

Title: Software Encryption Initiative
Presenters: Albert Kondi and Russell Davis
Track: 9
Day: Tuesday
Keywords: DES, Encryption, COMSEC
Abstract: Software encryption provides a cost-effective alternative to hardware encryption devices. This paper describes the analysis used in identifying software encryption, in a limited resource, risk taking environment, as viable for unclassified but sensitive communications security. The approach provides a glide path to using a Fortezza PC Card system.

Software Encryption Initiative

1.0 Preface

This paper represents the views of the authors and not necessarily the views of their respective employers. Some of this paper was done under contract number DAHC94-91-C-0002. In his 1997 budget, President Clinton called for a 3.4% decrease in Defense spending [1]. At the same time, the Director, National Security Agency (NSA) warned of a fundamental new danger from cyber attacks [2]. This paper addresses a method for providing network security in these times of limited resources. This paper describes a software encryption approach that provides a glide path into a Fortezza based solution.

Under US Army policy, the Reserve Component Automation System (RCAS) is required to incorporate Data Encryption Standard (DES) for the backbone networks. Encryption is the most cost-effective mechanism for ensuring confidentiality and integrity of information transported across wide area networks. Within the Defense community, the US Navy Support Automation Logistic Telecommunications Services (SALTS) program has purchased 3,000 copies of a software encryption product (Secret Agent). Additionally, the Defense Investigation Agency has purchased 100,000 copies of the same product. These Defense software encryption initiatives illustrate that software encryption is not a new idea.

The RCAS mission is to provide the Army Reserve Component (RC) with an integrated information delivery system with data and application systems that will support day-to-day operations and mobilization functions for the Army National Guard (ARNG) and the U.S. Army Reserve (USAR).

2.0 Department of Defense (DoD) Environment

Within the DoD community there exists a myriad of heterogeneous encryption systems. These are predominately based on hardware devices. For the classified environment, NSA type 1 approved devices are used for encryption. For strictly unclassified information, either the Data Encryption Standard (DES) or the NSA type 2 devices (including the Fortezza card) are used for confidentiality protection. If the unclassified information does not fall under the Warner Amendment, such as weapon systems, then SKIPJACK is the only type 2 algorithm approved. These devices typically afford confidentiality protection of information in transit over wide area networks (WAN). However, hardware is not always a panacea for communications security (COMSEC). Take for example the two US rotary wing aircraft in Northern Iraq that fell victim to friendly fire. US military aircraft normally include an identification friend or foe (IFF) capability that is centered on a NSA type 1 encryption device. The fact that such a system could fail (for whatever the reason) illustrates that fact that all encryption systems exhibit some degree of risk. In a risk taking, environment with scarce resources, the challenge for any encryption system is to achieve acceptable risk.

2.1 The Data Encryption Standard

When Federal Standard 1027 was still applicable, hardware encryption devices were required for DES implementations. The DES has been the algorithm of choice for unclassified sensitive data, especially for data confidentiality protection of network traffic. The Banking community currently uses DES Message Authentication Codes (MAC) for ensuring the integrity of wire transfers. On any given day, billions of dollars worth of wire transfers are protected with MAC authentication.

The current DES standard [3] allows the algorithm to be implemented in software. The DES algorithm uses a 56 bit key resulting in slightly more than 70 quadrillion, or 70 thousand million million (2^{56}) possible keys. While this is an impressive number, commercial off the shelf (COTS) workstations are increasing in performance at an exponential rate. For example, the desktop machines today are approximately 100 times more powerful than those of 10 years ago. Typical machines today operate in excess of 100,000,000 instructions per second. Every five years the National Institute of Standards and Technology (NIST) reaffirms its encryption algorithms. With respect to DES, this decision process will take place next year. It is possible that the NIST will introduce a new Government algorithm, such as the Advanced Encryption Standard (AES) discussed in the next section, thereby requiring a costly hardware upgrade of the current DES devices. For unclassified non-Warner amendment information, the NIST is responsible for implementing mandatory standards for all Government agencies, including the DoD.

2.2 Advanced Encryption Standard

On 2 January 1997, the NIST announced in the Federal Register the development of a Federal Information Processing Standard (FIPS) for Advanced Encryption Standard (AES). There are several requirements including that the algorithm be implementable in hardware or software. For sensitive but unclassified systems, the AES could eventually be the standard algorithm used. Moreover, it is likely that the AES will not interoperate with the DES, thereby making the existing investment questionable. The AES could be the standard algorithm that replaces the DES within the RCAS.

