Protecting Commercial Secure Web Servers From Key-Finding Threats

nCipher, Inc. – Version 1

Contents

Introduction: A New Class of Threats

Web servers have emerged as a crucial component in every organization's electronic business (e-business) strategy. Increasingly, Web servers pass everything from valuable transactions to confidential customer information. Given the importance of e-business to the organization and the central role played by the Web server, the security of the Web server becomes paramount. As a result, secure Web server features are offered by many well-known vendors including Apache, Netscape/iPlanet and Microsoft.

If a secure Web server is compromised, the organization faces the potential for devastating financial losses, possible liability for the disclosure of confidential information, and disruption of key business activities and relationships. As a result, organizations have initiated strategies to secure their Web servers against commonly recognized threats. In particular, digital certificates are widely used to confirm the identity of the parties in Web-based transactions and to enable encryption of information delivered using secure Web servers. The counterparts to these widely distributed digital certificates are the private encryption keys, which must be kept secret by each party in order to prevent them from being impersonated by a third party. Most security models currently focus on keeping these private keys on the Web server itself, and then strongly protecting the server.

Recent research, however, reveals an entirely new class of threat - the key-finding attack. Through key-finding, an attacker is able to discover and make use of the private key, which is critical to the effectiveness of any digital certificate-based security strategy. Until now, application developers and security managers believed that if they encrypted the key and buried it in the very large disk storage of the server, it would be virtually impossible to find and compromise the key. We now know[1] that it is possible for an attacker to find and make unauthorized use of private keys much more easily than previously thought.

This paper reviews the issues surrounding the key-finding threat and analyzes current security practices in light of this new threat. It examines best security practices and offers a range of solutions based on an informed assessment of the threat to the organization.

Background: Adoption of Public Key Cryptography

Most secure e-commerce sites today rely on cryptography and a Public Key Infrastructure (PKI) for security. A PKI enables users of a public network such as the Internet to exchange data and undertake financial transactions securely and privately. It does so through the use of a pair of cryptographic keys, a so-called “public key” and a “private key”, to encrypt and decrypt messages using a common cryptographic algorithm. The public key is distributed widely within a digital certificate, which has been endorsed by a trusted authority to authenticate the owner.

The private key, however, is closely held by the organization securing the asset, such as the Web server, and is never shared with anyone or sent across the Internet. Thus,

1 "Playing hide and seek with stored keys", Dr. Adi Shamir & Dr. Nicko van Someren, Financial Cryptography 1999, Anguilla BWI - http://www.ncipher.com/products/files/papers/anguilla/keyhide2.pdf

if the Web server encrypts a message to you, an e-commerce customer, using its private key, you can use the server's public key, which you extract from its digital certificate, to decrypt the message. Since the certificate is endorsed by a trusted source and no one else has the same private key, you have effectively authenticated the Web server. However, if someone could steal that private key, he/she can impersonate the Web server in a secure transaction and you, the customer, wouldn't even know it. Furthermore, any previous transactions conducted by that server could be decoded. Such a situation could create costly havoc.

There is a substantial PKI industry to support the use of public-key cryptography and numerous vendors offer products that allow organizations to deploy PKI security. It is, however, the responsibility of the organization itself to protect its own private keys.

Challenges of Managing Keys in Software

Security managers face two major key management challenges:

- The need to protect the private key to ensure it is not compromised.
- In the event that the private key is compromised, the organization must replace not only the private key but all copies of the corresponding public key as well.

Security managers have a number of strategies, described below, available to them to protect the private key, from simply following best security practices to storing the key in a separate, hardware security module. If there are many outstanding copies of the organization's public key, then replacing them is a daunting task, significantly disrupting customer and business relationships.

Until now, security managers have considered the private key to be quite secure within a Web server. They have assumed that the encrypted key, a very small piece of data, was safe when encrypted and stored within the vast data filing system of the Web server. They have further relied on the fact that finding the private key within the memory of the Web server was tantamount to finding the proverbial needle in a haystack. But, as noted above, recent research has proven this to be false. A determined attacker can find the key.

