GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Security Essentials: Network, Endpoint, and Cloud (Security 401)"
at http://www.giac.org/registration/gsec

Case Study: Implementing EFS within a Microsoft PKI.

GIAC Security Essentials
Certification (GSEC)
Practical Assignment
Version 1.4b

Option 2 - Case Study in
Information Security

Submitted by: Micha Prvulovic
Location: Online

© SANS Institute 2005, Author retains full rights.

Table of Contents

© SANS Institute 2005, Author retains full rights.

Abstract/Summary

The possibility of laptop theft and the loss of confidential customer information was identified as a major risk by upper management and our company lines of business. This resulted in a project team being formed to create and implement a solution to mitigate the risks. This paper will explore the issue of laptop theft, and discuss our implementation of Microsoft's[1] Encrypted File System via a 3-Tier PKI solution.

Before

Current Security Posture

Customer data on laptops within our organization were not being protected sufficiently. Our standard PC operating system is Microsoft Windows XP Professional with service pack 1 installed. All PC/Laptops in our Enterprise receive automatic Windows and Antivirus updates from MacAfee[2]. Hard drives are formatted with NTFS, and a standard corporate image is loaded. BIOS passwords are not used. PC/Laptops are protected by a complex password policy which is pushed down to the clients via Active Directory. Additionally, Administrator and Power User accounts are restricted to support staff only. General users run as members of the regular Windows user group.

Clients are not restricted in the types of data they have stored on the hard drive of their laptop computers. Some of the laptop users are sales people who need to access their data on the road as part of their basic job functions. Additionally, the company is spread out across Canada and the US so business travel is quite normal and increased the risk of laptop theft.

Problem Description

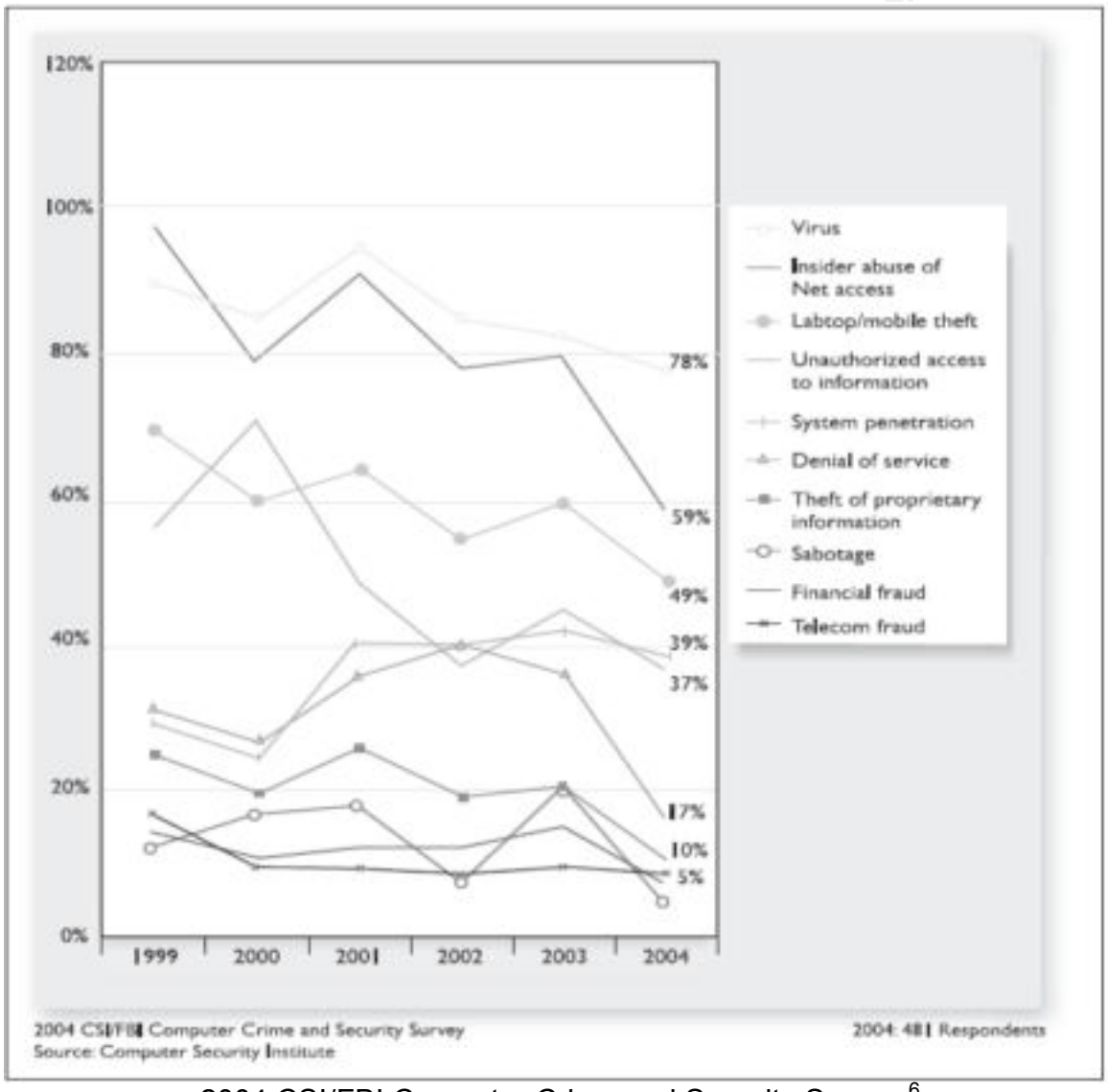
The problem facing our company is that data on a stolen laptop could be easily accessed by individuals using a variety of simple tools. Several password reset boot disks exist, such as EBCD[3] which would allow a thief to easily reset the password on a user or administrator account. Or a thief could simply remove the hard drive and install it in a new machine, and mount the hard drive to recover the information stored there.