2.3 The Defense Message System (DMS)

Recently, the NSA has championed a new Personal Computer Memory Card International Association (PCMCIA) compliant encryption device, called the Fortezza PC Card. These devices are inserted into a PCMCIA reader, located on the computer workstation and were originally designed for use in the DMS. When used with a trusted operating system, these provide excellent security. However, the Fortezza is also being used on untrusted operating system environments including DOS and Windows for Workgroups machines. The risk in using untrusted operating systems has long been known. For example, The US Air Force's Electronic Systems Division documented the following in 1973 [4]: *If an error in an operating system program allows a penetration program to work, that program will work every time it is executed – typically retrieving without detection any information accessible to the computer.* To this end, there are a number of hacker tools specifically designed to penetrate DOS environments.

Another example of hardware under the operating system's control is evident from the National Computer Security Center (NCSC) evaluation of Sentinel [5]: *The security mechanisms can be maintained only if both the operating system in which Sentinel runs, and Sentinel's operational files are protected from unauthorized modification. Since Sentinel's protection mechanisms are implemented in single-state machine hardware, it becomes essential that user/system separation be maintained. In systems with this type of architecture, non-privileged users operate in the same memory space as the security-related system functions; hence it would be possible for an experienced user to modify the operating system and circumvent the security mechanisms without the likelihood of detection.*

The Fortezza card is a hardware crypto implementation running on software. Consequently, Fortezza is only as good as the software it runs on. This argument becomes important when examining software encryption alternatives. If you must trust the software to properly operate the hardware device, why then would it not be trusted to perform encryption for unclassified but sensitive applications?

The DMS Fortezza cards implement the Digital Signature Algorithm (DSA), the Secure Hash Algorithm (SHA), the SKIPJACK encryption algorithm [6], and a keys exchange function. These devices are also being used in conjunction with Netscape Web servers to provide an encrypted session. The session uses the Secure Sockets Layer (SSL), a protocol that provides key exchange and packet encryption. The headers are left in the clear with the packet contents encrypted. It is anticipated that the Fortezza technology will interoperate with the proposed software solution discussed in this paper.

When comparing the Fortezza card to a software solution, the risk associated with software must be examined. In the case of the Fortezza card, it requires properly installed software device drivers before the device will operate. In the case of software encryption, all processing is done in software under operating system control. Additionally, a software version of the SKIPJACK algorithm has been used by the NSA [7]. If software cannot be trusted, then neither the Fortezza nor the software encryption options are viable. If the software can be trusted, then either solution is viable for providing encryption.

One notable feature missing from the Fortezza PC card (and Type 1 encryption devices such as the Secure Data Device (SDD)) is compression. Once information is encrypted, compression is no longer possible. Interestingly enough, many compression algorithms add to the overall length when attempting to compress encrypted text. For proper compression, plain text information is required. Compressing information prior to encryption is the only viable alternative and is not a new idea. For example, the Selective Encryption Terminal [8] performed compression before applying the DES software encryption.

This would result in the costly replacement of the Fortezza investment. In either the case of DES or SKIPJACK, the NIST would hopefully grandfather existing implementations. However, as new users enter the network, they might be forced to use a different encryption algorithm, thereby making communication with existing Fortezza (or DES) devices impossible.

Another risk to Fortezza is illustrated by National Security Decision Directive (NSDD) 145. In 1984 President Reagan signed NSDD-145. This directive gave the NSA responsibility for securing the nation's communications. With NSDD-145, the NSA embarked on the Commercial Communications Security Endorsement Program (CCEP). The NSA developed a number of cryptographic algorithms for use in classified (type 1 devices) and unclassified (type 2 devices). In the case of type 2 algorithms, the algorithm is classified and housed in a tamper resistant package. The algorithm used in the Fortezza PC card, SKIPJACK, is a type 2 algorithm.

