Global Information Assurance Certification Paper

Name: Tyler Tobin
Certification: GSEC
Smartcards: One stop shop? Deploying smartcards
Version 1.4.b option 2 – Case study In Information Security
Date Submitted: 10/10/2003

Outline:

I. Introduction: Smartcards: One-stop shop: good, bad and the ugly – Case Study

II. Introduction
- A. Abstract
- B. Approach and define the operational requirements of the technology
  - a. Define background/need for smartcard solution
  - b. List the criteria and define a global smartcard schema that includes a reusable solution
- C. Setting the stage (Before snapshot)
  - a. Risk defined
  - b. SPRINT process defined
    1. Basic risk analysis
    2. Quantify risk
    3. Manage risk
- D. Defining the stage (During snapshot)
  - a. Operating systems explored
  - b. Smartcard technical definitions
  - c. Measuring the possible solutions for the smart card deployment
- E. After snapshot: The state of security
- F. How we used a combination of vendor software and hardware to effectively create a total solution package.

III. Conclusion

Abstract

Security analysts are constantly looking for solutions to solve a number of security related issues. Analysts are looking for ways to reduce and organize identity chaos, increasing the use of non-repudiation and easily restricting user access to data. It is the specific smartcard scheme and general functionality that is difficult to define and deploy, but may provide the answer for security analyst striving to secure an enterprise. How can interoperability issues be securely managed, realized and set up within an organization employing tens of thousands of employees?

To understand this case study I will identify the authentication process associated with the deployment of smartcards from start to finish and the role that I played in it. I will identify the proliferation of corporate pressures that shaped and influenced my deployment and identify the results of a post-smartcard environment.

Background

Technology is always evolving and changing. The technology pendulum swings from side to side and corporate America is trying to adjust its' future to the technological tides. According to the Information Security magazine, "smart cards are a part of the European culture and business system," says Jason Wright, security technologies program leader at analyst firm Frost & Sullivan. Andy Briney's article "A Smart Card for Everyone" provides an interesting argument that supports my decision to develop a smartcard ecosystem. He states that "… in the U.S., it's going to require a whole shift in dynamics. You're going to have to get the card readers out, you're going to have to get people to understand the technology, and you're going to have to get them to break from what they're already using." (Briney) This is the task that I set out to do in my organization. I needed to find the lowest common denominator in introducing a decade old technology to a bleeding edge company.

Briney suggests that a change in the American corporate culture will aid in the success of smartcard deployment. I believe that the corporate change will in part, be driven by the Gramm-Leach Bliley Act of 1999 that strictly enforces data security and how data is accessed and stored by not only financial institutes but also companies that perform "lending, brokering or servicing any type of consumer loan, transferring or safeguarding money, preparing individual tax returns, providing financial advice or credit counseling, providing residential real estate settlement services, collecting consumer debts and an array of other activities." (Briney) The need for a simple and secure deployment strategy for smartcards may not be too far in the future for many American businesses.

Smartcards were originally conceived in the 1970's and has been plagued by the definition of what a smartcard truly is. Smartcards are hardware devices that corporations can use to control how their employees personal information is stored and used. The simplest smartcard would be a read-only memory (ROM) payphone card. This form factor is the most difficult to replicate or modify

because the data is written when the smartcard is manufactured. More advanced smartcards use electrically erasable and programmable read-only memory (EEPROM) which like ROM provides for a permanent memory store. EEPROM is limited to the number of times that you can read and write to it before the electrical components become unreliable. With the decreasing cost of random access memory (RAM) smartcards with RAM can now manage several applications and passwords and use authentication and ciphering techniques that I will explore later. Perhaps the definitive high level definition of what a smartcard truly is may come from ISO/IEC 7816. “Smart cards are credit card-sized, often made of flexible plastic (polyvinyl chloride or PVC), and are embedded with a micro-module containing a single silicon integrated circuit chip with memory and microprocessor.” (ISO/IEC 7816)

To compound matters, there has been a recent explosion in the smart card arena. Smartcards now use plastics that “contain antibacterial agents” (Briney) and are manufactured to be water proof. Optical memory cards, integrated finger-print cards and wireless smartcards make claims to function as a smartcard. As I look for a solution for my corporation it becomes increasingly difficult to define what smartcard ecosystem I should deploy.