Laptop theft is not a new problem. In fact, the FBI[4] and CSI[5] (Computer Security Institute) have compiled the Computer Crime and Security Survey[6] which tracked the most common security issues reported by respondents. The survey was first released in 1999.

The survey details the responses of nearly 500 computer security professionals from many different companies, government agencies, universities, as well as financial and medical institutions

© SANS Institute 2005 Author retains full rights.

In 1999 laptop theft was the second most common type of security incident reported by the survey respondents. This year's survey which was released in March 2004 shows that laptop theft has decreased overall but is still ranked the third most commonly reported computer crime by respondents. It's interesting to note that virus attacks and abuse of network access are the only crimes ranked ahead of it. The graph shown below illustrates this trend. We can see that Laptop theft was reported by approximately 70% of respondents in 1999 and has fallen to less than 50% in 2004.



2004 CSI/FBI Computer Crime and Security Survey.[6]

Other interesting factoids I found during my research on laptop thefts include:

- *Over 98% of stolen laptops are never recovered. (FBI).*[7]

© SANS Institute 2005 Author retains full rights.

- *69% of the Fortune 1000 companies experienced Laptop theft. (Computer Security Institute/FBI survey).*[7]

- *1 out of every 14 laptops sold in 1995 have been stolen since they were purchased.* (Information Week, 2000).[7]

- *1 in 10 laptop thefts occur in airports.* (BBB Study of 2000).[7]

- *IT security personnel admit that 57% of all network security compromises had their origins in laptop theft.* (CSI Study).[7]

Replacing stolen equipment is only part of the problem however. The real risks are associated with the vulnerable customer data stored on the laptops. This information could be anything from confidential customer information, financial or product data or even trade secrets. Any company information could be potentially useful to criminal enterprises which makes the risks associated with laptops so great.