The NSDD-145 was assailed from sources in and outside of Government. Consequently, the Congress passed and the President signed the Computer

Security Act of 87 (Public Law 100-235). In its responsibility for unclassified security was transferred from the NSA to the NIST. The reason we can use the Fortezza PC card is because it complies with the NIST Escrowed Encryption Standard (EES). The EES has met with considerable resistance from professional organizations such as the Association for Computing Machinery (ACM) and the American Civil Liberties Union. For example, Walker [9] points out: *It appears unlikely that government key escrow will relieve the tension between the government's and public's interests in encryption.* There have been calls to have the EES rescinded either by Executive Order or Public Law. For example, Senators Conrad Burns (R-MT), Leahy (D-VT), and Wyden (D-OR) bill, S.1726, is critical of escrowed encryption (the House version is HR 3011). The Congress could do to Fortezza what was done to NSDD-145, thereby making the EES and Fortezza a thing of the past.

The Fortezza PC card is part of the multilevel information system security initiative (MISSI). At least with respect to the US Army, MISSI falls short of Army needs [10]. Many of the legacy systems do not have PCMCIA compliant readers that are required to use Fortezza. Many of the legacy systems either cannot support encryption or there is a lack of funds to buy the encryptors. For these systems, encryption is normally waived. Software encryption provides an attractive alternative to the normal practice of waiving the requirement for encryption.

Moreover, with the advent of the AES, one must ask why the 80-bit key associated with the Fortezza PC card would be used. The AES will most likely be 112 bit or larger in key length.

One of the original design features of the Fortezza card was the ability to recover keys. Under normal circumstances, the key recovery would not be available to the users, but only law enforcement. Currently, the escrow agents are the NIST and Department of Treasury (DOT). It was envisioned that the court could authorize a "wire tap" and both the NIST & DOT would provide their key components to law enforcement. Once combined, law enforcement would have the ability to monitor all traffic. Recently, the DoD has announced its intention to remove the escrow component from the Fortezza cards.

Recently, the Fortezza has been approved for encryption of up to Secret information. This configuration is predicated on a number of other steps, but it does illustrate the confidence given the device. It is unusual that a NSA type 2 encryption device operating with an 80-bit key would be used for encrypting classified information.

2.4 Hardware Encryption

Hardware obtains its strength from being rigid and hard to change. The RCAS uses hardware DES encryption devices for encrypting wide area traffic. However, should the DES be replaced with another algorithm, such as the AES, then to maintain interoperability, all existing hardware would have to be replaced with devices using the new algorithm.

Hardware encryption is typically a separate device that contain a cryptographic engine, possibly consisting of a microprocessor. However, a separate device does not imply a secure solution. Although DES devices made by the same company work with their product line, many do not properly communicate with other products. Given the same key, mode of operation, and initialization vector, all DES products should work across vendor products. Unfortunately, when one purchases a DES hardware device, they are stuck with the single vendor product line from then on. If a flawed hardware implementation of DES is discovered, the correction costs are excessively high when compared to software.

2.5 Software Alternatives

Some have argued that for system security, COMSEC and computer security (COMPUSEC) must be combined [11]. By performing the COMSEC functions in software, for unclassified but sensitive communications, the system security goal is satisfied. If software encryption is utilized, it is important to have assurances that the software environment can be trusted. As Thompson [12] points out, you can't trust untrusted software.

The next RCAS release includes Microsoft Exchange. This product will replace the current Microsoft Mail solution. Additionally, starting with Exchange version 5.0, X.500 directory functionality is included. Any client application that is lightweight directory access protocol (LDAP) compliant will be able to access the directory. The directory is a logical storage location for X.509 version 3 certificates.

The NIST standard for crypto modules [13] should be used by the DoD when selecting encryption products for unclassified information. This mandatory standard now allows encryption to be performed in software.

Crypto modules that implement the DES must comply with FIPS 140-1 Security Levels. In the case of the RCAS, the C2 NT Operating System could be used to achieve a Level 2 Security Level as defined in FIPS 140-1. A crypto module at this level is equivalent to a Level 2 crypto module implemented within hardware.

| |
|---|
| “Level 2 also allows software cryptography in multi-user timeshared systems when used in conjunction with C2 or equivalent trusted operating system.” |
|---|

When compared with hardware, software is easy to upgrade and is usually cost effective. With the integrity controls found within the NT operating system, software nearly approaches the integrity expected with hardware. Moreover, if the operating system can not be trusted, then other serious problems exist which cannot be solved by encryption alone.