(Before Snapshot) Setting the Stage:

In an organization with tens of thousands of employees it can be difficult for a handful of over worked security analysts to redefine a successful ecosystem in which smartcard technology plays a significant role. Perhaps it requires a regulatory order from the Federal Government such as the Health Insurance Portability and Accountability Act of 1996 (HIPAA) that was signed into law on August 21, 1996 that prompts large organizations to action.

Regardless of external pressures, smartcards have not yet taken a foothold in the organization, until now. With advances in technology, smartcard utilization is now being given serious consideration as a line of defense in protecting corporate assets. As a security analyst I am responsible for managing risks and paying close attention to details during the life cycle of a product. The organization demands that I mitigate the risks of conducting business by calling upon smartcard technology. It is the use and definition of this technological risk that will help set the stage for the initial snap shot of my organization. The security mechanisms and underlying objects identified in figure1 map out the role for smartcards in the organization. I have tried to express the need for the consolidation of credentials as being the center of the organizations security architecture. The most obvious use for the smartcard is as a separate secure container for encryption keys with digital signatures and passwords, thus removing the keys from the users’ hard drive and increase the portability of employee access.

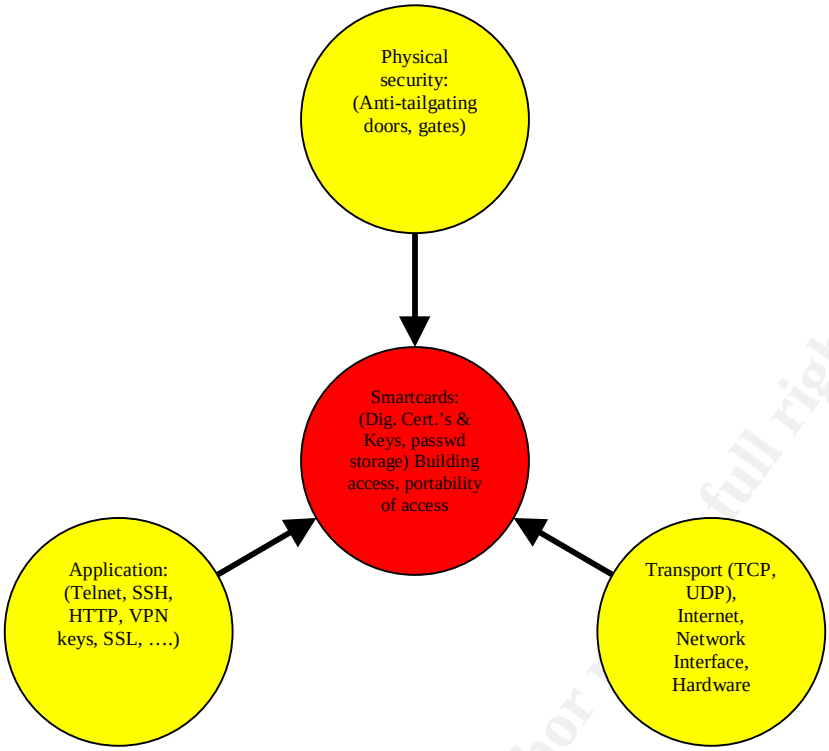

Figure 1 (Sub-component Break down)

Ultimately, the key business driver for my smartcard deployment was derived from management leaving their computers unlocked and unattended. As defined in the SANS Security Essentials with CISSP CBK, corporate policy is the first place to look at when setting direction or enforcing unwritten processes. "Enterprise-wide/corporate security policy" (Cole et al. 347) explicitly states that, "Associates with information security or sensitive information responsibilities are trained to use systems correctly, apply proper security controls, and understand potential risks." (Corporate Policy) Eventually a disgruntled employee noticed that the Vice President of Information Technology left their computer unlocked and unattended so it was quit easy for that person to send emails to other executive partners from this email account. Because the email was from a vice president, the impact sent shockwaves throughout the organization. One of the responsibilities for this executive's functions was to enforce and manage data security guidelines and for setting enterprise-wide security policies. The content of the email changed the direction of the corporation momentarily. For data security, this was considered the trigger event. Essentially after reviewing the incident, an executive committee recommended that if a smartcard would have been in place this breech would not have occurred. It was determined that

smartcards be put into practice and placed within security's tool belt to assist in preventing this type of incident from occurring in the future. During this period, the corporation was also placing large sums of money into anti-tailgating devices, identification proximity badges for building entry, and hiring security guards to patrol the newly constructed proximity fences. I think it would be safe to say that the corporation did not account for this top down breech of security.

