Interested in learning more about security?

SANS Institute InfoSec Reading Room

Using Hardware-Enabled Trusted Crypto to Thwart Advanced Threats

Using Hardware-Enabled Trusted Crypto to Thwart Advanced Threats

A SANS Whitepaper

Written by John Pescatore

September 2015

Sponsored by
Thales e-Security

Introduction

Barely a week goes by without a media report of yet another major cybersecurity incident. As of July 21, the Identity Theft Resource Center reports there had been 436 breaches in 2015, exposing more than 135 million records.[1] Several common security failures have enabled these attacks to cause high levels of business damage, including the following:

- **Lack of basic security hygiene.** Many attacks succeed because default passwords have not been changed, patches have not been installed, accounts have excess privileges or system configurations are level in wide-open, insecure states. The Critical Security Controls effort[2] focuses on those areas.
- **Vulnerable users.** Phishing attacks that capture user login information or enable malware installation are at the heart of many of the largest breaches. Changing user behavior through tailored awareness and education campaigns reduces, but does not eliminate, this weakness.[3]
- **Inability to protect data.** The ultimate goal of most cybercriminals or espionage agents is to obtain sensitive user information or critical business data. Encrypting stored data is one of the most effective ways to thwart such attacks, but encryption has been hard to implement and even harder to do right—securely, efficiently and effectively.

Barriers to using strong encryption for business advantage can be overcome through trusted crypto and other features that can be enabled by trusted hardware such as a hardware security module.

These barriers to using strong encryption for business advantage can be overcome through trusted crypto and other features that can be enabled by trusted hardware such as a hardware security module (HSM).

This paper describes how to provide a solid and secure underpinning for the entire life cycle of encryption technologies, from the creation of keys and certificates to the secure deletion of keying material. It also explains the advantages of protecting the keys and algorithms through a trusted execution environment (TEE), a secure space where critical encryption and key exchange algorithms and processes can run with high assurance that they cannot be compromised by malicious software. The same TEE can be used with any critical piece of software, providing additional security and business payback through more automated, integrated means of management.

[1] Identity Theft Resource Center, Data Breach Category Summary; www.idtheftcenter.org/images/breach/ITRCBreachStatsReportSummary2015.pdf

[2] Center for Internet Security, Critical Security Controls; www.cisecurity.org/critical-controls.cfm

[3] Gartner Magic Quadrant (subscription required); www.gartner.com/document/2871817

Why Advanced Targeted Attacks Succeed

The threats confronting organizations have changed over the past few years. Attackers have become better at evading detection, more selective about whom they attack and more adept at using counterfeit encryption keys and certificates:

- **Increased evasion.** Attacks are tailored to escape signature-based malware and intrusion-detection systems and use a wide array of communication paths and techniques to avoid detection by data loss prevention (DLP) technology.
- **Increased targeting.** At the same time, cybercriminals and nation-state espionage agents are focusing on specific data at specific companies in specific industries. This targeting often extends to the types of people who have access to sensitive data—C-level executives and IT administrators.
- **Increased use of counterfeit encryption keys and certificates.** For example, in 2014, Google discovered that the National Informatics Centre (NIC) of the Indian government had been issuing counterfeit keys that were used in counterfeit websites, causing the NIC to cease its certificate issuing operations.[4] Other examples: Components of the Heartbleed bug[5] include key compromise, and the POODLE attack[6] includes cipher block capability.

Advanced Attack Example

Let's consider a case in which the cybercriminal's goal was to obtain financial data that would be used for stock-trading fraud. After researching the target company, the cybercriminal focused on the CFO's administrative assistant for a targeted phishing attack. The attack enabled the cybercriminal to obtain the CFO's login credentials to the financial database and begin stealing sensitive financial data. The data extraction went on undetected for 10 days, at which time the data was seen on the Internet and an internal investigation was begun. See Figure 1.

[4] "Beyond Google, rogue digital certificates also targeted Yahoo domains, possibly others," *PCWorld*, July 10, 2014; www.pcworld.com/article/2452860/digital-certificate-breach-at-indian-authority-also-targeted-yahoo-domains-possibly-others.html

[5] http://heartbleed.com

[6] "This POODLE Bites: Exploiting The SSL 3.0 Fallback," OpenSSL Project, September 2014; www.openssl.org/~bodo/ssl-poodle.pdf

