



Hardware Security Modules

Where Companies Put Their Trust

Public Key Infrastructure (PKI) is now the principal way to assure trust in business; Hardware Security Modules (HSMs) are the predominant way to assure trust in PKI.

Introduction

Products like AEP Keyper demonstrate the important role Hardware Security Modules (HSMs) play in assuring trust in the electronically-enabled business world. In fact, it's not an overstatement to say that HSMs are where companies today put their trust. That's because HSMs are the cornerstone technology of Public Key Infrastructure (PKI) - the leading method of providing the four key elements of trust in any transaction: authenticity, confidentiality, integrity and non-repudiation.

HSMs are computing engines that perform virtually every task vital to a functioning PKI. They:

- Sign electronic documents
- Validate digital signatures
- Produce encryption keys
- Encrypt and decrypt digital certificates and messages
- Authenticate certificates and signatures
- Request and receive certificates and key materials on behalf of end-user organizations

And, perhaps most importantly, they protect the confidentiality of these materials, thereby ensuring trust throughout the entire PKI.

Because of their importance, every system and security administrator needs to understand what HSMs do, and why certain attributes of these modules are critically important. For example, the fact that security functions are performed in hardware instead of software means that key materials can be better protected - both from physical attacks and network attacks - than if they were running inside a general-purpose server. Another advantage of hardware is that it can be optimized to perform security functions much faster and more effectively than its software counterpart.

Security architecture is optimized when it leverages these inherent benefits of hardware:

- Key materials are stored on the HSM, not the server
- HSMs are isolated on their own Ethernet LAN (instead of a PCI card) for greater scalability, fault tolerance, load balancing, and hot swap capability
- Secure software interfaces for remote management, software downloads and interoperability with other PKI components are provided

This paper explains how PKI provides trust, how HSM supports PKI, and how an organization can know if it can trust a particular HSM.

How Trust Works

Businesses cannot operate without trust. In order for people to do business they need assurance that:

1. The people with whom they are dealing are actually the people they say they are.
2. There is no breach of confidentiality.
3. The transaction that is intended is the transaction that actually takes place.
4. The transaction can later be proven to have taken place.

These have always been the four cornerstones of trust: authenticity, confidentiality, integrity and non-repudiation. But, while businesses have always relied on trust, the way that trust operates today has fundamentally changed from the way trust was provided for in the past.

In our modern, global society, people often do business with people they have never met and may never encounter again. They may be on different sides of the planet; they may need to transact business within the first few seconds of coming into contact with each other. In fact, the transactions may not even be between people at all, but between people and machines, or simply between machines. Examples are everywhere: automatic tellers, electronic wire transfers, Web commerce, insurance claims

processing, electronic filing of tax forms. Anywhere business is conducted, trust still needs to be established. But now it must be established over barriers that didn't exist in earlier times, or at least not to the degree they exist now: between anonymous parties, over great distances, with very high reliability, with great speed and efficiency and very often as a completely automated process.

Trust in today's world relies on public key infrastructures (PKI). A PKI is three things: a methodology, a technology, and an infrastructure in which to conduct business. The method is relatively straightforward and relies on the presentation of trusted certificates between parties in a transaction. If you have a certificate that I trust, then I can also trust you. If I have a certificate that you trust, you can trust me. Sources of trust are called Certificate Authorities. Customers trust a business because they trust the certificate the business holds, and the reason they trust the certificate is because the certificate can be proven to have originated from a trusted source.

This linkage between the Certificate Authority (CA) and its holders and their customers is called a **chain of trust**, which ultimately depends on two things: the trust of the CA and the integrity of the certificate as it moves from one link in the chain to another. If the CA is compromised, then there is no trust. This might happen, for example, if an impostor were to issue certificates that fools credit card users into thinking they were dealing with a real bank when they were not. Equally, if the certificate is compromised, there can be no trust. Even if the CA is legitimate, the certificates it issues must still be protected so that they cannot be misappropriated.

How Security Modules Make PKI More Secure

Modern businesses depend on trust, trust depends on PKI and PKI depends on the technology of security modules.

The function of a security module is to issue, validate and store certificates in a protected environment. A security module can be one of two types: a Software Security Module (SSM) or a Hardware Security Module (HSM). The major difference is that an SSM is a program that runs on a general purpose computer, while a HSM is a dedicated computer specifically designed to function in a security role.