Within a week of this incident, management declared that smartcard technology was obtainable and that the service request was going to follow the critical path. This was significant to the organization because it would increase the level of non-repudiation, manage passwords, provide for certificate storage and increase the portability of access. However, my primary goal was to manage the smartcard securely and understand the risks associated with this smartcard effort.

Corporations our size do nothing fast. Projects can take years to complete. Often times making a change in our organization is likened to steering the Titanic with a small paddle, it takes a while for the direction of the boat to change and the change can be catastrophic. It was my task to quickly outline the risk associated with implementing smartcard technology. To begin with, I needed to define a trigger to kick off a risk assessment. The trigger for using SPRINT (**S**implified **P**rocess for **R**isk **I**dentification) can be directly linked to the business risks associated with implementing smartcards into an environment that depends on other security control mechanisms. I choose the SPRINT tool to expose risk. SPRINT is a tool provided by the Information Security Forum which is dedicated to clarifying and resolving key issues in information security, to provide an initial "first-look" at any risks in a structured manner.

Basic risk analysis: Evaluating the risks associated with smartcard technology was addressed in two fundamental phases. Initially and at an abbreviated level, I considered the three bedrock principles in defending my organization; "confidentiality, integrity and availability." (Cole et al. 295) My organization defers to the" Federal Information Security Management Act of 2002" to define these three bedrock principles in which I would measure risk against;

- Integrity, which means guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity;
- Confidentiality, which means preserving authorized restrictions on access and disclosure, including means for protecting personal privacy and proprietary information and
- Availability, which means ensuring timely and reliable access to and use of information

For this service request I defined and outlined my risk profile using the bedrock principles to coincide with the business goals. My matrix considered the following business consequences for using smartcards:

- Technology Risk: (High level support areas) which means not being able to produce a smartcard solution that is consistent with corporate support standards. Smartcard technology is not standardized and the solution presented may be obsolete before we can implement the cards and/or technical solutions;
- Threats and vulnerabilities (Product Risk), which is defined as the way in which the organization implements the smartcard solution. For example employees may choose to store "private personal" i.e. (AOL passwords) credentials/keys and call the corporate help desk if their credentials become corrupted;
- Critical Business applications (Development Risk), which means considering the return on investment (ROI) for smartcard implementation and support;
- Public Confidence: (Employee Risk), which means a dependence on the willingness, experience and ability for employees to use the solution presented.

Quantifying risk analysis: To further define my risks I also outlined the technical role smartcard technology could provide in my organization:

- Reduced sign-on (password sharing/storage)
- Data integrity (digital signatures, digital keys – PGP, SSH, VPN etc.) exploitations from local source (users hard-drive)
- PKI integration (includes non-repudiation)
- Increase 2nd factor authentication to systems/challenge response authentication

Additionally, I grew closer to my solution based on multiple internal studies, vendor presentations and industry white papers. To minimize my impact to the organization I also consider the corporate culture and the reasons and decisions behind a non-smartcard infrastructure. I adopted a liberal philosophy towards industry best practices when accepting the risks associated with smartcards to facilitate the ecosystem that I was trying to foster. During the course of the SPRINT process the overall operational risk yielded consistent and expected industrial risks. Technological risk, public confidence, business continuity and vendor relationships were risks that are of particular concern.

Managing risk: As the coordinator I completed a series of preparatory steps, including the arrangement of several meetings with the appropriate smartcard vendors and business areas being impacted. Ideally, risk management will be a constant pursuit. I determined that the risks could be mitigated through keeping a lookout for shifting technological changes in the industry. Managing employee confidence in the smartcard effort may have been difficult if employees have difficulty using the card. I assumed that team members would look out for one another and will spend time with one another in first learning, teaching and finally training others about the smartcard solution presented to them. If a business area discovers a deficiency with the smartcard, we will have a multi-tiered support structure in place to address the concerns in a timely manner. Most importunately, as a large corporation we must show due diligence when selecting a vendor to reduce our risk. Typically, impartiality to a vendor must be demonstrated to avoid showing favoritism. In this instance I had dozens of smartcard vendors and it was overwhelming in deciding which company to go with. Therefore I identified the number one requirement for the project. I needed a vendor that could minimize our exposure to risk and "allocate team member resources, talents, skills, knowledge, and experiences that are fully identified, recognized, and used whenever appropriate (Bateman)." This was the key to a successful smartcard implementation. For this paper I will assume creative liberty when talking about the vendor that I decided to use. The vendor participated in the organizations vendor selection process, met the criteria and showed a willingness to support me in this venture.