2.6 Public Key Infrastructure (PKI)

One challenge associated with the software approach is the need to distribute X.509 version 3 certificates. Work has been done to address how certificates are generated, stored, and retrieved. There are vendor products that perform the Certification Authority (CA) functionality. That is, to receive requests from an Organizational Registration Authority (ORA) and to generate a certificate based on that request. Once generated, certificates must be placed in a repository. Lightweight Directory Access Protocol (LDAP) compliant clients can then retrieve certificates as they are needed. The RCAS is concerned primarily with communications security (COMSEC). The issues associated with document storage do not apply to this environment.

The DMS will have a directory service. To implement a DMS environment, there must be one certification authority workstation (CAW) for every 500 to 1,000 users. Each CAW requires two full time people. The CAWs are used to update the Fortezza PC cards. Within Fortezza, there is a certificate with a single year validity period. Additionally, the compromised key list (CKL) and the certificate revocation list (CRL) are updated every 28 days. In the case of the certificate, it will be loaded onto the Fortezza card.

On 31 December 1996, Assistant Secretary of Defense, Emmett Paige, Jr. Released a memorandum describing signature implementation for the Defense Travel System (DTS). The memorandum addresses public key infrastructure devices, including software. The following is from that memorandum: *Applications that support software based digital signature keys must be interoperable with other Public Key Infrastructure (PKI) devices (hardware and software), such as the Components being fielded by the Defense Messaging System.*

2.7 DoD Standards

Both the Joint Technical Architecture (JTA) and the Army Technical Architecture (ATA) describe Internet Protocol (IP) version 6 (Ipv6) and the Encapsulating Security Payload (ESP) under emerging standards. The ESP is included in the IP Security (IPSec), which is mandatory for Ipv6 implementations. IPSec provides an encryption implementation for IP traffic. It works in the current Internet environment and that specified in the next version of the Internet (Ipv6).

Given that IPV6 will predominantly be a commercial off the shelf (COTS) software product, many within the DoD community will have a software encryption capability at their disposal.

The U.S. Army [14] requires approved cryptographic equipment used for protecting certain unclassified information, such as that within the RCAS. However, Army Regulation also states “if protection is not waived, techniques approved by NSA or the National Institute of Standards and Technology (NIST) will be used to protect the data.”

3.0 The RCAS Environment

Figure 1 illustrates the RCAS large site environment. Note there are hardware encryptors located at the Frame Relay and dial-up locations. These devices operated only with the same vendor products. That is, the system users are dependent on the vendor in that there will be no interoperability with different vendor products. In the absence of some hardware standard, launching a hardware solution for encryption leads to isolating the RCAS.

