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

3DES and Secure PIN-based Electronic Transaction Processing

Michael Buegler

July 25, 2004
GIAC Security Essentials Certification (GSEC)
Practical Assignment, Version 1.4b
Option 1

Abstract

One of the most important computing security issues today involves the secure processing of electronic financial transactions. This paper provides a technical look at how our company uses the triple-Data Encryption Standard (3DES) in conjunction with Hewlett Packard's Atalla Network Security Processor (NSP) and Atalla Key Block (AKB). This combination ensures the secure processing of transactions initiated with our Personal Identification Number (PIN) entry devices or with our stored value card.

The following discussion will provide an overview of the PIN-entry based electronic transaction processing services we provide. It will start with a discussion of the parties involved and describe what occurs when our stored value card is used for a purchase or when another issuer's card is used with one of our PIN-entry devices. I will discuss some of the requirements for participation in the STAR debit network and describe STAR's role in the processing of our electronic transactions. Vulnerabilities in the Data Encryption Standard (DES) and 3DES will also be discussed. In addition, the importance of DES key management techniques such as Derived Unique Key per Transaction (DUKPT) and dual control and split knowledge will be reviewed. This paper will also describe how the Atalla NSP protects PIN blocks and how the Atalla Key Block protects the integrity of 3DES keys stored in hostile environments.

Electronic Transaction Processing Overview

There are several entities involved in electronic financial transactions: the card issuer, the cardholder, the card acceptor, and the acquirer.

- The *card issuer* issues the identification card to the *cardholder* and guarantees payment to the acquirer as long as the cardholder is properly authenticated and the transaction authorized.
- The *card acceptor* is the entity that accepts the card for financial transaction purposes and submits the transaction information to the

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

acquirer. A card acceptor would typically be a merchant or a financial institution.
- The *acquirer* processes the transaction for the card acceptor by confirming the identity of the cardholder and issuer, and by providing card authorization services for the card acceptor.

Our company is authorized by the STAR network to function as both an issuer and an acquirer. This means that we can issue our stored value card to cardholders and they can use the card at any Automatic Teller Machine (ATM) or Point of Sale (POS) device. We, in turn, guarantee payment to the card acceptor through the card acceptor's acquirer. It also means that, as an acquirer, we can provide our own POS and PIN-entry devices, provide acquirer and issuer services for our own cards, and offer acquirer services for cards provided by other card issuers.

When a stored value cardholder makes a purchase at a merchant for whom we do not provide acquirer services, the process works as follows. The cardholder swipes their stored value card at a POS device and inputs their PIN to initiate the transaction process. The PIN is encrypted upon entry into the PIN-entry device. The merchant's acquirer processor receives the transaction information and forwards it to the debit network with whom they have contracted for debit services. STAR, Plus (VISA), Cirrus and Maestro (MasterCard), or NYCE provide debit networks. If the merchant's acquirer is a STAR processor, STAR will receive the transaction directly and forward it to us using 3DES encryption. If the merchant's acquirer is not a STAR processor, the acquirer will send the transaction to their debit network provider who will forward the transaction to STAR. STAR will then forward the transaction to us. In either case, we will authorize or decline the transaction and respond through STAR. STAR will forward our response as appropriate.

When a PIN-based transaction is initiated from one of our transaction-originating devices (for which we act as the acquirer), the transaction information comes to us directly. From this point, there are several different scenarios:

- If the cardholder is using one of our stored value cards (for which we are the issuer), then we authorize or decline the transaction.
- If the card is provided by another issuer, we then forward the transaction information to the STAR debit network.
- If the issuer is a member of the STAR debit network, then STAR forwards the transaction directly to the issuer for authorization.
- If the issuer is a member of another debit network, STAR forwards the transaction to the appropriate debit network.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Protecting the PIN Block

The fact that the cardholder is not present in electronic payment transactions is the primary hurdle in ensuring their security. To resolve this issue, “most security measures have focused on positively identifying the various parties, authenticating transactions, and protecting the financial data that’s sent or received” (Verifone, p. 7). It is the use of PINs that have been at the center of strengthening the process of authentication and identification.