Key-Finding Threat

Key-finding describes a threat by which an unauthorized user can find the private key used in a cryptographic security scheme. Once the key is found, the unauthorized user can impersonate the legitimate user in electronic transactions.

The key-finding threat occurs in the following way: Typically, in a commercial Web server, the key is encrypted and stored within the server, where it must be decrypted before it can be used. Since the key is only a few hundred bytes long and the storage space of the server may be many tens of gigabytes, conventional reasoning argues that an attacker is unlikely to ever find the key.

In practice, however, finding the key in a mass of data is not as hard as security managers have long assumed. The keys to the type of cryptographic systems used in secure Web servers are unusual numbers with specific mathematical properties that make it possible for an attacker to identify them. When carrying out a key-finding attack, the attacker need only look for these special characteristics.

To find a key, the attacker needs to be able to read the memory of a Web server process or, alternatively, to trigger a "core dump" and read the resulting diagnostic file. In an ideal world, where operating systems were truly secure, this would be difficult. In practice most operating systems provide some mechanism for one process to access the memory of another process, such as when using debugging tools.

Of course, the attacker must get permission to read the memory or the core file. It turns out that, even in the case of secure Web servers, this is often easier than security managers might think. In general, most operating systems will allow one process to access the memory of another process if the two processes belong to the same user. Generally, it is not difficult for an attacker to acquire the identity of a legitimate user: for example, CGI scripts usually share the same user identity across multiple processes. Therefore, once the attacker has found and copied the key, the Web server and its customers are open to the vulnerabilities previously discussed. Complicating efforts to thwart the key-finding threat, security managers face two different types of attackers: internal attackers, such as disgruntled employees, and outsiders, such as a hacker. Of the two, the internal attacker is the more common and more difficult to defend against – some reports suggest that up to 80% of security breaches are carried out by employees – as a result, the key-finding threat will most likely come from an internal source.

Key-Finding Defenses

Fortunately, security managers have a number of defenses against both external and internal key-finding threats, and the leading Web server vendors have designed their products to make it easier to defend against this threat. These defenses revolve around best practices, software defenses, and hardware defenses.

Best Practices

The first step in building your defense against the key-finding threat is to follow best security practices [see sidebar: Best Security Practices for Key Safety]. Start by determining your level of risk based on the value of the information you are protecting and the cost of replacing it should the information be lost or destroyed. Armed with this information, you can then make the necessary decisions based on a realistic cost-benefit risk analysis. This assessment should be a standard part in the development of a security policy.

Based on the cost-benefit analysis, security managers can then determine the level of key protection they require. The common practice is to identify a Web site as requiring

low, medium, or high security. Anything residing on a low-security site must be considered of such minimal value that it is utterly disposable. Thus, a low-security server does not even need authentication and is not a candidate for public key cryptography.

Software Defenses

Medium-level security can be achieved by following best security practices and fully implementing the security software capabilities built into the leading Web servers, such as Microsoft IIS, Netscape/iPlanet Enterprise Server, and Apache servers. In addition, security managers will need to keep current with the latest security patches and pay attention to the vendor's latest security bulletins.

For medium-level security, managers have a number of software-based measures they can take. These include the implementation and management of effective password and authentication programs, including the frequent changes of passwords and the use of passwords that are difficult to crack (such as alpha-numeric passwords not found in a dictionary). Managers also should adopt a multi-layered security strategy involving the following

- Use of firewalls, intrusion and detection software
- Frequent auditing
- Disallowing executable code uploads
- Limiting the number of people who have administrative rights on the server
- Implementation of all the security features in their Web server

Leading Web servers offer considerable ability to protect against a key-finding threat, such as ensuring that CGI scripts - a primary vehicle for gaining access to the server's memory - can only be executed from specific parts of the Web site. The best software measures, however, do not provide ironclad security. Even encrypting the key cannot ensure that a determined attacker, particularly an internal attacker, will not be able to find and use the key.