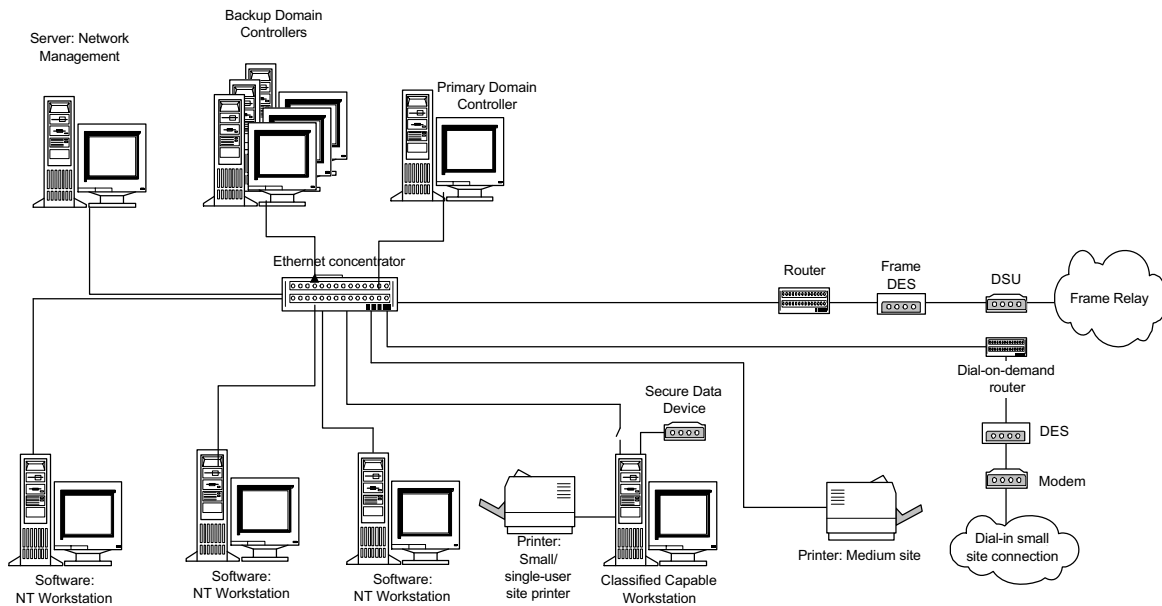


Figure 1 RCAS Large Site

Originally, the RCAS design included DES hardware encryption for all wide area network (WAN) communications. Figure 2 illustrates two local area network (LAN) segments securely communicating over a wide WAN. The traffic on either side of the gateway was not encrypted.

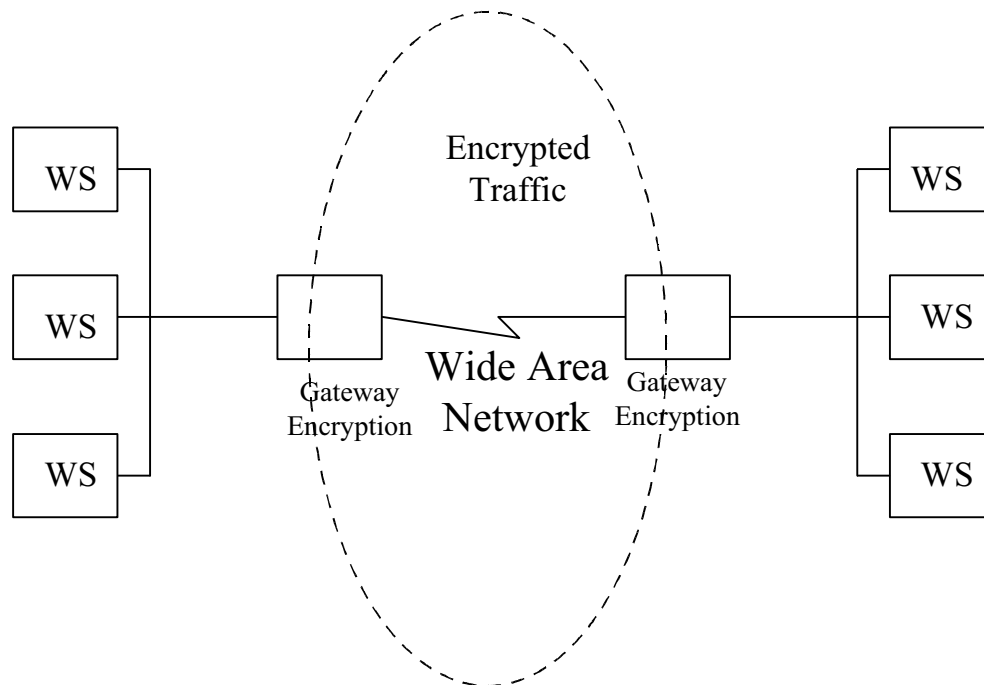


Figure 2 Original RCAS Wide Area Network Encrypted

In January 1997, the RCAS was given the approval to defer further deployment of the hardware encryption devices. There were several reasons for this decision. A major consideration was the cost associated with hardware encryption. The RCAS hardware encryption devices cost in excess of \$10 million, which is considerably higher than an equivalent software approach. Another reason for the encryption deferral was the availability of software encryption products that could be run on the workstation. Using the software approach provides protection on the LAN as well as the WAN communications.

Within the RCAS, workstations and servers use the NT operating system. The Windows NT operating system was evaluated by the NCSC. The NCSC determined that Windows NT met the C2 level of trust. The final evaluation report dated 1 March 1995 indicated that Windows meet the requirements for a B2 trusted path. What that means is the user is authenticated to the operating system. This is critical when using tokens such as Fortezza. It also has meaning when considering software encryption.

Starting with NT version 4.0, the operating system includes a Cryptographic Application Programming Interface (API) (CAPI). Microsoft will not initially include the cryptographic module(s) required for Government use and they expect third party vendors to fill this gap. Encryption (and decryption) would be performed at the operating system level. Microsoft is providing network encryption but not IPsec. It is necessary to purchase third party software products that perform the requisite encryption. A future release of Microsoft products may include a complete encryption solution.