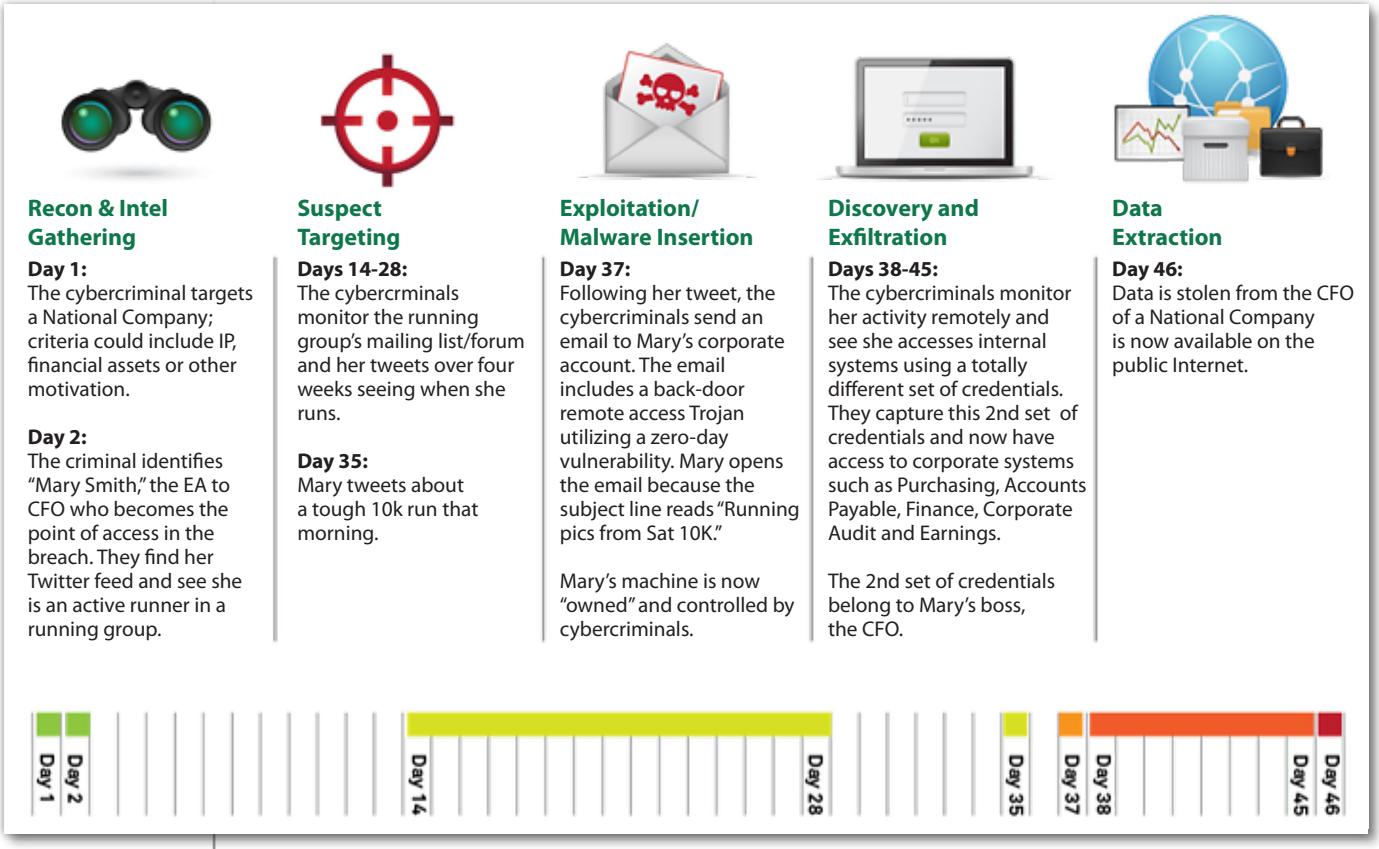

Figure 1. A Computer Gets "Owned," and Data Is Stolen[7]

There are many opportunities to disrupt this common form of attack, but it's important to realize that even if each step in this breach chain could be made 99 percent secure, 5 percent of attacks would still succeed.

Persistent data encryption is the most effective approach for addressing the common gaps in security controls. In the breach outlined in Figure 1, if the data had been encrypted and the encryption keys protected, the data stolen would have been worthless to the cybercriminal, and no breach would have been declared. For this reason, many compliance regimes (PCI, HIPAA, etc.) have long required that critical sensitive information be encrypted.

In order for encryption and other forms of host-based data protection to be effective, however, the servers running those algorithms must have trusted hardware-based capabilities.

[7] Neusentry

Example of Key Impersonation

Many attacks today also exploit vulnerabilities in the encryption key management infrastructure. If that infrastructure is weak, an advanced attack such as key impersonation can undermine the use of public key infrastructure technology. Figure 2 provides an example of how key impersonation is conducted in SSL.