Hardware Defenses

Hardware defenses provide the most effective security against key-finding threats. High-level security installations should follow best security practices and implement all software-based security defenses, but they will normally go one step further by adding secure hardware for key storage. High-level security is increasingly necessary for organizations with valuable e-business sites, which present an attractive target for determined attackers and, therefore, must be protected in a more comprehensive way. High-level security includes the following:

- Use of background checks and periodic reinvestigations of administrative personnel

- Measures to secure the premises physically using entry and exit control, photo IDs, or baggage searches

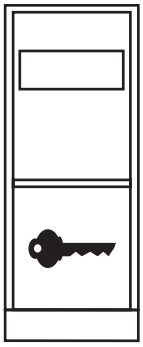
- Implementation of procedures such as a two-person rule in sensitive areas

- Implementation of key sharing, which requires the joint participation of more than one person to start the secure application and, therefore, decrypt and run the private key

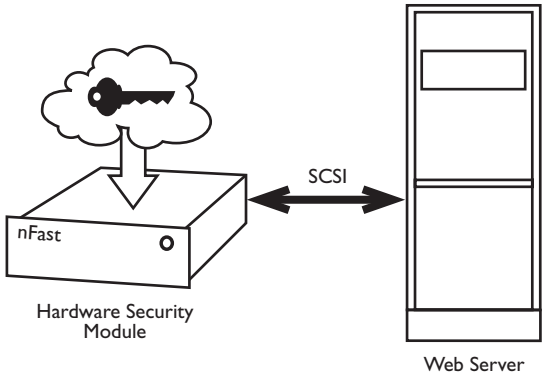
- Use of biometric hardware to identify individuals

- Deployment of hardware security modules to remove keys physically from the server

In the face of the key-finding threat, the last point is particularly important. For both high-level and many medium-level security situations, security managers should insist on the storage of private keys outside the server in separate, dedicated, secure hardware modules.



Key Stored in Software on Web Server – Making It Vulnerable to a Key Finding Attack



Encrypted Key Securely Stored in Hardware

Hardware-based security entails storing the private key outside the Web server. The key is stored in a cryptographic hardware security module (HSM) that is tamper resistant and, therefore, impervious to physical attacks. Because HSMs don't have an operating system, they are also safe from software threats. The most advanced HSMs have been independently validated by a US Federal security standard, FIPS140-1, which significantly increases the level of trust in a PKI system. An attacker will have no opportunity to find and decode the key.

Most HSMs are also designed to prevent unauthorized access by allowing the organization to control which users can use a key, and which users can move keys in and out of the hardware. In addition, some advanced HSMs support key sharing which requires the participation of multiple authorized users, for example two out of three, before a private key can be accessed. This greatly reduces the possibility that a single individual could launch such an attack, since operational responsibility is spread across multiple people. With a correctly configured server and an advanced HSM, the risks of unauthorized key removal is negligible. This is why the major Web server vendors all support HSM key storage.

Conclusion: Security and Performance

Now that research has proven that it is possible for an unauthorized intruder to find and compromise a private key, it is only a matter of time before a key-finding attack succeeds against an unprepared Web site. Even if nothing of value is actually taken or destroyed, the cost of such an attack, just in the time, expense, and disruption entailed in changing keys is very high. The negative publicity impact of such episodes can also be very high, as consumer concern about Internet security continues to run high. If e-commerce sales are actually diverted or confidential information is stolen, the costs and liabilities can be enormous.

Fortunately, organizations can defend against the key-finding threat straightforwardly. It requires following standard best security practices and fully implementing the security capabilities provided by leading Web server vendors. Most of all, it involves adopting an advanced HSM solution such as provided by nCipher. The cost of a high-performance HSM is insignificant compared to the value of the assets at risk and the cost of a successful key-finding attack.