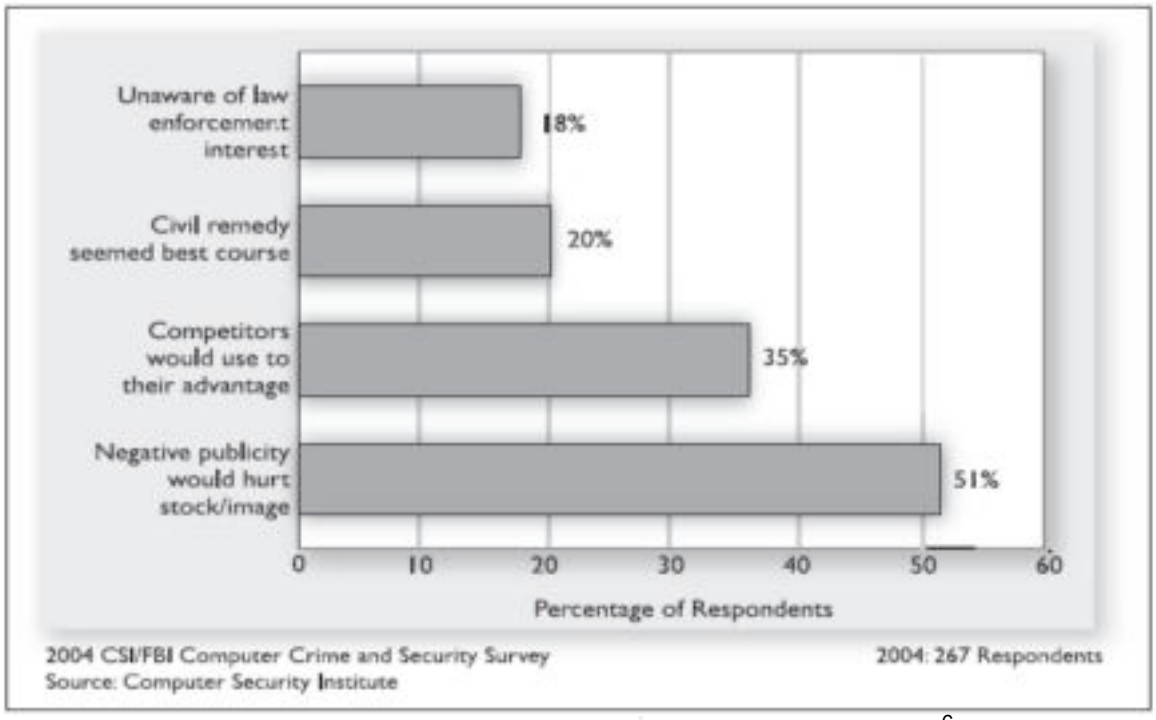
Current Risks

Customer satisfaction, client privacy, negative public perception, potential legal challenges and the risk of fraudulent criminal activities like identity theft are all legitimate risks identified with the lack of laptop data protection.

Customer satisfaction and client privacy were of particular concern. As a large service based organization with a public listing on the stock exchange, our reputation and how the public perceives our company is of the utmost importance. Our ability to attract new customers and keep share holders is partially based on the trust customers feel they can place in our business. This business/client trust could be irreversibly damaged if customer data was compromised because of theft.

The negative public perception that could result from lost or compromised customer data could be potentially damaging financially. Again referring to the FBI/CSI Computer Crime and Security Survey we can see that 51% of respondents reported that their company did not report computer crimes because of fear it would negatively affect their stocks or public image.[6]

© SANS Institute 2005 Author retains full rights.



2004 CSI/FBI Computer Crime and Security Survey.[6]

A good example of these risks was reported in a story published on March 25, 2004 by the online publication Information Week[7]. Paul McDougall reported on the theft of two laptops belonging to a division of GMAC Financial services[8].

The stolen laptops were taken from an employee's car at a GMAC office near the city of Atlanta. The information contained on the laptops included *"customer names, addresses, dates of birth, social security numbers, credit scores, marital status, and gender."*. [9]

Nearly 200,000 customers were affected by the theft. GMAC advised them in a letter to place fraud alerts on their credit files furthering the inconvenience suffered by the customers.