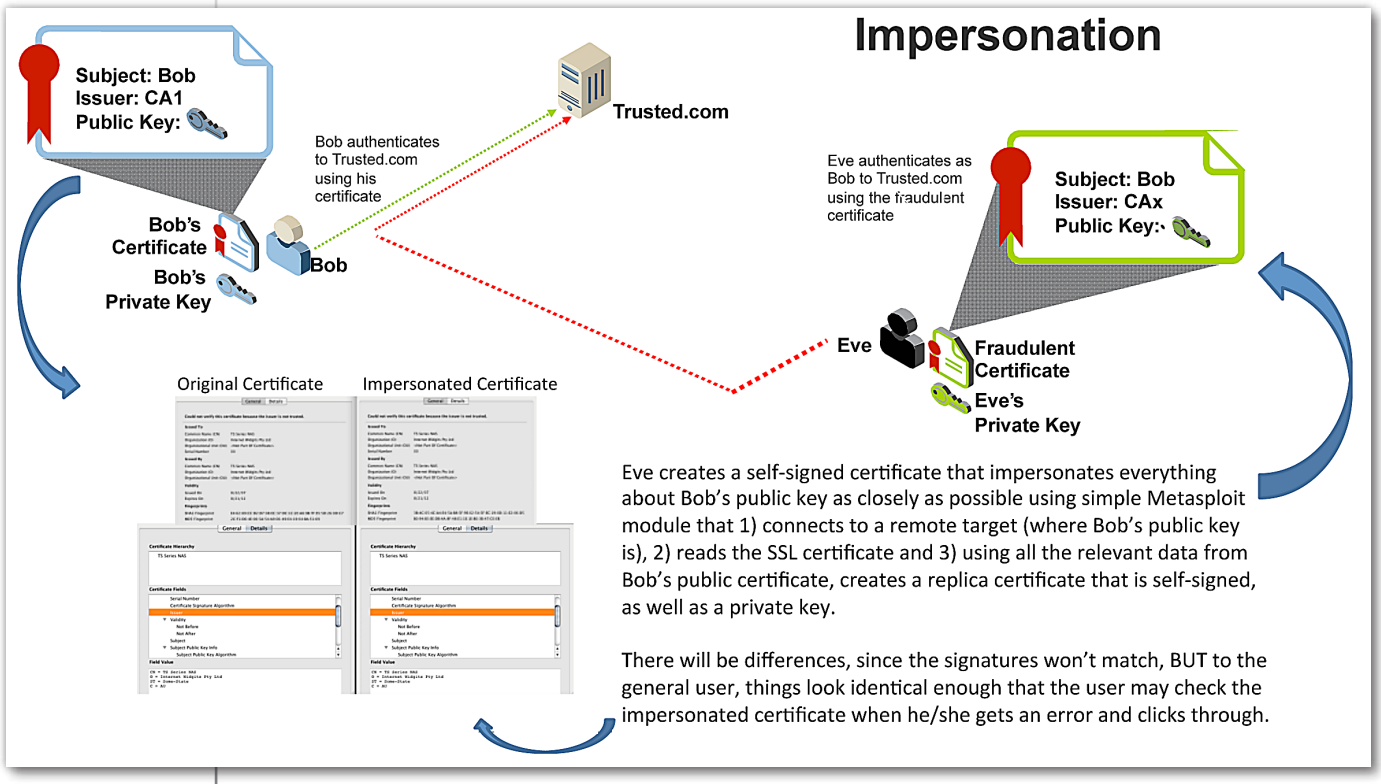

Figure 2. Key Impersonation Attack[8]

The use of trusted-hardware-based crypto processing at the certificate authority can present a tighter line of defense, while also providing secure management environments for the crypto components.

[8] Based on an illustration on the Catch22 (in)security blog, Chris John Riley, September 2011; http://blog.c22.cc/2011/09/04/ssl-certificate-impersonation-for-shits-and-giggles

Gaps in Existing Security Controls

Security is always like a game of chess where every move is met with a countermove. The security controls commonly used by enterprises have evolved over the years in response to changes in the threats (see Figure 3):

- **PC antiviral.** Viruses predated business use of the Internet, and signature-based antiviral controls were able to address the simple malware of the early 1980s.
- **Network-based controls.** The Morris worm of 1988 drove the development of the firewall, which was sufficient for port/protocol-based attacks up through the mid-1990s.
- **Content-based security.** Compromised documents (macro viruses, malicious PDFs, etc.) drove the deployment of email anti-malware inspection, and drive-by/watering-hole attacks drove the use of web security gateways. Application-level attacks drove the migration to next-generation firewalls. All of these security controls inspect content to identify malicious activity.