The cardholder’s PIN is the critical piece of information in the POS process because it verifies the identity of the person making the transaction (NYCE, p. 1). STAR requires the use of the ANSI x9.8 “PIN Management and Security” standard, which specifies the characteristics of a PIN block (STAR, Appendix J, p. J-11). A PIN block is a 64-bit block of data that is derived from combining the cardholder’s PIN and Primary Account Number (PAN). The PAN is the account number that specifies the issuer’s identification number and the account number of the cardholder. The PIN block format secures the cardholder’s PIN by mixing the four-character PIN with the 16 characters of the PAN. If the PIN block is not used, an attacker will know exactly where in the data the PIN is located (NYCE, p. 5). What results from combining the PIN and the PAN is a 64-bit block that fits perfectly into the 64-bit encryption blocks used by DES.

Central to legitimately authorizing a transaction is ensuring the secrecy of the cardholder’s PIN from the time it is entered into the originating device until the transaction is authorized or declined. To ensure the confidentiality of the PIN, STAR requires that it “be protected at all processing points within the STAR Network outside of the Authorization point within the Issuer Zone” (STAR, Appendix J, p. J-9). STAR requires that electronic transactions be broken up into zones. Each link on the way to completing a transaction’s authorization is a zone. For instance, there is a separate zone between a PIN-entry device and an acquirer processor, between the acquirer processor and STAR, and within the STAR network itself. Each zone uses a unique DES encryption key. This helps ensure the privacy of the PIN as it travels through processing points, and it keeps a single compromised key from violating the integrity of more than one segment of the network (NYCE, p. 3).

The fact that each zone uses a unique encryption key means that the data from the originating zone must be decrypted and then re-encrypted with the key for the next zone. This means there is a point during this translation process when the PIN block is clear text. Star requires that, when the PIN block is being translated in this manner, the process occur in a Host Security Module (HSM) (STAR, Appendix J, p. J-7). The HSM is a Tamper Resistant Security Module (TRSM). A TRSM makes determining keys, PINs, or other confidential information highly improbable. This is because TRSMs are designed to zero out any confidential information if attempts are made to penetrate the device. “Penetration includes breaking open the device, functional modification of the device’s operation by

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

programming or other means, the use of tapping devices, hardware maintenance procedures and/or other software maintenance procedures" (STAR, Appendix J, p. J-7). The TRSM that provides this service for our company is a Hewlett Packard Atalla NSP.

Brute Force Attacks on DES

The STAR debit network requires that, at a minimum, single-DES be used to encrypt PIN blocks. We have chosen to use 3DES because it is a stronger method of encryption than single-DES. "When strength is discussed in encryption, it refers to how hard it is to figure out the algorithm or key, whichever is not made public" (Harris, p. 464). With regard to DES, the algorithm is publicly known and considered secure.

While the publicly known DES algorithm has proven itself to be secure, the secret component, the key, has been found to be vulnerable to brute force key search attacks (Hewlett Packard). In fact, this kind of attack is the only known method for breaking DES. A key is a value that is composed of a large number of bits. Encryption algorithms contain a key space. The key is derived from the range of values contained in the key space. A larger key space means there is a greater number of key values available and that doing a brute force search of all possible keys will take that much longer to complete. Ultimately, the goal is to have a key space large enough to make it impractical to dedicate the time required to complete a brute force key search.

This has been found to be a problem for DES. While a DES key is made up of a 64-bit block, eight bits are used for parity. So this leaves 56 bits for the actual key. This means that to exhaustively search for all the key values, you have to try two-to-the-56th key values to find the utilized key. The feasibility of completing this kind of attack was demonstrated in January 1999 by the Electronic Frontier Foundation. They showed that a custom designed computer connected to thousands of PCs on the Internet could break DES in 22 hours. Currently, few people have this much computing power available to them. However, as time goes on and computing power continues to increase while the cost of such power decreases, DES will become more and more vulnerable.