As reported in the article, one GMAC insurance customer noted *"if company guidelines deem it acceptable to house data on laptops, in parked cars, then I would question their competence…"* .[9]

This incident resulted in GMAC *"undertaking a comprehensive review of our security policies and procedures,"*.[9] Performing a review such as this is usually an expensive and time consuming proposition.

So in the end GMAC has learned an expensive lesson, suffered negative public perception and inconvenienced a large number of their customers, some of which will no longer do business with them. These are the problems we hoped to avoid by implementing data protection on our own laptops.

© SANS Institute 2005 Author retains full rights.

During

Proposed Solution

After many meetings between the lines of business and upper management we were presented with a long list of customer requirements that the project team would need to address. The following table lists the key requirements put fourth by our customers.

1.	*The solution must ensure that no additional user authentication is required over what exists today.*
2.	*Laptop users currently send copies of files to both Branch and non-branch servers for document sharing purposes. The project must ensure that such files are stored in a unencrypted state in the servers.*
3.	*The need for training/communication must be kept to an absolute minimum – by ensuring the encryption/decryption process is transparent to the user.*
4.	*If there are other users of the laptop he/she must not be able to access information that they are not authorized to view or use.*
5.	*If a laptop hard drive is removed and re-installed into another computer, the sensitive data will retain its encryption. This will prevent authorized access to data.*

After hearing several different solutions the lines of business settled on utilizing Microsoft EFS as it matched their requirements closest. To understand and support EFS, an understanding of basic cryptography is required.

Introduction to Cryptography

According to Wikipedia.org Cryptography is defined as; "the study of ways to convert information from its normal, comprehensible form into an incomprehensible format, rendering it unreadable without secret knowledge".[10]

Symmetric and Asymmetric encryption are widely used today for cryptographic functions. Briefly both types use keys, however the way the keys are used in the process are different. Symmetric encryption uses the same key to encrypt and decrypt data. Asymmetric encryption on the other hand uses a key pair which consists of two different but mathematically related keys, these keys are referred to as the public and private keys.

In symmetric encryption, the key that is used for encryption must also be used for decryption. The biggest problem with this is ensuring the secure delivery of this key to the recipient. If the key is intercepted or stolen, all data encrypted with it are now compromised. The advantage to symmetric encryption is that it is often much faster than asymmetric encryption.

When data is encrypted two parts are required to complete the encryption. A key and a algorithm. An encryption algorithm determines how the data is changed

© SANS Institute 2005 Author retains full rights.

during the encryption process. The same algorithm must be used to encrypt and decrypt the data.

Typical symmetrical algorithms in use today are DES, DESX, Triple DES, and AES. DES was first developed in the 70's while DESX and Triple DES were both developed to increase the security of DES. AES (Advanced Encryption Standard) was developed as a replacement to DES and it's variants which have begun to be phased out. The US government adopted AES as their standard in 2000.

Common asymmetrical algorithms used today include RSA (Rivest Shamir Adleman) and DSA (Digital Signature Algorithm). These algorithms are considered more secure than the symmetric algorithms shown above, however the encryption process is much slower.

To take advantage of the strengths associated with symmetric (speed) and asymmetric (security) encryption the two are often combined. To do so, the symmetric key is used to encrypt and then the symmetric key itself is encrypted asymmetrically. This ensures that only the intended recipient can decrypt the symmetrical key by using their private or pubic key.

Introduction to EFS

Microsoft's Encrypting File System (EFS) provides transparent encryption capabilities to users of Windows 2000, Windows XP and Windows Server 2003. While it provides file confidentiality it does not perform any authentication protection or integrity checking. The ability to recover a users data or key is also available just in case a user key is lost or corrupted.

File or folder encryption with EFS is simple once you get the acronyms out of the way. When a person wishes to encrypt a file, a randomly generated FEK (File Encryption Key) is created by their computer. Next, a symmetric encryption algorithm is used to encrypt the file using the FEK as the symmetric key. The computer then retrieves a users EFS certificate (either Locally or from a Domain) and extracts the users public key and encrypts the FEK asymmetrically with it. Once this process is completed, the FEK is stored in the files header in a section called the DDF (Data Decryption Field).