In the wake of publicity around the key-finding threat, vendors will come forward with HSM solutions that appear to address the problem. When evaluating those solutions, however, security managers must look not only at the security issues but at Web server performance as well. As much as managers need to secure their Web servers from key-finding threats, they also cannot allow e-business systems to be hindered by slow HSM solutions. nCipher is the only vendor to offer an advanced HSM supporting administrative key sharing as well as the high performance needed to gracefully support the peak levels of secure transactions enjoyed by a successful e-Commerce Web site.

Best Security Practices for Key Safety

- **Assess value at risk from key-finding threat**
- **Determine necessary level of security (low, medium, high)**
- **Implement effective password and authentication process**
- **Implement and enforce audit procedures**
- **Recognize internal as well as external threats**
- **Implement all appropriate security features in the server**
- **Eliminate unnecessary services and tightly restrict server processes**
- **Isolate executable scripts and other programs**
- **Implement key sharing (multiple individuals required to run key)**
- **Implement HSM for key storage outside the server**
- **Consider performance & scalability needs alongside security**

Web server vendors may offer very detailed checklists for configuring and maintaining a secure Web server installation. Microsoft, for example, provides a list of NT and IIS configuration settings to achieve a medium security level. The settings can be found at the following Web address: http://www.microsoft.com/security/products/iis/CheckList.asp

For high security installations, Microsoft recommends hardware security modules such as nCipher's nFast/KM and nFast/CA. According to Microsoft, HSMs allow Web server managers the comfort of knowing that the keys are physically removed from the server and cannot simply be compromised by software attacks.

nCipher Product Information

nCipher is the recognized world leader in high-security and high-performance HSM solutions and has conducted extensive research, including demonstrating how a key-finding attack can be carried out.

nCipher's key management and high performance cryptographic acceleration solutions provide a secure hardware environment beyond the reach of almost any attacker. In the nFast key management environment, organizations can safely store and use their private key to encrypt and decrypt data or sign digital certificates.

The nFast Family of Hardware Security Solutions Provides:

- Secure key management with strong physical security
- Fast, flexible and secure key signing operations
- The world's highest digital signature throughput: scalable up to 2000 x 1024-bit public key signatures per second
- True hardware-based random number key generation
- Controlled access to private key material through nCipher key management architecture
- Extra security through support for key sharing
- Easy integration through common APIs such as PKCS#11, CryptoAPI and NSAPI
- Compatibility with all relevant Internet protocols (e.g. SSL, TLS, SET, IPSec, S/MIME) and support for digital signature algorithms including RSA, Diffie Hellman, and DSA.
- Compatibility with all leading Web servers (Microsoft, Netscape/iPlanet, Apache)
- nCipher products have prior validation to Federal Standard FIPS 140-1, Level 3
- Choice of form-factors: Fast SCSI-2 bus/CD-ROM drive 5.25” form-factor fits inside the standard server enclosure or rack; short-form PCI card fits standard PCI expansion bus slots

Contact information:

nCipher, Inc.
500 Unicorn Park Drive
Woburn, MA 01801
USA
Telephone: 1-800-NCIPHER
Telephone: (781) 994-4000
Fax: (781) 994-4001

nCipher, Inc.
2955 Campus Drive, Suite 400
San Mateo, CA 94403-2507
USA
Telephone: 1-877-NCIPHER
Telephone: (650) 295-7850
Fax: (650) 295-7700

nCipher, Inc.
57 Crystal Avenue
West Orange, NJ 07052
Telephone: (973) 325-3214
Fax: (973) 325-0141

nCipher Corporation Ltd.
Jupiter House
Station Road
Cambridge
CB1 2JD
UK
Telephone: +44 1223 723600
Fax: +44 1223 723601

Visit the nCipher Web site at
http://www.ncipher.com for the latest
product news and in-depth information.
Or, e-mail (USA) ussales@ncipher.com
(Europe and the rest of the world)
sales@ncipher.com

KFWP1199