Advantages of Hardware Security Modules

HSMs are physically isolated. They are not part of another computer's file system, they do not have a file system themselves, and they do not run an operating system. They are therefore virtually impossible to attack over a network. Most HSMs also offer tamper protection so that if someone attempts to open the module, the information inside will be erased. In addition, HSMs offer safeguards against software tampering. Another major advantage of HSMs is that, because their software and hardware is specifically dedicated to providing security functions, it can be specifically optimized for that purpose. HSMs perform security functions faster and with superior results than their software counterparts. For example, one of the processes at the heart of certificate generation and validation is the generation of random numbers.

HSMs have dedicated hardware specifically designed to generate random numbers and they can therefore generate numbers that have greater randomness than would be the case if the hardware were not specifically designed for that purpose.

Function of Security Modules

What do security modules do? They provide three critical services: certificate generation, validation, and protected storage. When a system, such as, a funds transfer system, requests a certificate to show to another bank that it can be trusted to execute a wire transfer, the security module is the device that actually generates the certificate. When the second bank receives the certificate (as part of the transaction), it also uses a security module, this time to check the certificate and see whether the certificate can be trusted - i.e. that it came from a trusted source and that the certificate was not compromised along the way.

To carry out these functions, security modules generate keys. A key is one part of a certificate - other parts include identity information such as the name of the person or entity issuing the certificate and the name of the person or entity holding the certificate. Other information might be the person's social security number or mother's maiden name. Certificates also contain a special field called a digital signature. It is by signing the certificate (using a certificate's key) that an authority or an organization provides proof that it was party to a transaction - the concept of non-repudiation.

Certificates and their keys also provide confidentiality using a technique called 'asymmetric security', which allows two parties to encrypt and decrypt each other's messages if they do not share the same key. (If they did share the same key that would be called 'symmetric security'.)

With asymmetric security, the security module generates *two* keys that have a mathematical relationship such that a message encoded with one of the keys can *only* be decoded by the other, and vice versa. One of these keys is kept private and the other made public. To decrypt a message encrypted by a specific sender (with their private key), simply use the sender's public key. To encrypt a *secret* key (one that only two parties to a message will share), encrypt it with the other person's public key. They (and only they) will be able to decrypt the secret key with their private key. Both parties can then use the secret key to exchange encrypted messages that no one else can read. This will not only ensure confidentiality, but also integrity, since if the message had been altered during transmission the key could not decode it. It also ensures authenticity, since only the holder of the secret key can decode the message encoded with that key.

Digital signatures use a similar technique to provide non-repudiation, as well as another layer of integrity and authentication. In a traditional business setting, non-repudiation is provided by signatures, sometimes witnessed and notarized by third parties. In an electronic environment, digital signatures provide this same element of trust.

To sign messages with digital signatures, security modules employ a second public/private key pair (i.e. different from the first pair that was used to encrypt and decrypt the message). The entire process is as follows:

1. A hash generator algorithm converts the message to a binary field of fixed length (called a digest).
2. The security module encrypts the digest using the second private key and attaches the encrypted digest to the end of the file (the message is now signed).
3. The security module encrypts the entire file using the first private key and sends it.

Authenticating the digital signature proceeds as follows:

1. The receiver's security module decrypts the message using the sender's first public key.
2. A digest is created by hashing the message with the same algorithm as the sender.
3. A second digest is created by decrypting the digital signature (the encrypted digest embedded in file). This is done using the sender's second public key.
4. The digests created from steps two and three are compared. If they are the same, the signature is authenticated.

Obviously, the digital signature also provides a second layer of integrity assurance since any alteration of the message will mean that the digests would not match. It provides additional authentication since only the holder of the key can sign the message.

ACCE Reflects Core Competency

PKI technology providers enable security functions in hardware because hardware can offer very strong security, performance, scalability, manageability, fault tolerance and remote management capabilities. But exploiting those potential advantages requires experience in multiple areas of competency - experience that can be leveraged and augmented again and again, every time the provider develops a new security HSM. Case in point: ACCE - **Advanced Configurable Crypto Environment** - the core of AEP Networks' cryptographic hardware products. ACCE reflects over 100 years of combined engineering effort focused on developing highly optimized hardware architectures for encryption/decryption, authentication, key management and key protection. For example, in AEP Keyper all critical security functions are housed within the tamper detection envelope, which is being accredited to FIPS 140-1, Level 4. Like other ACCE products, it can also be upgraded with new software and algorithms, and can be configured remotely. It supports a range of key management options, with protected internal key store for over 1,000 keys, backed by secure key export and transport options.