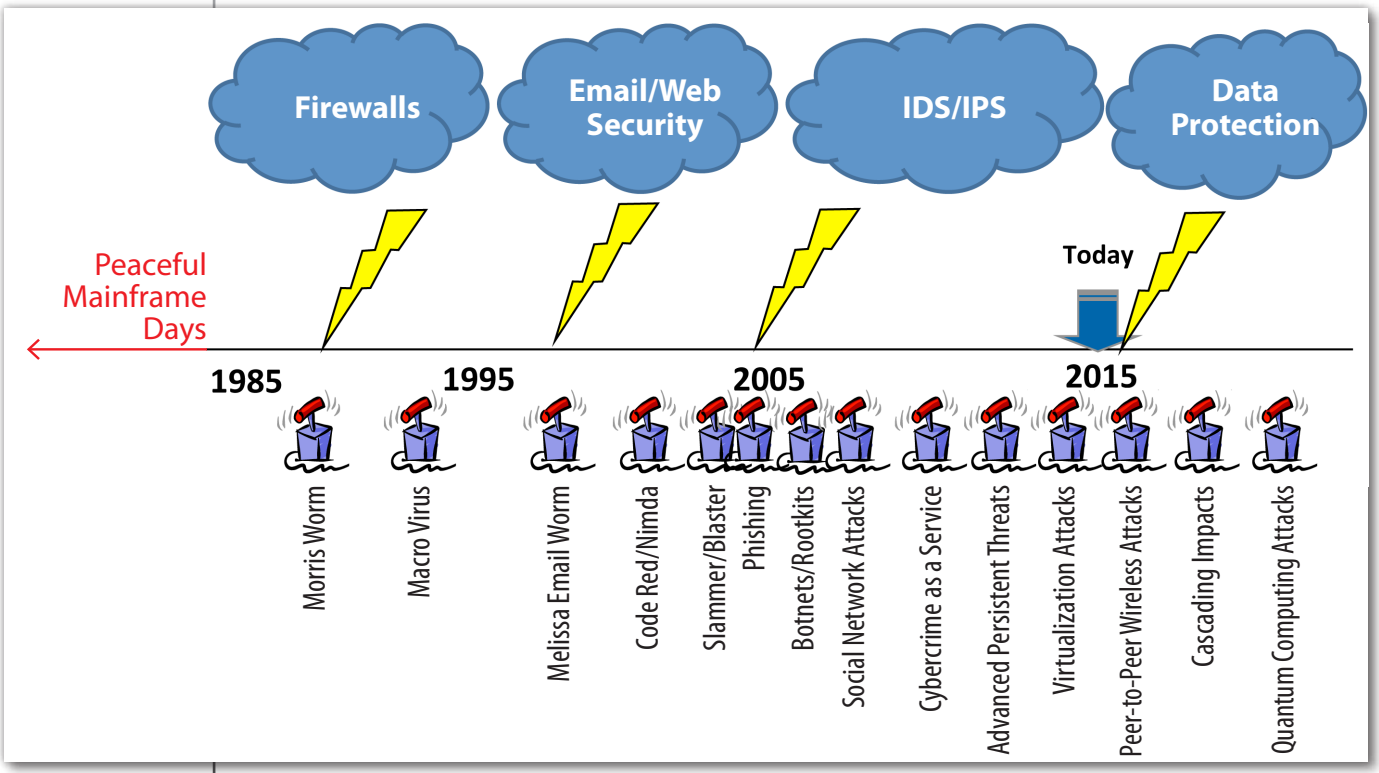

Figure 3. Threat and Response Timeline

By the mid-2000s, cybercriminals and espionage agents began using advanced targeted attacks (often called advanced persistent threats) that employed custom executables and evasion techniques to compromise PCs and servers and steal valuable and sensitive data.

The attackers' focus on data theft (vs. denial of service or vandalism) moved the focus to data protection. Common forms of data protection include the following:

- **Data loss prevention.** Network-based DLP can detect unusual flows of structured information, but attacker evasion techniques often blind network DLP. Host-based DLP offers enhanced visibility into attacks, but because advanced threats compromise the endpoint operating system, host-based controls are easily bypassed or disabled.
- **Transport encryption.** Secure Sockets Layer (SSL) and Transport Layer Security (TLS) are commonly used to protect sensitive data in motion over internal and external networks. However, SSL/TLS can provide a false sense of security, where key and certification generation and validation occur on vulnerable servers. By compromising weak software-only-based SSL server environments, attackers can compromise high volumes of user sessions with a single attack.
- **Persistent data encryption.** Encrypting stored data can provide a high level of protection against data breaches. However, if a server operating system is compromised, or if encryption and key management are not performed securely, data encryption can again just offer a false sense of security—and a very expensive one, at that.

Attackers learned that compromising the OS means all security services running on the OS can also be compromised or disabled.

Software Protecting Software

The common shortcoming of those three forms of data protection is the "software can't protect software conundrum."[9] General-purpose operating systems (such as Windows and Linux) were designed to allow executable application code to be loaded and then use operating system resources. It was this ability that precipitated the move from the mainframe to the client/server model and then to PC-based computing—users could install and run applications without requiring support from IT.

Unfortunately, attackers have taken advantage of this inherent shortcoming of operating systems to create malicious software-based exploits that can compromise the OS to capture passwords, steal sensitive data or cause denial-of-service attacks. Attackers also learned that compromising the OS means all security services running on the OS can also be compromised or disabled.

[9] "Software Security Is Soft Security: Hardware Is Required," Gartner, June 2000 (subscription required); www.gartner.com/document/359830

To effectively combat continually advancing threats while also meeting business demands, computing environments must evolve to add trusted hardware to protect (and accelerate) critical security functions.

With advanced threats continuing to succeed against server security at the operating system and application layers, CPU, PC and operating system vendors understood the need to add trusted hardware to the standard Windows/Intel platform[10] as far back as 1999, when the Trusted Computing Platform Alliance was formed (succeeded in 2003 by the Trusted Computing Group). This and other industry efforts have resulted in some important hardware security capabilities being built into the CPUs and motherboards used in servers, PCs and mobile devices.

Heterogeneous computing environments and performance issues, along with advances in attack techniques, have also driven the development and deployment of hardware security modules—stand-alone, trusted hardware appliances for running mission-critical security applications.