For recovery, the computer retrieves a recovery agent certificate (either Locally or from a Domain) and extracts the public key. Next the FEK is asymmetrically encrypted with the recovery agents public key and the encrypted FEK is placed in the file header in a section called the DRF (Data Recovery Field).

When a user needs to decrypt the file or folder, the computer will again retrieve the EFS certificate, but this time it will extract the users private key and decrypt the DDF. The FEK is then used to decrypt and deliver the file in plain text.

© SANS Institute 2005 Author retains full rights.

In the event a user's key is lost or corrupted the computer can retrieve the EFS recovery agent certificate, decrypt the FEK in the DRF and restore the file in plain text. All these processes happen transparently to the user. This is one of the benefits of EFS as it provides seamless encryption / decryption to the end users.

The type of Microsoft Windows installed on a computer determines which encryption algorithms are available for EFS encryption as seen in the following table.

Microsoft OS	Encryption Algorithm	Relative Security Level
Windows 2000	DES-X	Low
Windows XP, no Service Packs	Triple DES (3DES)	Medium
Windows XP w/SP1 +, or Windows 2003 server	AES (Rijndael)	High

Before we could design and implement the PKI it was important to understand Microsoft's implementation of a PKI.

Introduction to PKI

PKI stands for Public Key Infrastructure. A PKI can be described as a frame work which can be used to register, verify and authenticate a user, computer, or service's digital identity. The purpose of the PKI is to build and enforce trusts. Currently there are many ways to implement a PKI solution available from plenty of vendors including Microsoft, Entrust, Verisign and RSA. Some of the common pieces shared between the various PKI's available are Certificates, Certificate Authorities (CA) and Certificate Revocation Lists (CRL).

Certificates are an important part of a PKI as they act as the digital credentials issued by a CA server. Certificates usually contain a fair bit of information. Typically they contain; the location of their private key, which CA issued them, their public key itself, and the types of encryption algorithms they support.

Certificates are issued by Certificate Authorities. CA servers are an essential component of a PKI. Without a CA the PKI would not be able to verify, issue, or revoke certificates. The CA manages the certificates and CRLs within the PKI.

A CRL or Certificate Revocation List is a list of revoked certificates carried by the CA. The list details the serial number of a revoked certificate, the date the certificate was revoked and a reason code.

In a Enterprise PKI, you will typically find multiple CAs. These CAs will be organized into a CA hierarchy consisting of a single Root CA and multiple Policy and Issuing CAs.

© SANS Institute 2005 Author retains full rights.

Microsoft recommended we use a 3-Tier solution to meet our PKI needs for EFS as they believe it was the best balance between security and scalability in our environment. A typical Microsoft 3-Tier solution consists of an offline Root CA server at the top of the hierarchy (1st tier), followed by one or more offline Policy CA servers on the 2nd tier, with multiple online Issuing CA servers on the 3rd tier.

The reason for keeping the Root CA and Policy CA offline is to add to their assurance levels by removing the threat of network based attacks. Physical security is very important in maintaining these assurance levels.

Since we already had an Entrust[11] Root CA server available we decided to use it as our Root CA. We would implement a single offline Policy CA and a single Issuing CA. During the solution implementation these were referred to as the Entrust Root, the Intermediate CA (Policy), and the Subordinate CA (Issuing). Before I continue, please note that I will be referring to the Intermediate and Subordinate CA's as IntCA and SubCA for simplicities sake.

The SubCA server would join our existing Active Directory forest as this would allow us to issue certificates to laptops that were part of our domain. This would also enable us to remove the Domain Administrators from the role of Recovery Agent (default in EFS) and bring it under the management of the security department. This would help segregate the duties of the various teams that would be supporting the infrastructure.