AEP Keyper reflects other HSM best practices as well - for example, that connecting modules via the Ethernet rather than PCI achieves maximum scalability and manageability. Adding processing power is never an issue. You simply plug in as many modules as you want to the Ethernet cable and Load Balancer keeps your systems (and your business) running.

Putting the 'I' in PKI

Methodology and technology alone, however, are not sufficient to create a PKI. By definition, a PKI also requires an infrastructure. That infrastructure consists of a source of trust called a Certificate Authority (CA) - the issuer of the "root certificate" which becomes part of all certificates granted to users. Those users request certificates from Registration Authorities (RAs) that act as intermediaries for the CA. Together, the CA, RAs and end-user organizations comprise the PKI. At all three levels in the hierarchy security modules play key roles.

The CA

The CA, or Certificate Authority, is a workstation with one or more security modules attached that issue certificates in response to requests from RAs on behalf of end-users. The CA must demonstrate an absolute level of trust. For example, it must ensure that keys are protected from theft, both physical and virtual. Rigorous procedures must be in place to ensure secure handling of keys and these procedures must be strictly enforced. The fact that standards are written down, and that procedures for handling key materials are audited (usually by an outside accounting firm) are examples of a CA's methodology and organizational infrastructure at work. An example of a technical standard is FIPS (Federal Information Processing Standard), a U.S. government standard regulating a Hardware Security Module's built-in protection against tampering. A FIPS Level-3 HSM, for example, will automatically sense physical tampering and erase sensitive data if the module is opened. A FIPS Level 4 HSM will also erase the data if the box is physically opened. Furthermore, if the circuits are probed electronically (a technique that might be employed to surreptitiously extract data from memory). AEP Keyper works with all the major certificate authority software vendor's stacks to provide the physical security for keys and other data that is required to underpin the security of the CA.

The RA

The RA, or Registration Authority, is an intermediate layer in the infrastructure. As the name suggests, the RA's role is to register users and organizations that request certificates. Most CAs do not deal directly with end-user organizations, since to do so would be extremely time and resource-consuming; at any one time there might be many organizations requesting certificates, and registration itself is typically a manual process. Another reason for offloading registration tasks is that different organizations have different registration procedures (for example, different identification requirements of the person requesting registration) and these can be more easily satisfied if each organization handles its own registration process. Typically this process involves the physical appearance of an end-user requesting registration before the person operating the RA (an RA administrator). As a rule, the RA itself is a workstation with a HSM attached - either a PCI or Ethernet connection.

The RA administrator typically asks for authentication of the requestor's identity such as two forms of photo identification, as well as other information such as the name of the person's department within the organization. The administrator fills this information into an on-screen form, part of an application running in the administrator's workstation. That application, in turn, submits this information to the HSM, which in turn passes the certificate request onto the CA over the network (obviously in encrypted form).

If the request is granted, the RA then receives the certificate back from the CA. This certificate includes two public/private key pairs (one for message encryption and one for digital signature), the identity of the authorizing CA and the identity of the user. The HSM then provides this information to the end-user, for example in the form of a smart card.

The End User

The third layer in the PKI hierarchy is the organization, department, or individual who ultimately uses the certificate to conduct trusted business transactions. Upon receiving their certificates, including the public/private key pairs, end-users store them in their own security modules. Applications access the certificates programmatically - typically through interfaces that submit documents for digital signatures and/or encryption. End-users access the security modules by using so-called *two-level* security, similar to a bank ATM. Two-level security requires the user to both *have* something (i.e. a smart card) and *know* something (i.e. a pin code). This procedure gives the organization a very strong assurance that only the person authorized will have access to sensitive information.

End-users are now able to encrypt/decrypt messages and sign/authenticate digital signatures with other members of the PKI infrastructure.

Security Module Attributes

By understanding the components of trust, the components of PKI, and the principles of how these various components fit together to provide trust, it is easier to fully appreciate the importance of the security module. The security module is the PKI technology cornerstone. It performs all the functions that are required of a PKI and it does so at all three levels of the infrastructure. Even those things that it does not provide directly rely on the security module. Take the onscreen forms the RA administrator needs to acquire the end-user identification information - an application typically provided outside the security module. It is the security module that accepts the information from this application, constructs valid certificate requests, and sends those requests onto the CA (where another security module is waiting to receive them). Obviously, how well the security module performs its functions at each of the PKI levels has a lot to do with how well PKI can provide trust in business transactions.