CPU Capabilities

There are two key areas where hardware-based security improvements have been made to the CPUs used in modern computing environments:

- **Buffer overflow resistance.** Over the years, operating systems such as Windows have added a number of built-in capabilities to increase the operating system's resistance to compromise (see sidebar on next page). These mechanisms have been effective against simple malware, but malware developers have learned how to subvert all of them to varying degrees. The Heartbleed vulnerability, which exploited a weakness in the SSL heartbeat function to download chunks of memory and expose unprotected private keys, is a good example.
- **Trusted storage.** Trusted storage capabilities provide a strongly isolated environment in which security-critical values (such as encryption keys or hardware identifiers) can be stored and accessed only by highly trusted software. The Trusted Platform Module built into all Intel CPUs is the most common example.
- **Trusted execution environments (TEE).** A TEE provides highly isolated and trustable program storage, volatile memory and execution space. TEEs can be used as trusted "virtual CPUs" to run security-critical applications and algorithms. ARM Trustzone[11] and Intel TXT are examples of TEE technology.

[10] SANS Institute Reading Room, "Implementing Hardware Roots of Trust: The Trusted Platform Module Comes of Age," June 2013; www.sans.org/reading-room/whitepapers/analyst/implementing-hardware-roots-trust-trusted-platform-module-age-35070

[11] ARM; www.arm.com/products/processors/technologies/trustzone/tee-smc.php

Combating Buffer Overflows

One of the most common forms of attack against operating systems is the buffer overflow exploit. In this form of attack, a huge block of data is sent as input to an operating system function that is expecting a small value. When the large block of data is stored in the memory area intended for a small amount of data, the variable buffer can "overflow" into adjacent executable memory space. This may simply cause the system to crash. But if the large block of data contains specially crafted executable code, the attacker can gain control of the target system.

Microsoft added several forms of protection to Windows after Windows XP SP2 to make buffer overflows more difficult, including the following:

- **/GS switch.** Instructs the C/C++ compiler to generate and check a random number that is placed in a function's stack. If that value gets corrupted, the application is terminated to prevent misuse.
- **Data Execution Prevention/No eXecute (DEP/NX).** DEP and NX use features added to Intel and AMD CPUs to prevent executables from being launched from memory areas designated as data only.
- **Address Space Layout Randomization (ASLR).** When Windows boots up, ASLR moves .dll and .exe files to random memory locations, making it harder for malware to target buffer space.

Linux has also added DEP/NX and ASLR support.

These mechanisms have made buffer overflow attacks much harder, but attackers have found ways around them and uncovered other exploit paths that don't require buffer overflows to succeed.

These hardware-augmented approaches can greatly increase the attack resistance of a computing device, but several barriers to widespread use have slowed adoption:

- Operating system vendors have to modify their software to take advantage of the built-in capabilities and to provide secure and usable management capabilities.
- As the computing world has moved away from a Wintel monoculture, the resulting heterogeneity in CPU and operating system use makes it more difficult to use built-in hardware security.
- Embedded systems (such as point-of-sale [POS] devices and ICS/SCADA appliances) often cannot effectively use the capabilities.
- System administrators can and do make mistakes that disable or bypass built-in hardware security, and they can fall prey to phishing efforts, handing over to attackers the system administrator credentials they need to do the same.

The bottom line is a long-standing truism in security: Infrastructure can never fully secure itself. High-security environments need trusted storage and TEEs that are totally isolated from IT operations and system administration.

Hardware Security Modules

As discussed above, security built into CPUs can improve security, but they are still vulnerable to a wide range of vulnerabilities and attacks. Just as firewalls are used to provide network security that is separate from the operational network, HSMs have evolved to provide trusted storage and execution for high-security applications. HSMs are purpose-built, highly secure appliances or stand-alone processors that implement trusted storage, encryption functions and can also include a TEE.

An HSM offloads security-critical applications or algorithms from the general-purpose CPUs in general-purpose computing devices. HSMs have most commonly been used in encryption applications, for secure key generation, trusted encryption/decryption and secure certificate operations in certificate authorities used to generate and manage public-key certificates. For these reasons, many HSMs also include dedicated cryptographic processors that implement standards-compliant cryptographic algorithms and provide performance acceleration.[12] See Figure 4.

HSMs have most commonly been used in encryption applications, for secure key generation, trusted encryption/ decryption and secure certificate operations in certificate authorities used to generate and manage public-key certificates.