It is worth noting that the RCAS is concerned primarily with communications encryption. That is, there is no effort to provide file encryption capability on the workstations. Thus, issues such key recovery, after a period of time, are not drivers within the RCAS. The digital signature capability is used for session authentication as opposed to document authentication.

3.1 Windows NT

The RCAS is predominately a Microsoft Windows NT workstations and servers environment. Periodically, the RCAS includes newer versions of software and more powerful hardware. In this environment, future versions of Microsoft products will be deployed. The Windows NT products include features used to ingest X.509 version 3 certificates and to cache them locally. The certificate cache reduces the telecommunications required when compared with implementations requiring a certificate be retrieved each time it is needed. The certificate server provides an automated method for issuing certificates. In contrast with the MISSI approach, CAW requires users present their Fortezza card for a certificate download.

3.2 Microsoft Exchange Server

Within the RCAS, Microsoft Exchange is used as the mail solution. The latest Microsoft Exchange Server provides an X.500 directory capability. It supports any LDAP client. It provides SSL for the Network News Transport Protocol (NNTP), mail, Web, and LDAP. The SSL protocol is an application to application protocol that includes an option for compression (version 3 of the SSL protocol). Interestingly enough, at the time of this writing, the LDAP protocol is specified in the draft NIST Minimum Interoperability Specification for PKI Components (MISPC). It would appear that the NIST welcomes the commercial solutions from their future standards.

The current version of the Microsoft Exchange Server does not include the DMS capability. This must be purchased separately off the DMS contract. However, it is expected that Microsoft Exchange Server version 6.0, when used with the Fortezza CSP will provide the requisite functionality. At that time, the Directory Access Protocol (DAP) will also be available to the RCAS community. By using the Fortezza CSP, the Fortezza cards can be used for a multitude of non-DMS uses.

When the RCAS selected Microsoft Exchange, the DMS functionality was advertised in the NSA's MISSI products document. It was later learned that there would be an additional cost for DMS capable products. This illustrates one of the challenges associated with integrating MISSI products. There were also a number of challenges with PCMCIA readers that would not work with the

Fortezza PC cards. At the time of this writing, we continue to be surprised with the costs associated with MISSI. We perhaps will not know the full cost until it has been paid for.

3.3 Microsoft Certificate Server

Microsoft currently has a certificate server currently available for beta testing. When combined with a Digital Signature Algorithm (DSA) CSP, it generates X.509 version certificates, the same generated by the Certificate Authority Workstation (CAW) found in the DMS environment. Two full time persons are required to man the CAW. Moreover, every 500-1,000 users require a CAW. The users must have their Fortezza cards physically loaded by the CAW. In contrast, the Microsoft Certificate Server can support a much larger community. The Microsoft Certificate Server deals with electronic requests in a client-server mode of operation. We expect one Certificate Server could support the ARNG and another the USAR.

The DSA is specified in FIPS Publication 186. This is the same standard used in the Fortezza card. Thus, the signature components can verify signatures generated in software or generated by a Fortezza card. Using the DSA provides the public key infrastructure interoperability previously described in section 2.6.

4.0 Analysis

The RCAS workstations use the NT operating system. The NT operating system includes complete C2 security functionality. Cryptographic algorithms implemented in a C2 operating system can achieve a Level 2 rating as defined within FIPS Pub 140-1. This level is comparable to many of the available hardware products. In fact, many of the existing hardware products have not been tested in the NIST approved voluntary labs.

Software can be upgraded for a fraction of the cost associated with hardware. Moreover, software does not require additional space nor power. The flexibility associated with software should be considered when selecting an encryption approach for unclassified communications.

Figure 3 illustrates how software encryption would actually enhance the security of the RCAS. In this environment, encryption occurs at each workstation. Consequently, all sensitive LAN traffic is encrypted. With untrusted workstations sharing the LAN, software encryption protects sensitive information from sniffer attacks.