In summary, risk has been assumed, vendors have been selected and following the critical path, decisions have been made to implement my solution as soon as possible. After forty days of effort, I think that I was prepared to present my findings to management and begin the smartcard implementation.

(During Snapshot) Defining the Stage:

With the key risks defined, I turned my attention towards understanding the necessary steps to must take in order to begin the smartcard implementation effort. I considered our corporate business goals, measured our risks and more clearly understood the political agendas working against the organizations smartcard effort. It was time to present these requirements and findings to management and deploy our smartcards.

Management was aware of the smartcard effort and prepared for the findings of my hard work. I presented the SPRINT results and was prepared to discuss the solution to correct the security issues that concerned them. The presentation was received perfectly, and generated a number of suggestions and ideas.

Management essentially suggested that I focus on integrating smartcard technologies into an environment that was as diverse as it was complex; they did not want to see smartcards used only for email. It was suggested that the smartcards store multiple credentials to enhance the *Public Confidence: (Employee Risk),* mentioned above. Management suggested that if I let employees store their personal credentials to web-sites on the smartcard, enterprise-wide acceptance of the smartcard would be more successful.

Management's position was simple; put information onto the smartcard that an employee would not want to share with others. I suggested that we place employee's passwords to their books of business onto the smartcard. Now if an employee shared their smartcard and password with another employee, that employee could see what the other one earns.

The scope of the effort changed from non-repudiation to an enterprise solution for key management, wide-spread portability and simplifying sign-on. To account for these scope changes, I needed to explore smartcard ready operating systems, applications, mid-tier components and hardware requirements to meet management's suggestions.

First I realized that the smartcard would need to be flexible simply because the Intel/Windows environments handle security a bit differently with each version. My organization required that I addressed the mixed mode/native mode environment. Most of my technical skills are UNIX based so I researched Microsoft's web-sit TechNet to provide an excellent description of what I was encountering. To view the definition on TechNet, hover over and left-click the native-mode hot-spot to see how Microsoft defines a mixed mode/native mode environment.

> "In Windows 2000 domains, the domain mode in which all domain controllers in a domain are running Windows 2000 and a domain administrator has switched the domain operation mode from mixed mode to native mode. Native mode supports universal groups and nesting of groups. In native mode, domain controllers running Windows NT 4.0 or earlier are not supported. In Windows Server 2003 domains, native mode is referred to as *Windows 2000 native* and it is one of three domain functional levels available." (TechNet)

Setting up the smartcard software required that the vendor and my small team understand how security works in the differing versions of the operating systems on both the end users pc, and the administrators machine at home. The vendor developed several run-time environments (RTE) to accommodate for the non native Intel environment. To install the smartcard package silently I developed several issuance packages that included the vendor's code. Several of these packages were created because of the vast differences between end users computers. The fat-client software package was approximately 40 megabytes and a silent install was critical to the smartcard effort. I could not afford for the organization to come to a grinding halt if my smartcard package required user intervention upon installation. Once the vendor's code was compiled, the Intel/Windows solution was non-interactive and based on a silent install script.

As part of the administration kit provided by the vendor, I replaced the current users Microsoft GINA with the vendor's GINA to facilitate the smartcard simplified sign-on. The new GINA needed to be modified to replicate the same "look and feel" so that users would not get confused. A second GINA for administrators was required because that was what the vendor developed. The GINA included a workaround if the smartcard was lost or forgotten. The GINA would allow for the user to hit the "Esc" key twice upon logon to bypass the

smartcard GINA. At this point I requested that the vendor consolidate the GINA's to minimize troubleshooting and support issues. Also, the GINA was configured so that when the user pulled out their smartcard, the computers screen would lock to the windows security "ctrl+alt+del" screen. By removing the smartcard the users SSH client session would also be terminated.