Key Length

Given that DES is vulnerable to brute force key attacks, DES's security lies in the length of the keys and in the consequent increase in the key space. This is where 3DES has proven to be very valuable. It extends the DES algorithm by increasing the key length (i.e., the number of keys used). Increasing the key length means that there are more keys that must be brute forced in order to find the active encryption key. In addition, 3DES maintains compatibility with single-DES by

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

continuing to use the DES algorithm while simply adding additional keys and encryption cycles to the encryption process.

Triple-DES uses three cycles of encrypting and decrypting (Harris, p. 487). During the first cycle, the plaintext is encrypted. During the second, the result of the first cycle is decrypted. And during the third cycle, the result of the second cycle is encrypted. This encrypt-decrypt-encrypt process can use different key lengths:

- A single-length key (1-key DES) uses the same 56-bit key for each cycle. The result here is the same as using DES. This key length results in 56 bits of unique key material.
- A double-length key (2-key DES) uses two 56-bit keys. The first key is used during the first and third cycle and the second key is used during the second cycle. This key length results in 112 bits of unique key material.
- A triple-length key (3-key DES) uses a unique 56-bit key for each cycle. This key length results in 168 bits of unique key material.

This flexibility in key length has enabled the DES algorithm to remain viable. Making changes to the electronic financial transaction infrastructure is complicated, expensive, and time consuming. The ability to continue using single-DES while migrating to 3DES has helped make 3DES the current encryption method of choice for electronic transactions.

Key Management – Master/Session

Since the key in DES (and not the algorithm) is what is most vulnerable to attack, ensuring the secrecy of the key is paramount. What helps establish this secrecy is key management. “Key management is the method used to securely inject, change, and protect the identity of…keys” (Verifone, p. 9). There are two leading methods for managing the keys at the originating device: Master/Session and Derived Unique Key per Transaction (DUKPT).

With Master/Session, the terminal or PIN entry device is injected with a master key that is used to encrypt and decrypt the session (or working) key. The master key does not perform the encryption of the PIN block. This is carried out by the session key. The term “session” refers to the period—or session—during which the key is valid. Session keys can be changed as often as desired, even after each transaction. Changing session keys this frequently would significantly increase system security. However, it would also “greatly increase the processing and communications workload and costs for both the host system and terminals” (Verifone, p. 10). This method of key management is probably most appropriate for situations when there are not too many originating devices communicating with an HSM.

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Key Management - DUKPT

As its name indicates, Derived Unique Key per Transaction generates a new key for each transaction—and DUKPT does this without incurring the processing and communications costs of Master/Session key management (Verifone, p. 10). This efficiency occurs because each transaction contains all the information needed by the acquirer's HSM to determine the key. DUKPT uses a base derivation key which resides in the HSM of the acquirer. The PIN encryption key that each originating device uses is created with the base derivation key. DUKPT performs its tasks as follows:

- Before the PIN-entry device is sent to a card acceptor, it is injected with an initial PIN encryption key and an initial key serial number.
- The HSM, with which the PIN-entry device will communicate, contains a counter. Creating a PIN encryption key involves incrementing this counter and encrypting the new counter value with the HSM's derivation key. The resulting PIN encryption key is then transferred to the PIN-entry device. Because the counter is incremented for each new PIN encryption key, each PIN-entry device gets a unique initial PIN encryption key.
- The counter is also used to create the initial key serial number. The contents of the counter are transferred to the PIN-entry device and become its initial key serial number. Just as with the initial PIN encryption key, the counter (by being incremented) provides each device with a unique serial number.
- When the PIN encryption key is injected into the PIN-entry device, its encryption counter is set to zero. Whenever the device encrypts a PIN, this counter is incremented. When this occurs, the new counter value is encrypted using the PIN encryption key from the last transaction. A new PIN encryption key—that is unique to this transaction—is thereby created.
- While the PIN encryption key changes with each transaction, the originating device's initial key serial number does not.
- The serial number and the encryption counter are linked together and sent to the HSM with every PIN block.
- The HSM is able to determine the initial PIN encryption key of the originating device by encrypting the initial key serial number with its derivation key.
- Beginning with the PIN entry device's initial PIN encryption key, the HSM can then calculate upward to the current counter value, identify the current PIN encryption key, and decrypt the message.