To summarize, security module functions include:

- Generating digital certificates, including public/private key pairs
- Encrypting and decrypting messages with those keys
- Generating hash values and signing messages with digital signatures
- Validating digital signatures
- Interoperating with third-party applications
- Protecting certificates and keys from both physical and network-based attacks
- Issuing and accepting requests for key materials
- Providing a two-level secure user interface (i.e. smart card reader and key pad)

What should users look for when considering a security module to support a CA, RA, or end-user environment? The ability to carry out its functions implies a set of attributes in the way security modules are designed and built. Some of those attributes include:

- Hardware based with third party certification
- Keys offloaded from host
- Ethernet attached
- Scalability and redundancy
- Flexible form factors
- Secure remote management

Hardware Based

Many security administrators specify hardware modules because they are more secure and provide higher performance than software security modules. As previously stated, hardware modules are optimized to generate the random numbers for encrypting and decrypting keys and messages and producing keys and hash values. This is high-speed, computationally intensive processing for which un-optimized, general-purpose, computer architectures are at a decided disadvantage (i.e. the type of architecture that would host a software-based security module).

Software modules are also less secure than a hardware module running as an attached processor to the general-purpose server. That's because general-purpose servers have operating systems and file systems with public interfaces that are potentially open to compromise. Hardware modules have no operating systems or file systems so these cannot be compromised.

Another sign of the higher security that hardware modules provide is the fact they are subject to a set of security standards that do not apply to software modules. FIPS 140-1 and its European counterpart, ITSEC E3 are examples. As previously noted FIPS 140-1 level 4 is the highest standard and certifies the module will protect against both physical and electronic tampering.

Key Stored on Module

Surprisingly, not all hardware security modules fully exploit the inherent security advantages of using a separate hardware module for key generation *and* for key storage purposes. Even though they generate keys in hardware that is separate from the

host, they don't store them there, but on the host itself, encrypted under another key. Storing the keys on the host (even in encrypted form) means that a hacker can potentially remove them to another computer where they can be analyzed and perhaps compromised.

Connectivity

Connectivity is a key differentiator among HSMs. It determines how the module attaches to the rest of the PKI. In most cases, that connection is through a PCI interface card on the back of a server. This server is typically running applications (like those that register users) that access the module for PKI services (like encoding a message). A few modules, however, such as AEP Keyper, can also attach via an Ethernet connection. What's the advantage? An Ethernet attachment provides a number of benefits that greatly enhance the module's ability to support security applications at all three levels of the PKI:

Security

One of the advantages of using a hardware security module is that keys can be stored outside the host in a more highly protected environment. It therefore matters greatly how the security module and host are connected physically. Connecting the two over a private Ethernet means that the security module can be effectively shut off from the outside world. Because packet forwarding is switched off, the only access is through the host.

Scalability

There are a limited number of PKI slots on the back of a computer. This severely limits the ability to grow processing power (for example, when a CA needs to handle more certificate requests). An Ethernet connection has no such restriction. As many HSMs can be added as are needed.

Fault Tolerance

Since HSMs are connected over a network, if one goes offline, the remaining HSMs can take over. Aggregate processing throughput for the entire network may fall, but at least the security functions themselves stay online.

Hot Swapping

Another advantage of network connections is that taking a module off the network does not crash the network. Removing a PCI card, on the other hand, requires a reboot. This limits the flexibility of administrators to service or upgrade equipment.

Load Sharing

This is a similar benefit to fault tolerance, except that a module does not have to go completely offline before other modules step in and share the processing load. In fact, the exact distribution of processing load over a number of modules ought to be an option the security administrator can configure - whether, for example, 10 hosts are allocated to one device or to five or to 10. What's important is that the security administrator has the flexibility to balance the load in a way that achieves the administrator's goals, such as to maximize the aggregate throughput or give the most important applications priority access to resources.

A Choice of Interfaces

Flexibility is the key to accommodating today's security infrastructures - organizations are dynamic and change is constant. The same module may operate in a number of different configurations, and several different environments, and roles.

One of the ways Hardware Security Modules provide flexibility is through the types of interfaces they offer to users and applications. We just saw how important it is that hardware modules provide highly restrictive interfaces in the sense that they don't have operating systems or file systems. On the other hand, it is also important that users and application designers have choices about how they want to access the module. AEP Keyper, for example, provides an API (application programming interface) for direct program-to-program communication and optimum performance. Organizations can host this API on their servers and run their own security applications inside the hardware security module. In these types of scenarios the module effectively becomes a cryptographic accelerator for high-speed key generation, PIN validation, online transaction processing or other functions.