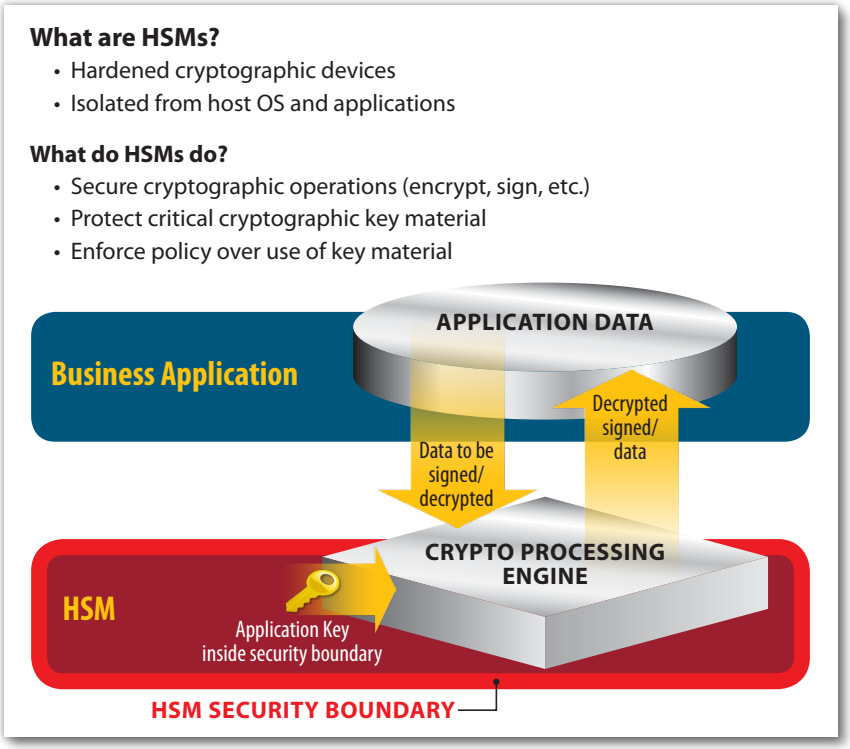

Figure 4. How HSMs Work

By providing this secure environment for all cryptographic functions, HSMs enable all uses of encryption to be highly reliable and trustable. This is particularly important for persistent data encryption, but it is also required for secure transport encryption, such as SSL, and other "secure Internet plumbing," such as DNS and other protocols.

[12] Microsoft Azure Key Vault; http://azure.microsoft.com/en-us/services/key-vault

Use Case Examples

While the media focuses on the breaches, many businesses have been using trusted hardware-based security to avoid or minimize the business damage from advanced targeted attacks. Below are three use cases to demonstrate how secure crypto and execution environments protect businesses and improve risk and compliance.

Point-of-Sale Systems

One of the most common forms of data breaches has involved cybercriminals compromising retail POS systems and obtaining millions of credit card records, enabling billions of dollars of identity theft and new account fraud. The PCI DSS has mandated encryption of card data and requires secure handling of secret and private keys (PCI DSS 3.5.2).

HSMs are used today to support PCI DSS for encryption and key management, protect high-value servers and provide auditable trails used for security, response and compliance (among other high-security business processes).

A typical retail HSM architecture is shown in Figure 5.