To add further security to the implementation it was decided that a Hardware Security Module (HSM) would be used to secure the cryptographic keys of the IntCA. An HSM was purchased from nCipher.[12]

Once the planning sessions were over, and all support teams were identified a proof of concept was prepared. Once that was successfully completed, we moved to the QA phase of the project. The QA environment was made to match production as closely as possible. As QA sign-off was completed it was time to move to the implementation of the production environment.

Solution Implementation

The first step of the implementation was the easiest. Our Entrust Root CA was already built and running in the production environment and was directly supported by my department. On the other hand, the IntCA (Policy) and SubCA's (Issuing) had to be built from the ground up. As I had already built the QA servers, the production installations let me test the procedures I had written earlier in the implementation.

A high level overview of the installation follows and will focus on the key areas of the implementation instead of offering a step by step guide. Furthermore, the hardware setup and installation of windows was almost identical.

© SANS Institute 2005 Author retains full rights.

In total, six IBM[13] xSeries 345 were purchased for the project. Two were used for the production environment, two for the QA environment and the two remaining servers would be stored at a backup site for disaster recovery purposes. The servers featured dual Intel Xeon processors, 1GB of RAM, duel integrated 10/100/1000 Ethernet and hot swappable power supplies, hard drives, and fans. We had four 36GB hard drives per server.

The servers were placed in our secured environment. To access the servers I would now need a manager sign-off and two witnesses to verify any of my actions while logged onto the servers. This was done to ensure these servers would meet the highest assurances. All of the following steps were witnessed and required all of our personal sign-offs at the end.

The first steps I took was to apply two firmware revisions to the servers. The BIOS was updated to version 1.16 and the Integrated Systems Management Processor was updated to version 1.09. This one done to correct errors that were discovered in our proof of concept. Two RAID arrays were configured; drives 0 & 1 were set at RAID 1 while drives 2 & 3 were made hot swaps. Since the intermediate server was an offline server, it's network adapters were disabled in the BIOS.

The next step was to install Windows Server 2003 Enterprise edition on the servers. Both servers were installed offline without any network connectivity. Two separate complex Administrator passwords were created for each server. Once the installation was completed, I applied all patches that were available at the time. MacAfee Virus scan Enterprise edition was installed on both servers, with updated Engine and DAT files.

The last step of the OS install was to harden the servers and schedule a vulnerability assessment. The Microsoft Security Configuration and Analysis Tool (SCA) console was used to apply the following templates to the Intermediate CA server;

- The 'Enterprise Client – Domain.inf
- Enterprise Client - Member Server Baseline.inf
- Enterprise Client – Certificate Service.inf

While these templates were applied to the Subordinate CA;

- The 'High Security – Member Server Baseline.inf
- High Security – IIS Server.inf
- Enterprise Client – Certificate Service.inf

In addition to these templates, the following steps were also taken;

- BIOS passwords were enabled and set to comply with company policy.

© SANS Institute 2005 Author retains full rights.

- Remote logging was enable on the Subordinate CA.
- The local Administrator accounts was;
  - Disabled and renamed.
  - Descriptive text was removed.
- A dummy Administrator account was created;
  - Complex password was created.
  - Removed from all user groups.
  - Descriptive text was added.
- Guest account was renamed, and descriptive text was removed.
- The SAM file created in the %systemroot%\repair directory was deleted as it contained our original Administrator password used during setup.
- All unnecessary services were stopped and disabled.

To ensure the integrity of the servers, a vulnerability assessment was completed by members of the Information security team within our organizations. The servers passed without major tweaking based on the hardening we performed, the physical security in place, and the extra precautions that existed for accessing the servers.

Next I had to begin configuring the IntCA and SubCA servers and here is where things began to differ. To begin with, the IntCA would be protected by the HSM purchased from nCipher. The IntCA was powered down and the HSM was attached via SCSI. The HSM has two modes, an operations mode for regular use and a pre-initialization setup mode. I switched to pre-initialization mode and ran the setup software.

