU]

- As standards and normal operational processes stabilize, and as the volume of processing needs continues to push hardware and software solutions, embedding the solution in silicon becomes the next step in the technology chronology. Often, an ASIC is developed.

[diagram: Hardware solution – ASIC solution]

- Finally, more progressive solutions include specialized network processors.

[diagram: Specialized hardware solution]

Throughout this chronology, the underlying technology remains essentially intact. An old framework, such as the 8086 architecture, inherently limits future growth and creative solutions that incorporate newer technologies.

For example, router architecture has not changed dramatically, even though network demand continues to ratchet up notches at a time. Securing ecommerce transactions has followed the same path that was carved by routers in their technological evolution. What has changed is the hardware, software and services strategies that help to offload the burden the shear volume of number crunching imposes on the old architectures.

The Inability of Current "Evolutionary" Security Solutions to Enable

Adding more and more muscle doesn't change the fact that the underlying problem is that there's more and more data that has to be pushed through the same pipes.

To avoid getting locked into a certain scale or set of standards, it's important to build a PKI platform with scalability and flexibility in mind. Modularity is beneficial in this case.

The alternatives are to speed up the algorithm, get a bigger processor, partition the load, or develop a services strategy.

SSL is used by so many systems that it has become the de facto standard. Any challengers to this algorithm will have an uphill battle being adopted.

Specialized hardware devices

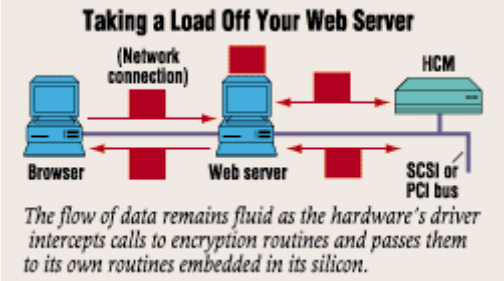
Initial efforts to speed up SSL transactions focused on optimizing the server software but this only led to meager gains, with most servers supporting only 40-50 transactions per second (tps).

The next series of solutions used specialized hardware in the form of add-in coprocessor cards, which were distributed on each web server. In this case, the server downloads much of the math intensive functions to the card when processing an SSL transaction. These cards usually feature a semiconductor coprocessor designed to handle SSL's crytographic operations such as RSA key generation and ARC-4 bulk encryption. Using the system PCI bus to communicate to the server CPU, these coprocessors raise the bar on performance to 100-200 tps. However, in many cases the bulk encryption is still done on the server CPU, since that is faster than facing delays associated with communicating over the PCI bus to the card.

Moving security for Web transactions to a device specialized in crunching algorithms makes a lot of sense. Designing such a device with only one focus in mind often provides the boost intended. Without such

accelerators, an average Web server can crunch between 40 and 170 SSL tps.[6] SSL accelerator vendors offer solutions that give as much as 2400 RSA operations (SSL's decryption process) per second.

Two types of accelerators exist. They are *terminators* (also called *appliances*), which act like a gatekeeper between the client and the Web server, and *off-loaders* (or *peripherals*), such as a PCI card or a SCSI device that takes care of key generation and signings as well as encryption and decryption.



[Robert – might want something like this – grabbed it from Secure Computing]

A terminator can minimize investment of time and money. Management is easier because it is physically external to the Web server and scales easily. But terminators that can decrypt and re-encrypt for completely secure transactions, may not be the ideal solution that also requires speed. An off-loader can be limited by internal space and slots, but is a more secure solution, especially where end-to-end security is important.

The key metrics are scalability, reduced response times and increased transactions per second. Also important are installation and integration with the Web server, as well as the physical security of the devices.

Some of the drawbacks to hardware solutions like these are:

- Costs can become prohibitive. Since every server in the Web farm may need a card, total costs can quickly mount for large sites.
- Numerous certificates are required. Each server needs to maintain its own certificate, which makes management for a large web site very demanding.
- Difficult to scale. Adding SSL capacity means adding more servers and companion accelerator cards.

SSL appliances can offer several advantages over SSL accelerator cards:

- Adding SSL capacity becomes easier to scale and manage (simply add another box).
- Managing certificates is much simpler, since the certificates and private keys reside on a central platform rather than distributed on individual servers.

Flexibility is a hallmark for a good general processor CPU. Such a processor must be able to run a variety of heavy CPU using programs like Excel spreadsheets or games like Tomb Raider. But for security, network engineers know what problem they are solving. Because no universal security engine exists, any hardware solution that overcomes the encryption/decryption barrier to speed must be dedicated to solving the SSL solution.