Next I examined the Hewlett Packard (HP-UX11.x) and AIX (5.2) UNIX operating systems. It was determined that it may be possible to extend the UNIX file system to use a smartcard though (SCFS) smartcard file system. OpenBSD-2.4 (Honeyman et al.) will allow for a user to mount the smartcard and allow for the smartcard file system to be accessed through UNIX. Although the UNIX operating system is a large part of my organization, policy prohibits us me exploring open source code as a production solution. I focused my efforts on using SSH2 keys within an emulation package or client and call the keys from the smartcard and pass them within the UNIX/SSH session. To address this first security hurdle, our vendor developed code that would allow for users SSH keys to be imported from the user's hard drive onto the smartcard; however, the application was difficult to navigate and use. The export utility was a single entity within the software package. The user was required to follow a series of instruction that I had to develop and publish. Unfortunately if the user skipped a step, the SSH private key would remain on the users hard drive and on the smartcard. To further complicate the issue, I was forced to decide on two SSH vendor's client's solutions XYZ Corporation and ABC Incorporated. Only ABC Incorporated would work with the SSH export software utility and our smartcard software. This issue is core to securing a trusted internal network. Communications in UNIX is often times sent in the clear from the user's desktop to the server. To passively enforce the use of a smartcard, a user would be required to plug in their smartcard, authenticate to the card and open a SSH client to a UNIX server. The smartcard would pass the encrypted SSH keys into the session and allow for secure internal communications.

Again I would like to emphasize that I maintain creative freedom and I will not mention the vendor that provided the smartcards or SSH solutions due to legal issues that may arise from what is written in this paper. I would like to say that the smartcard that I choose utilized current/cutting edge technologies. I decided on a USB token to minimize the hardware requirements that other cards required. It was important that the tool store encryption keys and not data files. A minimum of a full RSA 1,024-bit key encryption string, as well as 3DES 168-bit encryption and SHA-1 message digesting was defined by my team. Because our firewalls were Intel and UNIX based I needed a very flexible smartcard. UNIX firewalls in my organization required that the smartcard generate an on-board DSA-2048-key string. This was another problem that I needed the vendor to account for. The smartcard solution was not capable of generating a DSA key encryption string. Again, individuals that required this encryption could not use the smartcard. The vendor would be responsible for developing a solution to accommodate our needs. This is why it is important for a good vendor relationship. The vendor provided smartcard product specifications and support applications which are listed in figure2 below.

Supported Applications (Figure 2) (Aladdin)

Network Security Clients:	Windows 2000 Smartcard network logon, CheckPoint SecuRemote VPN client, RAS Dialup / RADIUS. Solution partner's NT logon, PC security and file encryption support.
PKI Solution partners:	Baltimore Technologies, Entrust, Microsoft Certificate Servers, GlobalSign, GTE CyberTrust, Thawte, VeriSign, DST, RSA Keon, Celo, Ashley Laurent, PC Guardian, ITEC, eSesix, Conware, Cryptomathic
eBusiness Security Clients:	Web Browser SSL v3 functionality: Microsoft Internet Explorer public key authentication. Netscape Navigator public key authentication and signing. WAC Authentication. Support for various Solution partners' communication & encryption applications.
Email Clients:	Microsoft Outlook/Outlook Express & Internet Explorer, Netscape Messenger

Product Specifications

Operating systems	win9X, Me, win 2000, XP and winNT
Certifications & standards	PKCS#11 v2.01, CAPI (Microsoft Crypto API), Siemens/Infinion APDU commands PC/SC, X.509v3 certificates, SSLv3, IPSec/IKE PRO
Models (by memory size)	16 k & 32k
On board security algorithms / processors	RSA 1024-bit, DES, 3DES (Triple DES) , SHA1, (MD5 - optional)
Smartcard chip security level	Smartcard chip security level ITSEC LE4 Smart card security certification (infinion).
Speed	RSA 1024 Bit signature approx. 1.0 sec RSA 1024 Bit key generation approx.25 sec
On board random number generation	Random number (RAND) is derived from an internal real hardware random number generator
Dimensions	47 x 16 x 8 mm (1.85 x 0.63 x 0.31 inches)
ISO specification support	Support for ISO 7816 1-4 specifications
Weight	5g
Power dissipation	120mW
Operating temperature	0 C to 70 C (32 F to 158 F)
Storage temperature	-40 C to 85 C (-40 F to 185 F)
Humidity rating	0-100% without condensation
Water resistance certification	IP X8 - IEC 529
Connector	USB type A (Universal Serial Bus)
Casing	Hard molded plastic, tamper evident
Memory data retention	At least 10 years
Memory cell rewrites	At least 100,000