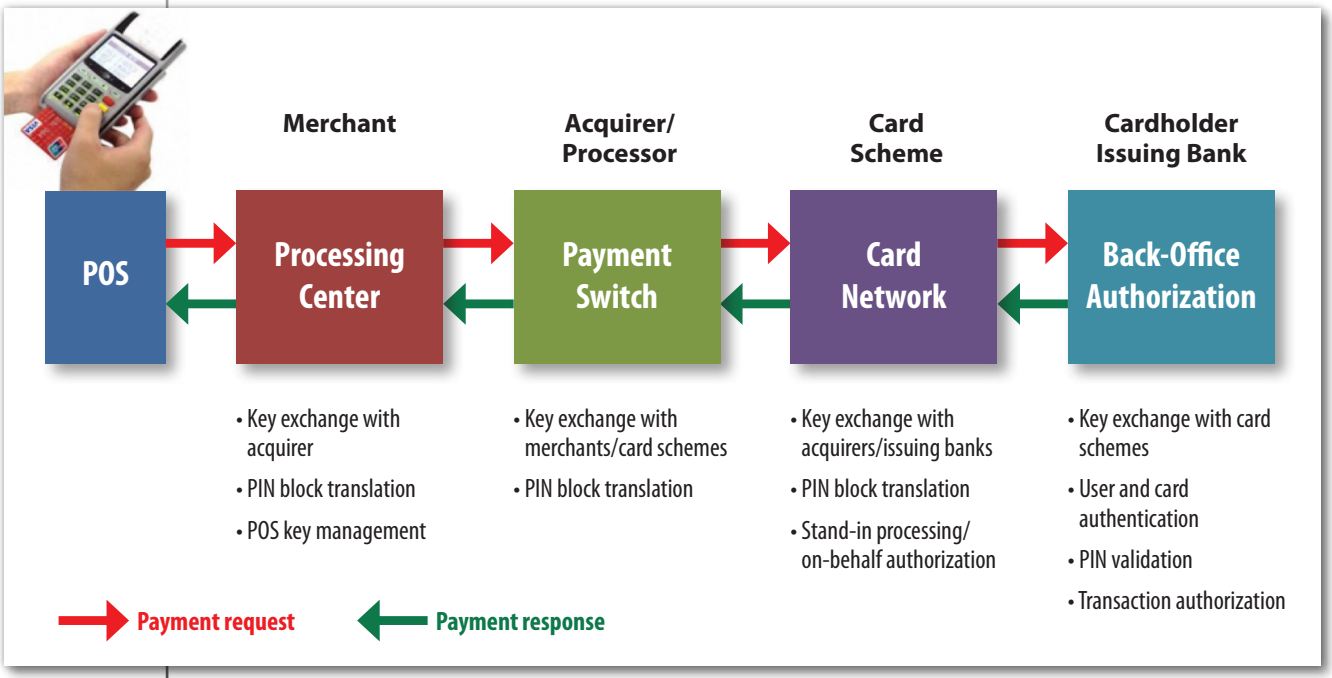

Figure 5. Typical Retail HSM Deployment

Recognizing the importance of HSMs in enabling encryption to be used to secure credit card information, the PCI Security Standards Council has defined a set of security requirements for HSM products.[13] In retail, HSMs are widely used to assure secure and reliable encryption in both brick-and-mortar, card-present transactions and online e-commerce.

High-Value Servers

Public-key algorithms are used for key exchange and transport encryption in a number of critical Internet protocols used in high-value applications. The most common example is SSL, which is the most widely used transport encryption on the web. However, since 2007, attacks have been active that can inspect server-side RAM and extract private keys by using randomization detection techniques. This attack has been demonstrated against a wide variety of protocols:

- PKI certificate authorities
- SSL servers
- SSH servers
- Secure DNS servers
- VPN servers

The use of secure execution space for encryption is required to thwart these attacks.

Where a TEE is available and has enough capacity, other sensitive applications and algorithms can be run in the highly secure TEE space. Examples include digital currency (such as Bitcoin) processing, digital watermarking and other forms of digital content protection.

[13] PCI Security Standards Council, Payment Card Industry PIN Transaction Security Hardware Security Module, Required Device Information, May 2012; www.pcisecuritystandards.org/documents/PCI_HSM_Security_Requirements_v2.docx

Security Services

Encryption is the primary example of a security service that requires a TEE. When encryption can be run on trusted hardware, more advanced encryption-based services are enabled, such as tokenization, application and database encryption and code signing. A TEE can enable secure delivery of several other functions, as well:

- **Timestamping.** For non-repudiation and other legal aspects of transactions, timestamping services must have provable accuracy, reliability and tamper-proofing. Forensics investigation of breaches also relies on trustable event timestamping.
- **Monitoring.** Continuous monitoring of security and vulnerability status is key to the prevention of more attacks and to the faster detection of attacks that do penetrate defenses.
- **Audit.** Advanced attackers try to hide their tracks by disabling security software agents on endpoints and by attempting to modify or delete audit trails. Trusted execution of critical security-monitoring and logging capabilities is required to combat these threats.

Deployment Considerations

Encryption done well and securely from the rest of the operating system can provide high return on investment for secure commerce, but encryption done badly inevitably leads to high levels of self-inflicted wounds. TEEs in general and specific products such as HSMs should be thoroughly evaluated. The most important evaluation criteria include the following:

- **Security.** As the cornerstone of a security data encryption architecture, TEEs and HSMs must be designed with high security in mind and must be assessed for security levels appropriate to the planned applications. Factors include secure software development, tamper-proofing, standards adherence and attack surface minimalization. The PCI[14] and the Open DNSSec organization[15] have good templates for evaluating the security of HSMs.
- **Reliability.** Encryption is a business-critical service and cannot be a single point of failure. HSMs need high levels of reliability and must support high-availability/failover configurations.

[14] PCI Security Standards Council, Payment Card Industry PIN Transaction Security Hardware Security Module, Evaluation Vendor Questionnaire, May 2012; www.pcisecuritystandards.org/documents/PCI_HSM_Eval_Vendor_Questionnaire_v2.pdf

[15] OpenDNSSEC Project Wiki, HSM Buyers' Guide; https://wiki.opendnssec.org/display/DOCREF/HSM+Buyers'+Guide

- **Performance and scale.** Each use case will place different demands on cryptographic hardware. Frequency of encrypting, decrypting, signing, key creation, key exchange, key rollover, etc. should be specified. The number of active and archived keys that can be supported is also an important parameter.
- **Integration.** For common applications such as SSL, HSMs may be operated transparently. However, for interfacing with custom applications and for "future proofing" concerns, APIs, toolkits and reference implementations should be evaluated.
- **Virtualization and cloud support.** Persistent data encryption has high promise for enabling cloud storage of sensitive information, but by definition cloud services run in virtualized, shared services environments. HSM capabilities and architectures for use in cloud and virtualized environments should be evaluated.

Both threats and business demands will continue to evolve and increase. Hardware-based security is key to ensuring security controls keep up with those changes. Security architectures, processes and product selection need to focus on both effectiveness in dealing with advanced threats and efficiency in operational expense and business disruption.

The following section provides pointers to additional information.

Conclusion: Future Need for Trusted Encryption Services

Ever since IT moved away from the mainframe, business demand has driven critical and sensitive data to be stored, processed in and accessed from a growing number and variety of locations and devices. As discussed earlier, this movement makes it increasingly difficult to protect the devices and increases the need for strong data protection services such as encryption.