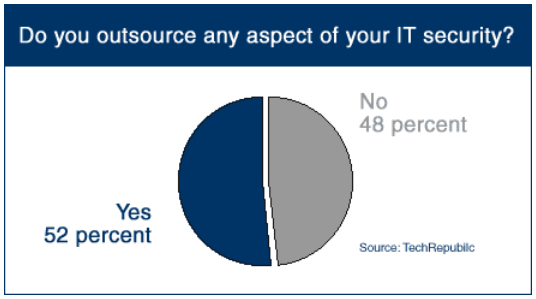
A few specialized chips have been developed that take care of SSL key generation. A chip dedicated to processing for security purposes must also have network functionality so that it has the appropriate ingress and egress for the security algorithms. The traffic cop functionality that directs https: data to the crypto engine should be embedded in the specialty chip. Otherwise, security risks, performance variability and design complexity can overwhelm the overall functionality of the secured transaction.

[6] Network Computing, June 11, 2001.

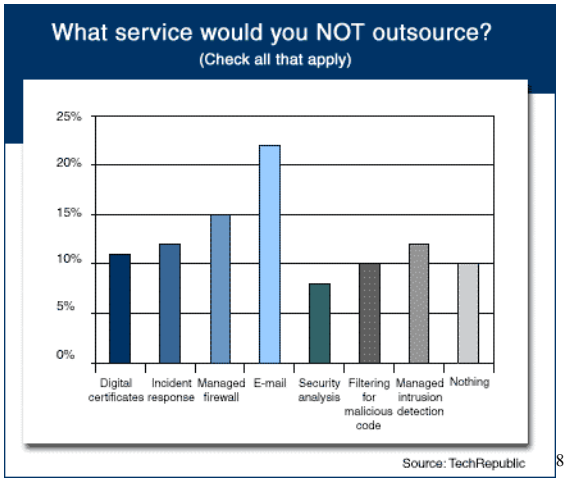
Given the universality of the Internet, it is economically infeasible to ensure security specialists administer the security solutions. Not surprisingly, network security consultants are compensated at a much higher rate than network administrators or even network consultants.[7]

Whether it's a denial of service attack, a malicious break-in, or data theft, the perpetrator is most likely an employee or a former employee. Yet companies continue to focus most of their attention on preventing external attacks.

Overwhelmed with managing network growth (physically and complexity), many companies have turned to outsourcing components of their Internet, including security. Respondents to a 2000 survey conducted by TechRepublic—an arm of Gartner—found that over half of the IT managers outsource a portion of their IT security because they lacked the skilled resources internally.



Amazingly, there are few services that companies will NOT outsource. For example, 88 percent would or are outsourcing digital certificates.



[8]

General Design Requirements—A New Class of Semiconductors

Cardinal principal (in any business): Never trust a piece of equipment to do a job that it was not designed to do.

Example: Routers should not act as a company's main resistance from external attacks.

Example: Network administrators should not be the last line of defense for protecting ecommerce data.

---

[7] TechRepublic, Dec 27, 2000.

[8] TechRepublic, Inc., Sep 22, 2000.

Example: General purpose microprocessors should not be relied on to process high volumes of CPU-intensive cryptographic algorithms.

General purpose microprocessors are not designed to efficiently implement security-focused software. Turning up the amplifier on processing transactions requires stronger more robust architecture and intent. General purpose chips are just not built for high volume traffic, especially in transactions that are secured with cryptographic software.

Hardware, like SSL accelerators that are designed to offload the traffic and process, can double or triple capacity of flow-through. Processing secure transactions averages between between 300 and 600 per second.[9] But even these provide only incremental performance improvements over software solutions. Also, they come with their own overhead, requiring extensive integration efforts and security domain knowledge.

But for evolutionary solutions that offer true leaps in magnitude, security semiconductors must provide OEMs with an easy-to-implement architecture.

For ecommerce transactions, what is needed is a silicon-based solution that is specifically designed to do the complex mathematics, quickly, efficiently and seamlessly. Only the car enthusiast really cares about the details of what is under a hood. And usually that knowledge is used as leverage in a race with money on the table. Ultimately, what goes on behind the scenes must be reliable and thoroughly trustworthy.

But an ideal chip-based solution should be like a motor humming 24/7 under the hood. Maintenance not required. Tinkering allowed.

A processor that provides OEM system designers heavy-duty processing power specifically architected for complex algorithms means designers don't have to make tradeoffs elsewhere when balancing system resource needs and financial considerations.

Further, the processor should accommodate network expansion without loss of performance. The processor should be easily deployed in a network so that non-security administrators can deploy and then align with the existing hardware.

Solutions that ease the "pain" of processing a transaction without losing security or integrity must

- Be fiscally feasible, both low total and per user costs.
- Require minimal management.
- Be modifiable and customizable.
- Be compatible with most browsers.
- Be easily expandable.
- Provide interoperability.
- Provide a non-intrusive, but always-on presence.
- Have embedded network functionality.