After examining the different types of operating systems in the organization I turned my attention towards application security. I was determined to see if our applications could be compatible with smartcards. First I explored the 100's of applications to see if they could be secured through smartcard technology. I did this by scouring through internal documentation and testing applications one at a time in our development lab. Most application owners requested that their application call a smartcard and I found very little resistance to this form of authorization. I quickly discovered that most application credentials could be stored on the smartcard. Unfortunately most applications could not benefit from our PKI certificate exchange. Certificate exchange is the desired form of authentication with applications in a PKI. To perform this testing I

had to first we call upon our private corporate public key infrastructure. RSA Security the PKI experts, provide an excellent definition of what a PKI is.

”A public-key infrastructure (PKI) consists of protocols, services, and standards supporting applications of public-key cryptography. The term PKI, which is relatively recent, is defined variously in current literature. PKI sometimes refers simply to a trust hierarchy based on public-key certificates [1] (Webopedia), and in other contexts embraces encryption and digital signature services provided to end-user applications as well [OG99]. A middle view is that a PKI includes services and protocols for managing public keys, often through the use of Certification Authority (CA) and Registration Authority (RA) components, but not necessarily for performing cryptographic operations with the keys.”(RSA Security)

Next I focused on applications that were PKI compatible. Essentially I determined that with a partnership of technologies between Entrust Express, Entrust Secure Messaging Solution and Microsoft Outlook 2000 we could leverage the keys generated thru our PKI and store them on the smartcard for encrypting internal email communication. These applications were the only PKI applications that I could include in the smartcard roll out. Non-repudiation with email and enabling our users to store their digital certificates and digital signatures on the smartcards appeared to be possible. To enable for an easy transition into the PKI, the effort adopted the organizations communication channels that have been established for distributing the (reference/PIN’s) as discussed in chapter 6 of the SANS Security Essentials with CISSP CBK. (Cole et al. 281) Essentially employees will receive their authorization number through their email account and their manager will deliver their reference number in person. This security step was already in place. Now users could authenticate to the distributed directory in the morning and access their PKI applications until their smartcard was removed. When the user left their desk and removed their smartcard, access to the PKI applications was disabled. When the user needed to use the PKI, they had to authenticate to their smartcard through a password or PIN. Again users had to enter their PKI password to use the applications. This was an excellent form of controlling a user’s access to data.

It was apparent that very few applications can be a part of the PKI. Other corporate applications were not coded to receive a PKI certificate which was critical if they wanted to utilize true 2nd factor authentication with the smartcard. Essentially I focused only on Entrust Secure Messaging Solution and Entrust Express. The smartcard software that was provided by the vendor could either export Entrust credentials or generate them on the card with the on-board microprocessor. This was important to securing our ecosystem because now PKI certificates were being stored and transported on the smartcard itself, not on the user’s hard drive. Furthermore applications that use web based authentication can store their credentials on the smartcard, not in the cache store on the local hard drive. Management knew that this was the hook for getting business partners buy-in to the new technology. I nudged the implementation by demonstrating how a user’s credit card information could be stored on the

smartcard and then passed to the web site when the user needed to query their account after entering their PIN.

I had a number of versioned software packages that could address my specific concerns when securing the smartcard ecosystem in the enterprise. The key piece to this implementation is the Microsoft Active Directory snap-in that the vendor promised would allow for remote smartcard administration. The snap-in would allow for the administration of smartcards, on-line or off-line, so that changes could be made to the J2ME operating system that the smartcard used. J2ME is an optimized Java runtime environment that was developed as part of the Java 2 platform Enterprise Edition that is specific to addressing the commercial market which ranges from appliances such as refrigerators and toasters to smartcard microprocessors. The snap-in SDK (software developer kit) could also allow for the administrators to patch or update the J2ME operating system, allocate memory from one store to another and disable the smartcard if need be. The smartcard that we implemented had two separate memory stores. The general memory store was allocated to storing certificates, keys and passwords to web sites. The restricted memory store was devoted to the J2ME operating system. The business area that was deemed smartcard administrators would have this snap-in loaded on their workstation. Again this required further documentation on how to troubleshoot, resolve, manage and escalate technical issues.