In zones where we provide the POS (or originating) device and act as both the issuer and acquirer, we use DUKPT to manage keys. This method is particularly effective for situations like ours in which there are many POS PIN-entry devices communicating with relatively few HSMs. Since DUKPT generates unique keys for each transaction, it simplifies key management because transaction-

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

originating devices can be used that do not rely exclusively on security procedures and physical barriers. It is particularly effective for us because we are the sole acquirer for the POS PIN entry devices we provide. We can therefore manage the physical and procedural security of the HSM (the Atalla NSP) on site and still provide a high level of security for remotely entered PINs.

Key Management – Symmetric Keys

Triple-DES is a symmetric key algorithm. This means that both parties encrypt and decrypt data using the same key. Symmetric keys are also referred to as secret keys because each of the parties is responsible for keeping the key secret.

As discussed earlier, each zone in a STAR debit network has its own 3DES keys for encrypting PIN blocks. These keys are encrypted with a Master File Key (MFK) that is unique to the HSM that operates at the intersection of each zone. The MFK is the 3DES key that is used to encrypt all of the working keys used by the HSM. Working keys carry out specific cryptographic functions. These keys would include PIN encryption keys and Key Exchange Keys (KEKs). In the zone where we communicate with STAR, the 3DES symmetric keys we use are managed by using dual control and split knowledge.

The way these 3DES keys are managed is that STAR uses the MFK on their HSM to generate a Key Exchange Key (KEK) for the zone we share. The KEK is generated as two key components, each of which is securely delivered to a different contact at our company. Each of these contacts is responsible for loading their key component into the Atalla NSP. This ensures that the working keys are managed using dual control and split knowledge. Two people each have access to a piece of the whole key (split knowledge) and it takes the two of them together (dual control) to combine those components into a working KEK in the HSM. No single person has knowledge of the full key, and no single person can put the key into use.

The KEK works as a Zone Control Master Key (STAR, Appendix J, p. J-12). It is used to encrypt working keys that are generated by the STAR HSM's MFK. These working keys are known as the Issuer Working Key (IWK) and the Acquirer Working Key (AWK) (STAR, Appendix J, p. J-12). These keys are exchanged dynamically with direct processors. As a direct processor, new working keys are regularly generated and are provided to us encrypted under the KEK. The AWK enables us, as the acquirer processor, to encrypt PIN blocks for STAR and the IWK enables us, as the issuer processor, to decrypt PIN blocks from STAR.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Key Storage Vulnerabilities

The form of 3DES that we use with our Atalla NSP is called double-length (or two-key) 3DES. The ciphertext resulting from double-length 3DES encryption is substantially more secure than that used by DES. Just as with DES, breaking 3DES encryption requires that the attacker engage in an exhaustive key search. However, with 3DES the work involved is substantially greater. The number of key combinations that must be tried until a match is found is calculated by multiplying 2-to-the-56th x 2-to-the-56th. This results in 2-to-the-112th different key options. Based on this number of possible keys, it is estimated that it should take more than 200 trillion years to crack 3DES. (Hewlett Packard).

While this would be the case if 3DES keys were created, used and discarded after each transaction, the reality is that experts think 3DES will last approximately 10 years in the real world (Hewlett Packard). The reason for this is that 3DES keys are stored outside an HSM in places such as a host's memory or in a database. In addition, keys are also sent to other systems. The reality is that keys are mostly used in what is considered a hostile environment.

It is the keys that are stored outside an HSM that are most subject to attack. The ANSI X9.24 standard specifies that keys in a hostile environment must be encrypted with 3DES and stored in a database as two DES key components. The greatest risk to these keys comes from the inside. In fact, up to 80 percent of bank fraud comes from insider attacks from sources such as angry employees, vendors or contractors. (Hewlett Packard).