Trying to move a boulder with a toothpick is nonsensical. Yet solutions being offered to process big computations for security's sake are not addressing the real issue. Trying to get more through the same bottleneck is not the answer. Instead, the solution must be able to relieve the pressure

Ultimately a networking security system on a silicon chip is the most feasible solution – as long as it offers performance levels 100x higher than today's hardware solutions, and up to 1000x faster than security software running on server CPUs.

About Layer N Systems

[9] nCipher's nShield™ for example is reputed to be 400 tps.

Based in Austin, Texas, Layer N Networks was founded in March 2000 as a fabless semiconductor company to build a new class of network security semiconductors. Layer N's vision is to end the rationing of security and enable universal, transparent, robust Internet security everywhere. Layer N's security chip solutions are designed for processing 100,000 transactions per second. For more information, go to: www.LayerN.com.

Sidebar: Secure Sockets Layer (SSL) explained

For commerce to flourish on the Internet, precient Netscape developed what has become the de facto standard for securing transactions over the Internet.

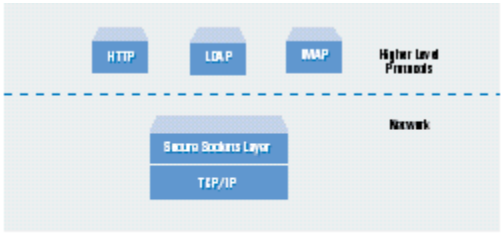


Figure 1 Secure Sockets Layer Protocol

[Create drawing from scratch]

To ensure the high level of security that web transactions demand, SSL uses advanced encryption and authentication algorithms. Part of SSL's attractiveness for Internet applications is its use of public key cryptography such as RSA. RSA allows a secret key (usually 512 to 1,024 bits long) to be securely exchanged over an open, public network. Once both the server and client ends of the SSL connection have established a common secret key using RSA, the next step is to switch over to a bulk encryption method such as ARC-4 or DES which is used to encrypt the actual application data (such as an HTTP web page). Bulk encryption uses far less processing power than key exchange methods.

Cryptographic algorithms are typically very math intensive, meaning heavy processing loads for web servers. The RSA public key exchange algorithm in particular requires exponentiation.[10] of very large numbers in generating encryption keys that are typically 512-bits or longer. Large key sizes deflect code-breaking code-breaking algorithms just because they do require so much processing power. The scale of the numbers involved in key cryptography is staggering: For example, a 265-bit key approximates the total number of atoms in the known universe[11] (equivalent to 10 followed by 76 zeroes)! In contrast, RSA algorithms might require the calculation of raising a 512-bit number to another 512-bit number, a figure beyond comprehension.

Sidebar: RSA explained

(**R**ivest-**S**hamir-**A**dleman) A highly-secure cryptography method by RSA Data Security, Inc., Redwood City, CA, (www.rsa.com). It uses a two-part key. The private key is kept by the owner; the public key is published. Data is encrypted by using the recipient's public key, which can only be decrypted by the recipient's private key. RSA is very computation intensive, thus it is often used to create a digital envelope, which holds an RSA-encrypted DES key and DES-encrypted data. This method encrypts the secret DES key so that it can be transmitted over the network, but encrypts and decrypts the actual message using the much faster DES algorithm. RSA is also used for authentication by creating a digital signature. In this case, the sender's private key is used for encryption, and the sender's public key is used for decryption. The RSA

---

[10] Exponetiation means to raise one number to the power of another. For example, 2^4 equals 2x2x2x2, or 16.

[11] RSA Laboratories Bulletin number 13 – April 2000, "A Cost-Based Security Analysis of Symmetric and Asymmetric Key Lengths".

algorithm is also implemented in hardware. As RSA chips get faster, RSA encoding and decoding add less overhead to the operation. [12]

Sidebar: The SSL exchange

Now a standard component for nearly every Web site, the SSL (Secure Socket Layer) is a low-level encryption standard (devised and used by Netscape). Transactions are encrypted to verify the server's identity to the your system, the client. Also encrypted is data in transit. Finally, depending on your history of transactions with the specific site, you may be asked to provide verification.

This handshake occurs every time a session is initiated. With Microsoft Internet Explorer (IE) 5.0 and later, SSL sessions are set to time out every two minutes, requiring renegotiation of the session -- and a new SSL handshake. SSL handshaking is incredibly expensive in terms of resources for the Web server, and the renegotiation caused by IE 5.x adds unprecedented burdens to SSL-enabled Web servers.

1. You start the process to buy online.

2. Your computer sends its SSL version number, cipher settings, random number and a time stamp to the merchant’s server.

3. The merchant’s server sends you its SSL version number, cipher settings, random numbers, a certificate (which includes a public key), and a request for your computer to send a certificate back.

4. If your system accepts the server's certificate, it creates, encrypts and sends the result to the server. If authentication of your system is requested, your system collects all SSL messages exchanged by your system, encrypts with a private key and sends it along with the your certificate to the server.

[12] www.edtn.com/encyclopedia