At this point I have addressed the management suggestions, operating systems and applications that would be impacted by the smartcard deployment. As a final note I would like to talk about the hardware requirements. Figure 2 outlines the compatible operating systems, unfortunately I did not explore the USB 2.0 requirements of our docking stations or desktops that we had in the environment. In an organization my size, functions or support areas must take ownership of the hardware that they require. USB was not owned or supported by a function because our basic enterprise load did not require that the USB technology be supported. Because my smartcard used USB I needed to perform political gymnastics to assign ownership of the USB to my normal support channels. I determined who was the gatekeeper to this decision and explained why I needed to use USB. After several meetings and phone conversations plus management involvement, USB support was developed and implemented into the normal support flow in the enterprise.

Distributing the smartcards:

The smartcard vendor shipped the smartcard packages to me and I simply used inter-office mail to distribute the smartcards. The package contains a smartcard, USB extension cord, suction cup and a printed instruction set. I decided to deploy the smartcards to select individuals depending on the operating systems they used. I deployed several hundred smartcards to our NT administrators, W2K, XP and UNIX administrators. The distribution structure was in place and I thought, prepared for the smartcard roll-out. The floor support personal were not expecting 75 smartcard packages delivered to each floor everyday for two weeks. With the assistance of the designated smartcard support

specialist on each floor, the user was to plug in the USB smartcard and follow the printed instructions on how to set up their smartcard. If the owner for the smartcard was out of the office, the floor administrators were to place the cards into a secure cabinet. Apparently some floor administrators placed all of the cards into the cabinet and did not distribute any of the cards. This was discovered later in the process.

(After snapshot) The state of security:

This section of the paper will be devoted to lessons learned and drawing a crystal clear description on how effectively I enhanced the state of security for the enterprise with the smartcard roll-out.

Initially I was looking for a solution to secure the enterprise email against unauthorized use. The smartcard I used was chosen explicitly because the vendor relationship was well received by me and management. Unfortunately, the political environment shifted the scope of my effort and I was faced with a one stop shop smartcard solution. Now I was focused on providing support solutions and defining standards for reducing sign-on, key consolidation, application security and documenting all of my decisions.

The steps to explore a smartcard ecosystem in an enterprise that had never considered the technology were challenging. As a security analyst my breadth and experience only went so far. I did manage to create a smartcard ecosystem that worked. PKI keys, SSH keys and passwords were successfully exported to the smartcards. Users were now able to transport their credentials to perform their jobs off-site with a higher degree of non-repudiation. User's workstations were locking when the smartcard was removed from the computer. I raised the bar in enhancing the three bedrock principle of security (CIA) confidentiality, integrity and availability of data.

The pit falls to implementing a secure smartcard ecosystem were costly. I did not account for a number of hurdles that essentially stopped my implementation. The number one show stopper would be the fact that the vendor could not deliver an Active Directory snap-in for the administrators. The snap-in product was in alpha phase during the initial implementation. What I described in the previous section was hypothetical with regards to the Active Directory snap-in functionality. To format the smartcards I used an unsupported beta version of a modified Active Directory snap-in that was used in testing. Smartcard administrators were required to go from desk to desk throughout the country to support technical issues with the user's smartcard. We could use a remote assistance application, however if the user's desktop computer was an older computer with a slow CPU, troubleshooting often times came to a grinding halt. We did not have a populated smartcard database to provide users names, employee numbers and smartcard software versions that the employee received during the initial roll-out. If a user was released from the enterprise their smartcard could not be disabled or modified because we had no idea of who they were. I also lost track of who had a smartcard, and what version of software they

were running. The vendor insisted that their snap-in would allow for administrators to see when a user's smartcard was plugged into their computer so that silent changes to be pushed to the card without the user knowing. Again at this time, no snap-in was available through my vendor.

The software that was required for the smartcard was a fat-client application. I did not account for the amount of configuration required in the application for each individual's needs or job function. Some user's needed SSH keys, others did not. Some users would reach their 20 kilobyte memory limit because they were storing AOL passwords or Monster logins. If a user had their PKI credentials recovered a number of times they would also reach their memory limit. I did not account for smartcard memory store allocation changes which were configurable through the non-existing snap-in. Some employees never received their smartcards because the floor support personal did not distribute them.