The nCipher software would enable me create administrator and operator cards, and begin the configuration of Certificate Services on the IntCA server. This is referred to as "Creating the nCipher security world" by nCipher[12]. As part of the software installation, the private keys of the IntCA are removed and placed on smart cards. A simple application wizard was run to create six administrator cards and six operator cards, each with a long, complex, pass phrase assigned to it. The administrator cards are only used to create more operator cards in the future. The operator cards are now required for administering the certificate servers on the CA. For added security, any support person would now be required to use two operator cards out of the six to gain access to the certificate services.

Without an operator card they would not be able to start or stop the service or make any configuration changes. The cards were split into three sets of four (two admin, two operator) and were stored in safety deposit boxes at three different company sites. The keys to the deposit boxes were kept in the dual custody of several different managers and would require their assistance to gain access to the cards.

© SANS Institute 2005 Author retains full rights.

Once that was completed, the Certificate Services installation was initiated. The first step was to create a certificate signing request file (CSR) as our CA was offline. This file was generated and exported to removable media. The CSR was physically taken to our Entrust Root CA and was signed by the Entrust RA (Registration Authority). Once this was completed, I copied the signed certificate, the Entrust Root public certificate, and the Entrust CRL back to the removable media. These would be needed to complete the chain of trust between the Entrust Root CA and our IntCA servers. The three files were returned and manually installed into the IntCA's local certificate store. This process required the use of the HSM operator cards as the Certificate Services service was stopped and restarted several times.

As the IntCA server was offline, it would not be able to connect to the Active Directory domain to automatically publish it's own CRL. Normally a registry key is created automatically but would now require manual user intervention. From a command prompt I ran the following command to set the registry;

"certutil.exe –setreg ca\DSConfigDN CN=Configuration,DC=adroot,DC=*our_domain_name*,DC=net"

Additionally, the IntCA's CDP (CRL distribution point) and AIA (Authority Info Access) would also need to be modified to reflect it's offline nature. Using the Certificate Authority tool found under Administrative Tools, I brought up the properties page of the IntCA and clicked the extensions tab. Here I removed all locations paths except the "local" location path for both the CDP and AIA. Since the IntCA's CRL would be manually published to Active Directory, I added our LDAP location path to both the CDP and AIA. After these steps were completed our IntCA server installation was complete and I was ready to move onto the SubCA server.

The SubCA server differed as it was joined to our Active Directory domain. This was performed by members of our AD team as I did not have the appropriate rights. Once the SubCA had joined AD I moved in to configure the system. Like the IntCA server, the Subordinate needed to be signed, but this time it would be by the IntCA. The certificate signing request file was created and exported to removable media. The removable media was brought to the IntCA and signed. I then copied the signed certificate, the IntCA's public certificate and CRL back to my removable media. Then I returned to the SubCA and installed them manually and placed them in the Subordinate's local store. This ended my role in configuring the servers.

After

Solution Testing and Validation

Testing was done in four rounds. Once during proof of concept, then a more rigorous QA cycle, followed by a pre-production cycle and finally a pilot phase of the project where the solution was rolled out to a limited number of users. We

© SANS Institute 2005 Author retains full rights.

tested the clients ability to request EFS certificates from the CA servers, along with the ability to encrypt and recover data. For the most part the line of business was pleased. All the encryption appeared transparent and did not effect the clients ability to login or off the laptops.

In the end only a few issues were identified. A script was developed that would automatically encrypt a group of specified folders on a laptop after an EFS certificate was successfully requested. This lead to some extended boot times but was deemed acceptable by the lines of business.

Another issue was identified as a result of the clients laptop data being synchronized on a company server when they logged on or off the company network. The first time this synchronization occurred resulted in extended boot times. The synchronization was added to allow the Helpdesk a first level of file recovery. Any time a laptop user logged onto the company network, their files were backed up in plain text to a company server. This was implemented to prevent an influx of calls to the recovery agents.

Risk Assessment

As the project came to a close it became clear that although EFS provides a basic level of file and folder level encryption it also provided a false sense of security among the user base. Two limitations of EFS became readily apparent to the project team.