Several trends are exacerbating this demand:

- **Bring your own device (BYOD).** BYOD (or more broadly, choose your own IT) has grown out of the consumerization of IT, where users apply new technologies at work on their own initiative and consumers choose nontraditional technologies for their transactions with a business. Smartphones and tablets are highly visible exemplars of this trend. Facebook, Twitter, Shapchat, Instagram and other social forums also represent consumer services that have become key elements of business plans but also threaten to provide new paths for attackers to steal customer information and critical business information.
- **Migration to the cloud.** Almost all enterprises are moving to hybrid data center architectures where mission-critical production systems are run by a mix of traditional data center servers, virtualized private cloud, public infrastructure as a service and commercial software as a service.[16] Trying to run endpoint security agents or other security controls across such a diverse environment is difficult for all but the largest enterprises.
- **The Internet of Things (IoT).** The IoT takes BYOD and cloud to the ultimate extreme: Every device has an IP address and can play a role in delivering services. Not only will trusted encryption be required for sensitive data that may reside on medical devices, mobile payment systems, etc., but trusted certificate services for a wide variety of protocols that will evolve out of the IoT will be required as well.

Security services are by definition "critical infrastructure" because they enable critical business functions and are high-priority targets for attackers. While software-based security services are sufficient for some low-value applications or low-value processes, hardware-enabled security services are required to assure industrial strength security services can be delivered in the face of increasingly sophisticated threats and complex IT environments. Encryption is the most visible security service benefiting from hardware-enabled trust, but all security controls and high-value applications can gain similar advantage.

[16] SANS Institute InfoSec Reading Room, "Conquering Network Security Challenges in Distributed Enterprises," August 2014; www.sans.org/reading-room/whitepapers/analyst/conquering-network-security-challenges-distributed-enterprises-36007

Appendix: Resources

ARM Trust Zone

http://genode.org/documentation/articles/trustzone

Trusted Execution Environment

www.nfcworld.com/technology/tee/

Hardware Security Modules Secure Encryption

www.informationweek.com/interop/the-rise-of-bring-your-own-encryption-/a/d-id/1320796

About the Author

John Pescatore joined SANS as director of emerging technologies in January 2013, bringing with him over 35 years of experience in computer, network and information security. Prior to SANS, he was Gartner's lead security analyst for more than 13 years, working with global 5,000 corporations, government agencies and major technology and service providers. In 2008, John was named one of the top 15 most influential people in security and has frequently testified before Congress on issues relating to cybersecurity.

Sponsor

SANS would like to thank its sponsor:

THALES

Last Updated: September 21st, 2015

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Bangalore 2015	**Bangalore, IN**	**Sep 28, 2015 - Oct 17, 2015**	**Live Event**
SANS Seattle 2015	**Seattle, WAUS**	**Oct 05, 2015 - Oct 10, 2015**	**Live Event**
SANS DFIR Prague 2015	**Prague, CZ**	**Oct 05, 2015 - Oct 17, 2015**	**Live Event**
SOS: SANS October Singapore 2015	**Singapore, SG**	**Oct 12, 2015 - Oct 24, 2015**	**Live Event**
SANS Tysons Corner 2015	**Tysons Corner, VAUS**	**Oct 12, 2015 - Oct 17, 2015**	**Live Event**
SANS Gulf Region 2015	**Dubai, AE**	**Oct 17, 2015 - Oct 29, 2015**	**Live Event**
SANS Tokyo Autumn 2015	**Tokyo, JP**	**Oct 19, 2015 - Oct 31, 2015**	**Live Event**
SANS Cyber Defense San Diego 2015	**San Diego, CAUS**	**Oct 19, 2015 - Oct 24, 2015**	**Live Event**
SANS Sydney 2015	**Sydney, AU**	**Nov 09, 2015 - Nov 21, 2015**	**Live Event**
SANS South Florida 2015	**Fort Lauderdale, FLUS**	**Nov 09, 2015 - Nov 14, 2015**	**Live Event**
SANS London 2015	**London, GB**	**Nov 14, 2015 - Nov 23, 2015**	**Live Event**
Pen Test Hackfest Summit & Training	**Alexandria, VAUS**	**Nov 16, 2015 - Nov 23, 2015**	**Live Event**
SANS Hyderabad 2015	**Hyderabad, IN**	**Nov 24, 2015 - Dec 04, 2015**	**Live Event**
SANS Cape Town 2015	**Cape Town, ZA**	**Nov 30, 2015 - Dec 05, 2015**	**Live Event**
SANS San Francisco 2015	**San Francisco, CAUS**	**Nov 30, 2015 - Dec 05, 2015**	**Live Event**
Security Leadership Summit & Training	**Dallas, TXUS**	**Dec 03, 2015 - Dec 10, 2015**	**Live Event**
SANS Cyber Defense Initiative 2015	**Washington, DCUS**	**Dec 12, 2015 - Dec 19, 2015**	**Live Event**
SANS ICS Amsterdam 2015	**OnlineNL**	**Sep 22, 2015 - Sep 28, 2015**	**Live Event**
SANS OnDemand	**Books & MP3s OnlyUS**	**Anytime**	**Self Paced**