I failed to consider a number of smartcard solutions and vendors to provide a secure smartcard ecosystem. It was not clear to me that an enterprise our size would require a number of smartcard solutions to meet the variety of applications that I attempted to accommodate for. For example the smartcard I implemented could be re-engineered to store fingerprint data files. Unfortunately the changes to the software would take the vendor months to develop. If I had another smartcard solution that could store fingerprint data files all I would need to do is distribute that smartcard. By deciding on one solution I limited the security that the enterprise could call on. Now that I look back, I can not find one product whether it is the hand-soap in our restrooms or the monitors on our desks that are exactly the same in this enterprise. One vendor was not going to meet the security requirements that I required.

Conclusion: Lessons learned

With the lessons learned I still enhanced the state of security in the enterprise. I could not account for users sharing their smartcard and password/PIN's with others which was a significant vulnerability. I did mitigate the risk by allowing users to store personal information on their smartcard in the hope that users would not want to share that information. The use of non-repudiation was applauded by both the legal department and auditing. SSH2 keys, PKI certificates and passwords were removed from the user's hard drive in some cases. Confidentiality of data was more secure with the use of 2nd factor authentication which was provided by the smartcard. Users were more

comfortable with using a smartcard that required something they know (password) and something they have (smartcard) to store their data. Essentially protecting personal privacy and proprietary information was achieved with the use of a smartcard. I facilitated the availability of data to the enterprise by consolidating user's credentials to a local store. Now users could be in a remote location and still ensure that their credentials were available in a timely and reliable method. The smartcard enhanced improper information modification by the user and other people which ensured that the integrity of the certificates or passwords were authentic.

Currently the enterprise engineering department is devoting a significant portion of its resources to continue where I left off. Now there are multiple teams devoted to multi-factor authentication, implementation and support. I believe that my grass roots effort and hard work did identify the lowest common denominator in introducing a decade old technology to a bleeding edge company.

References and additional reading

"Aladdin Securing the Global Village." (2003)
URL : http://www.ealaddin.com/etoken/PRO/default.asp?cf=tl (11 Sept. 2003)

Bateman, Arnold. "Team Building: Developing a productive Team." (May 1997)
URL: http://www.ianr.unl.edu/pubs/misc/cc352.htm (8 July, 2003)

Briney, Andy. "A Smart Card for Everyone." 01 March 2002.
URL: http://infosecuritymag.techtarget.com/2002/mar/cover.shtml (5 July, 2003).

Cole, Eric., Fossen, Jason,. Pomeranz, Hal,. Northcutt, Stepher. “SANS Security Essential with CISSP CBK.” The SANS Institute: 04 (2003): 347.

“Cryptographic Token Support.” (27 march, 2003)
URL: http://publib7b.boulder.ibm.com/wasinfo1/en/info/aes/ae/rsec_cryptts.html (9 Sept, 2003)

“DNP Products and Services.” (1999)
URL: http://www.dnp.co.jp/international/card/ac.html (11July. 2003)

Federal Information Security Management Act of 2002 (Title III of E-Gov) H.R. 2458-48, SEC. 301. Information Security.
URL: http://csrc.nist.gov/policies/FISMA-final.pdf (June 10th. 2003)

“Financial Privacy: The Gramm-Leach Bliley Act.”
URL: http://www.ftc.gov/privacy/glbact (20 Aug. 2003).

“Frontmatter: Open Group Guide.” (1998)
URL: http://www.opengroup.org/onlinepubs/009219899/front.htm (Aug 18. 2003)

Honeyman, Peter., Itori, Naomaru., Rees, Jim. “SCFS: A UNIX File system for Smartcards.” (10 May. 1999) URL:
http://www.usenix.org/publications/library/proceedings/smartcard99/full_papers/itoiSCFS/itoiSCFS.pdf (9 July. 2003)

“Information Security Forum.” (2003)
URL: http://www.securityforum.org/html/frameset.htm (6 Oct. 2003)

ISO 7816-1 “Smart Card Standard: Physical Characteristics of Integrated Circuit Cards” (9 Sept. 2003)
URL: http://www.cardwerk.com/smartcards/smartcard_standard_ISO7816-1.aspx (21 Aug. 2003)

“Microsoft TechNet.” (2003)
URL:
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/deployguide/dnsbm_wir_vrro.asp (June. 2003)

“RSA Security: The Open Group, *Architecture for Public-Key Infrastructure* (APKI).” 1999.
URL: http://www.rsasecurity.com/rsalabs/faq/4-1-3-1.html (18 Aug. 2003)

“Webopedia: PKI.” (2003)
URL: http://webopedia.internet.com/TERM/P/PKI.html (18 Aug. 2003)