An attack on a 3DES key would target the two 56-bit key components of each key. These components would most likely be stored in a database outside the HSM. The attack could work something like this:

- Key1 and Key2 are stored in a database.
- The attacker targets Key1, which is composed of two single-length DES keys known as Key1Left and Key1Right.
- Key1Left and Key1Right are encrypted under 3DES and stored in the host database as Ciphertext1 and Ciphertext2.
- The attacker issues a command to the HSM to generate a plaintext/ciphertext pair from Ciphertext1.
- The adversary now goes off site and attacks the key with a 2-to-the-56th search to find Key1Left.
- Key2Left and Key2Right are also encrypted under 3DES and are located in the database as Ciphertext3 and Ciphertext4.
- Knowing Key1Left, the attacker can take Ciphertext1 and replace the Ciphertext3 part of the Ciphertext3/Ciphertext4 cryptogram. This modifies the key to Ciphertext1/Ciphertext4.
- The attacker then uses Ciphertext1/Ciphertext4 to create another plaintext/ciphertext pair with the HSM. The adversary takes the

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

plaintext/ciphertext pair off site and attacks Ciphertext4 with another 2-to-the-56th search of the key space to find the Key2Right (Hewlett Packard).

The attacker can continue this process of breaking keys and creating illegitimate keys. Ultimately through this process, the PIN blocks and other encrypted data in the database can be corrupted or accessed.

Atalla Key Block

What this attack demonstrates is that while 3DES can provide confidentiality, it cannot ensure message integrity. The way to provide data integrity and truly secure key management is through the use of a key block. "A key block is a data structure used to store or exchange cryptographic keys within hostile environments." (Hewlett Packard).

As described above, simply encrypting keys is not a sufficient means of securing keys in an electronic transaction environment. In the attack described, keys can be used in a manner for which they were not intended and knowledge of working keys can damage the security of other keys. A key block can counter these problems by ensuring that:

- Key encryption will be accomplished using a proper key size and a secure algorithm.
- Through the key block, the HSM will have control information that will enable it to determine that keys are being used correctly.
- Any changes to or manipulation of the encrypted keys or control information will be detectable (Hewlett Packard).

Atalla Key Block Structure

The Atalla Key Block (AKB) provides these features by using a structure consisting of three parts: a header, an encrypted key field, and a Message Authentication Code (MAC) value (Hewlett Packard).

The header stores key attributes such as the algorithm and key usage parameters. Before an Atalla NSP uses a key in the Atalla Key Block format, the header is checked to confirm that the key is being used properly.

The encrypted key field holds the key data. This field is encrypted with 3DES using Cipher Block Chaining (CBC) mode. Triple-DES is a block cipher. Data is encrypted in blocks of 64 bits. CBC is the mode of operation that this encryption process follows. With CBC, the encrypted blocks of data are "chained" together. The plaintext being encrypted is XORed ("exclusive OR" is a substitution cipher) with the cipher text of the previous block. This means that if any changes are

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

made to the order of the blocks, the decrypted data becomes garbled. This chaining process also hides any patterns and makes the resulting ciphertext more random (Harris 486). In addition to thoroughly encrypting the key data, it should also be noted that this key field is able to hide more vulnerable single-length keys by padding the key field. Through this padding technique, all keys appear to be the same length.

Triple DES using CBC does provide increased security by making it difficult to rearrange ciphertext blocks. However, key integrity can still be jeopardized, as described in the 3DES attack, when parts of keys are substituted for other key components and thereby provide knowledge of other keys (Okiok, p. 6). By substituting key components, brute force can be used on single-DES keys twice. "This can be done on average in 2 x 2-to-the-55th= 2-to-the-56th steps, which is much smaller than 2 to the 112th (the number of steps needed to try each and every double length 3DES key)" (Okiok, p. 6). It should be noted that these attacks are effective even when CBC is involved.