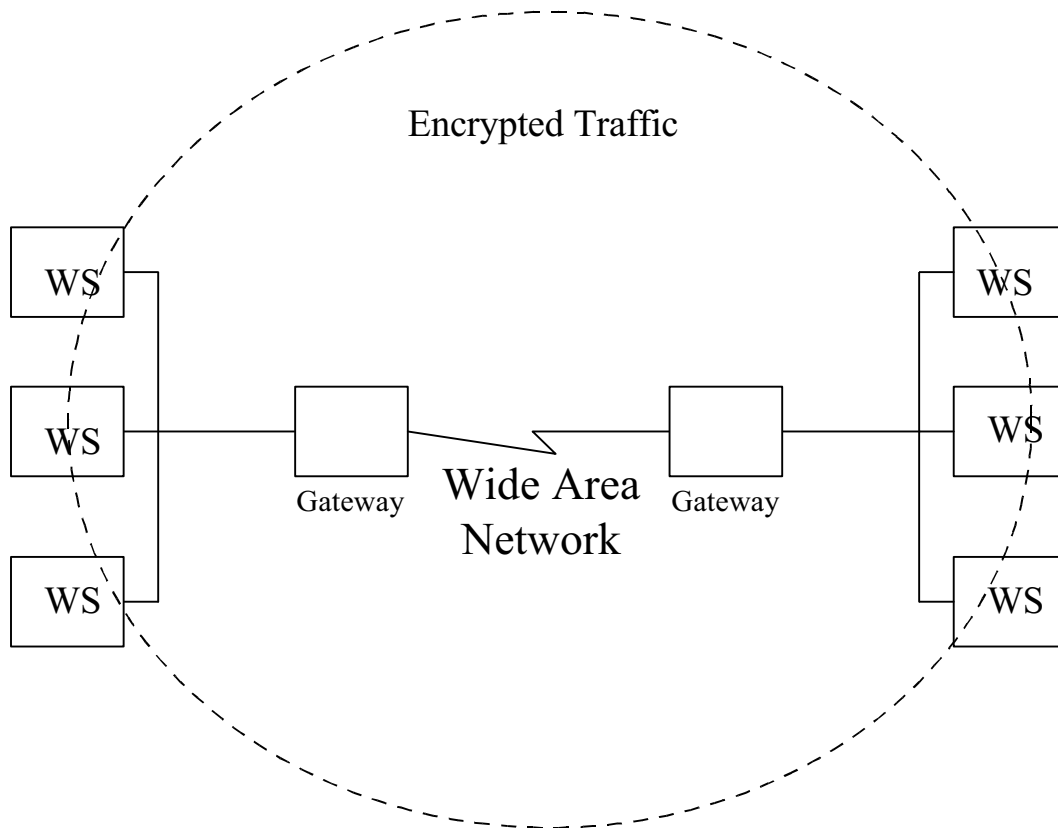


Figure 3 Software Approach

Starting with Microsoft NT version 4.0, the operating system includes a Cryptographic API (CAPI). With the Microsoft Service Pack 2 for Windows NT 4.0, the CAPI includes DES and triple DES encryption algorithms. It is possible that the NIST will approved triple DES as the AES. If this is the case, then there will be no upgrade issues within the RCAS. If on the other hand, a new algorithm is needed, then additional software would be required.

4.1 Legacy Systems

Within the Army and perhaps the Department of Defense (DoD), there exist a number of legacy applications that currently do not use encryption. Additionally, the information traversing untrusted networks includes Privacy Act information and other sensitive information that requires confidentiality protection. Many of these applications include some distributed access. If a hardware encryption solution were considered, all sites would typically have to be upgraded simultaneously with the new encryption devices. In such environments, hardware options are limited. In contrast, IPSec solutions negotiate a session key during the start of the session. Unencrypted sessions can be allowed until such time that all sites have had their software upgraded.

4.2 Software Security

Perhaps the biggest question the reader should have is how secure is the software encryption solution as compared to a hardware alternative. When compared to the original RCAS design, the software encryption approach encrypts traffic at the LAN level. Additionally, the encryption keys are used only for one session. The hardware devices used a significantly longer crypto period (how long the key is used). When compared with the Fortezza card, the analysis is quite different. When used with a trusted operating system, the Fortezza card provides excellent protection. However, when using an untrusted operating system, such as the Disk Operating System (DOS), Fortezza card security is questionable. For example, the Fortezza card includes a privileged command that exports the x (private) key. A disgruntled employee, or hacker, or prankster could set up a Trojan horse program on the untrusted system. Should a privileged user access his or her Fortezza on that machine, the x key could be compromised. It is sufficiently important to use the correct operating system when COMSEC is controlled by software.