The 1st issue identified was the inability of EFS to encrypt system folders. Many applications would decrypt a file the user was accessing and store them in clear text in one of several temporary locations on a users hard drive. These temporary folders are marked as system folders and thus were not candidates for encryption. Without the encryption the files or parts of files stored in the temp folders were still vulnerable.

The second problem identified was that EFS in itself was only as secure as the clients user account. If the user account password was compromised all files would be decrypted transparently to the thief. As previously stated, hacker tools like Linux boot disks are routinely available to reset Administrator and user accounts which would grant the thief access to the data. I believe tying encryption to the user account weakens EFS overall.

Conclusion

Overall, I was pleased with the implementation of the PKI environment. It was a great opportunity and introduction to cryptography for me. However I was not totally satisfied with EFS and it's limitations. Several new products are coming of age that promise to encrypt the entire contents of a hard drive. I believe these new solutions could be a potentially better solution in the end.

© SANS Institute 2005 Author retains full rights.

References

Komar, Brian. Microsoft Windows server 2003 PKI and Certificate Security. Redmond, Washington: Microsoft Press, 2004. (3-10. 17. 29-31. 69-70.)

No Author listed. Cryptography. [Online] Available http://en.wikipedia.org/wiki/Cryptography/, October 25, 2004.

No Author listed. DES. [Online] Available http://en.wikipedia.org/wiki/DES/, October 25, 2004.

No Author listed. DESX. [Online] Available http://en.wikipedia.org/wiki/DESX/, October 25, 2004.

No Author listed. DESX. [Online] Available http://en.wikipedia.org/wiki/Triple_DES/, October 25, 2004.

No Author listed. DESX. [Online] Available http://en.wikipedia.org/wiki/AES/, October 25, 2004.

No Author listed. DESX. [Online] Available http://en.wikipedia.org/wiki/RSA/, October 25, 2004.

No Author listed. DESX. [Online] Available http://en.wikipedia.org/wiki/DSA/, October 25, 2004.

Quinton, Reg. Markan, Stephen. Windows 2000/XP Encrypted File System. [Online] Available http://ist.uwaterloo.ca/security/position/20020619/paper.pdf, October 30, 2004.

No Author Listed. How to Cite Internet Resources. [Online] Available http://205.146.39.13/linktuts/bgcite.htm, November 5, 2004.

1) Microsoft URL: http://www.microsoft.com/, October, 2004.

2) MacAfee URL: http://www.mcafee.com/, October, 2004.

3) Kupchik, Mikhail. EBCD – Emergency Boot CD. [Online] Available http://ebcd.pcministry.com/, October 27, 2004.

4) FBI URL: http://www.fbi.gov/, October, 2004.

5) CSI URL: http://www.gocsi.com/, October, 2004.

6) Gordon , A. Lawrence. Loeb, P. Martin. Lucyshyn, William. Richardson, Robert. CSI/FBI Computer Crime and Security Survey. [Online] Available http://i.cmpnet.com/gocsi/db_area/pdfs/fbi/FBI2004.pdf, October 15, 2004. (9. 14.)

© SANS Institute 2005 Author retains full rights.

7) What if your laptop gets stolen? [Online] Available http://www.ztrace.com/FactsPage.asp, October 20, 2004.

8) Information Week URL: http://www.informationweek.com/, October, 2004.

9) GMAC URL: http://www.gmacfs.com/, October, 2004.

10) McDougall, Paul. Laptop Theft Puts GMAC Customers' Data At Risk. [Online] Available http://www.informationweek.com/story/showArticle.jhtml?articleID=18402703, October 20, 2004.

11) Entrust URL: http://www.entrust.com/, October 2004.

12) nCipher URL: http://www.ncipher.com/, October, 2004.

13) IBM URL: http://www.ibm.ca/en/, October, 2004.

© SANS Institute 2005, Author retains full rights.

© SANS Institute 2005 Author retains full rights.