This API is in addition to a PKCS#11 v2.01 interface - the industry standard that allows applications and modules in different PKIs to interoperate (perform each other's functions) as if each of them were part of the same PKI. Interoperability is important because it means organizations can exchange secure communications even though they acquire security technology from

different sources. Having both an API and a PKCS#11 interface means that a security administrator can optimize for performance without sacrificing interoperability.

A third type of interface is the one that handles software downloads. In order to use the API, for example, organizations will need to download applications to the module. The module's manufacturer will also want to download software from time to time, such as to upgrade software functionality and algorithms. This will keep the module from becoming obsolete without having to replace it. All such downloads need to be handled in a secure manner - meaning that the downloaded information itself cannot be corrupted and that the interface for these downloads cannot be exploited for malicious purposes. Such a purpose might be to download code that duplicates certificates and sends them to an unintended third party. Isolating the module within a private Ethernet link is still another way to protect the module from attack via the download link.

A fourth type of interface defines how the module interoperates with end-users. One of the roles of a module is to provide certificates to users on smart cards. A smart card is an example of an "out-of-band" communication, which is considered to be a more secure method for delivering key materials since delivery does not rely on the same network those materials are intended to protect. It is also, as previously stated, an example of two-level security whereby the user must not only know something (i.e. a PIN code), but also have something (i.e. the smart card) in order to come into possession of the key materials.

Flexible Form Factors

Security modules operate at multiple levels in the PKI hierarchy - CA, RA and end-user - and provide a variety of PKI-related functions, including custom applications the organization writes itself and downloads to the module. This means that different modules should be equipped differently. Not all modules, for example, will support end-user access so they don't require a keypad or a smart card reader. Moreover, some will be deployed as production modules, forever attached as a dedicated processor to a specific host, and therefore can be connected via a lower-cost PCI interface rather than via Ethernet. Availability of an "OEM" (Original Equipment Manufacturer) version - as opposed to a "fully populated" version - allows organizations to more cost-effectively deploy cryptographic applications widely throughout the enterprise.

Secure Remote Management

Because modules are distributed across a network, it makes sense that they be managed across a network as well. System initialization, feature enablement, software upgrades, module diagnostics and self-tests are all events that can be more rapidly and cost-effectively managed from a central station instead of physically visiting each and every module in the PKI. Of course, remote management - like any other network-based interaction with the security module - must be highly secure and tamperproof.

Who Do You Trust?

The bottom line is this: in a PKI, the security of any business transaction is only as good as the security modules on which it relies for the execution of security functions. Furthermore, those security functions are tightly interconnected to the day-to-day operation - so security itself cannot be the organization's only concern when evaluating alternative HSM solutions. Processing speed, flexibility, and scalability, as well as physical and virtual protection of certificates and key materials, all become part of the equation. If implemented properly a security module, and security architecture, should provide business enhancement through assured, effective transactions with customers and business partners. Lower performance should never be a trade-off for higher security. Select the right hardware security module for your PKI and it won't be.

The AEP Keyper Solution

AEP Keyper is a range of enhanced security products offering the highest level of security and performance in PKI environments, where the management and storage of cryptographic keys is fundamental. AEP Keyper is available in three modules - Professional, PCI and OEM. The Professional and PCI modules provide key storage, high-speed signature and key generation, while the OEM module offers a feature rich development environment allowing rapid delivery of custom secure applications.

AEP Keyper is ideally suited to businesses deploying a PKI where the protection of cryptographic keys is a priority, such as PKI Signing, Commercial CAs, transaction signing and code and document signing.

Contact AEP Networks

Corporate Headquarters	Government Solutions Group
AEP Networks 347 Elizabeth Ave., Suite 100 Somerset NJ 08873 Toll-Free: 1-877-638-4552 Tel: +1 732-652-5200	AEP Networks 40 West Gude Drive, Suite 200 Rockville, MD 20850 Toll-Free: 1-800-495-8663 Tel: +1 240-399-1200
Europe	Asia-Pacific
AEP Networks Focus 31, West Wing, Cleveland Road Hemel Hempstead Herts HP2 7BW U.K. Tel: +44 1442 458 600	AEP Networks 2107 Tower 2 Lippo Centre 89 Queensway Hong Kong Tel: +852 2845 1118
Japan	
JOYO Bldg 6-22-6 Shimbashi Minato-ku Tokyo 105-0004 Japan Tel: 81-3-3432-3336	

© AEP Networks, Inc. All rights reserved. The AEP Networks Logo is a trademark of AEP Networks, Inc., with registration pending in the U.S. All trademarks or registered trademarks mentioned in these documents are property of their respective owners. www.aepnetworks.com
info@aepnetworks.com