4.3 Cost Analysis

The DMS uses certificates have a one year expiration date. That is, once every year, the Fortezza cards must be loaded by their respective CAW. The loading process takes approximately 12 minutes using the current generation of CAWS. The RCAS consists of approximately 56,000 workstations. However, the ARNG and USAR will have a number of additional machines procured from other than RCAS sources. The potential number of users within the ARNG is 300,000 and there are 200,000 potential users within the USAR. The ARNG and UASR respectively provided the numbers used in this analysis. Using 1,000 users per CAW, the Reserve Component (ARNG & USAR) would require approximately 500 CAWs, requiring 2 full time people per CAW. Using a 12 minute load time, which excludes the time required to get the Fortezza card to and from the CAWs, the annual Fortezza load time is approximately 100,000 hours or 50 man years (using 2,000 hours per man year). Thus, using the Fortezza/CAW solution for the RC, not counting the personnel turn over annually would result in at least a 1,050 man years of effort. Additionally, there is a cost associated with the Fortezza card. These devices cost approximately \$69 per device. Its is also worth noting that the DMS was designed specifically for messaging. That is, to protect all message traffic, as the software solution does, requires additional software, such as IPsec or SSL. Thus, you need the software that has the encryption capability therein, in order to have a Fortezza based solution. With

the DoD personnel cuts and limited budgets, software encryption provides an attractive alternative.

Microsoft has provided a streamline architecture necessary for a public key based key management scheme. Within the RCAS, most of the infrastructure was selected independent of the software encryption needed.

5.0 Alternative Course of Action

Affordability is our issue. With scarce resources, we recommend that an IPsec based encryption solution be used to provide COMSEC. There should be a certificate server and an X.500 repository used for key management. The workstation clients should be compliant with LDAP and directly access the X.500 directory for certificates as needed. The workstations should have a certificate cache capability and operate on X.509 version 3 certificates. The key exchange should be Diffie-Hellman and use the Digital Signature Algorithm.

6.0 Summary

In these times of limited budgets, software offers an attractive alternative to the costly hardware approaches. RCAS management is still trying to convince the appropriate people that software encryption offers an alternative to hardware encryption. To date, the RCAS has received approval to defer further fielding of hardware encryption devices pending a decision by the Department of Army (DA) on software encryption or the selection of standard hardware. There have been a number of Pentagon briefings and we expect to have approval to field a software encryption solution.

7.0 References

- [1] Fulghum, David A., *Pentagon Budget Suffers New Cuts*, Aviation Week and Space Technology, volume 146, number 6, pages 24-25, February 10, 1997.
- [2] Covault, Craig, *Cyber Threat Challenges Intelligence Capability*, Aviation Week and Space Technology, volume 146, number 6, pages 20-21, February 10, 1997.
- [3] FIPS Publication 46-2, Data Encryption Standard, NIST, December, 1993.
- [4] Electronics Systems Division, Air Force Systems Command, Computer Security Developments Summary, report MCI-74-1, December, 1973.
- [5] National Computer Security Center, Final Evaluation Report of Computer Security Corporation Sentinel, CSC-EPL-87/004, 1987.
- [6] FIPS Publication 185, Escrowed Encryption Standard, NIST.
- [7] Hofman, Lance J., *Building in Big Brother*, pages 119-130, copyright 1995 by Springer-Verlag.

- [8] *The Selective Encryption Terminal: A New Approach to Privacy Protection*, The Mitre Corporation, METREK Division, Report M76-56, September, 1976.
- [9] Walker, Steven T., et al, "Commercial Key Recovery," *Communications of the ACM*, volume 39, number 3, pages 41-47, March, 1996.
- [10] "Army Information Operations Protection Command and Control," *Signal*, Volume 50, number 11, pages 47-50, July, 1996.
- [11] "COMSEC Integration Alternatives," *Proceedings of the 11th National Computer Security Conference*, pages 122-125, October, 1988.
- [12] Thompson, Ken, "Reflections on Trusting Trust," *Communications of the ACM*, volume 27, number 8, pages 761-763, August 1984.
- [13] FIPS Publication 140-1, *Security Requirements for Cryptographic Modules*, NIST, January, 1994.
- [14] Army Regulation 380-19, *Information Systems Security*, Department of Army, August, 1990.