This is where the AKB's MAC field comes in. It ensures the integrity of the key and prevents tampering. Using a MAC involves implementing a one-way hash. The data is run through a publicly known hashing algorithm. This creates a value (known as a message digest) that represents the data. The sender then attaches this value to the data and sends the data to the recipient. The recipient runs the data just received through the same hashing algorithm and compares the resulting value with the value sent by the sender. If both message digests match, the implication is the data has not been changed or tampered with en route. However, it is possible for someone to intercept the data, make changes to it, compute a new message digest, and send that falsified message digest to the originally intended recipient. The recipient will then run the data through the hash, find that the resulting message digest matches the falsified message digest, and erroneously conclude the integrity of the data is intact.

Using a MAC value solves this vulnerability. A MAC value is the result of combining data and a symmetric key, and running them through a hashing algorithm. When using a MAC value, the sender utilizes a symmetric key that is shared with the recipient and links it with the data. This combination is then run through a one-way hashing algorithm. The MAC value that results is appended to the data and sent to the recipient. If it is intercepted, the interceptor cannot recalculate the message digest because the interceptor does not possess the symmetric key. Only the intended recipient and the sender have that component. When the recipient receives the data, the recipient concatenates the symmetric key with the data, runs the result through the hash, and compares the message digests. If they are identical, data integrity has been maintained.

When the Atalla NSP is going to use an AKB formatted key, it can verify the integrity of that key by using the MAC. The MAC value of the key can be confirmed by the NSP because the NSP has the symmetric key upon which the

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

AKB MAC is based. The MAC cryptographically binds the header and key fields together. In effect, the MAC "cryptographically tie[s] the functionality of a key to the key itself…" (Okiok, p. 6). This means an attacker cannot manipulate parts of the key without generating erroneous MAC values and revealing that tampering has occurred. The NSP would detect the bad MAC value and not use the illegitimate key. As a consequence, key integrity is maintained.

To summarize, using the AKB secures 3DES keys by keeping an attacker from:

- changing the attributes or bits of keys;
- utilizing part of a key as if it were a whole key;
- changing the arrangement of a key;
- putting parts of a key into a different key;
- recognizing less secure keys (Hewlett Packard).

Conclusion

As can be seen from the above discussion, 3DES, in combination with good key management, can provide strong security for the processing of electronic financial transactions. By implementing DUKPT with 3DES, we are able to use unique keys with each transaction in the acquirer zone. When interfacing directly with STAR, split knowledge and dual control promote the secure management of symmetric keys between the STAR debit network and our own. The Hewlett Packard Atalla NSP and the Atalla Key Block ensure that PIN blocks and 3DES keys are managed and stored securely. In addition, using these Atalla tools make it possible to remedy many of the data integrity issues to which DES is susceptible. As a consequence, by implementing these technologies where appropriate, our company is able to provide customers with issuer and acquirer services that ensure the integrity, confidentiality, and overall security of their electronic financial transactions.

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

References

Accredited Standards Committee X9, Inc. “American National Standard for Financial Services: ANS X9.24-2002.” American National Standards Institute. Approved November 8, 2002.

Global Insight Group Ltd. “Triple DES in the POS Environment.” URL: http://www.gigl.net/document/triple-des.pdf (June 26, 2004).

Graf, Michelle. “Securing Trust in the Payment Industry.” URL: http://www.greensheet.com/PriorIssues-/030501-/7.htm (June 29, 2004).

Harris, Shon. CISSP Certification All-In-One Exam Guide. New York: McGraw-Hill/Osborne, 2003.

Hewlett Packard. “Atalla Key Block.” URL: http://h20138.www2.hp.com/object/ATKEYBLWP.html (June 23, 2004).

Koc, Cetin Kaya. “Security & Cryptography Notes.” Fall Term 2003 – CRN: 17733. School of Electrical Engineering & Computer Science, Oregon State University. URL: http://islab.oregonstate.edu/koc/ece399/sc/05-AES.pdf (June 25, 2004).

NYCE. “Best Practices for PIN Encryption.” URL: http://www.nyce.net/pdf/PIN_debit_encryption.pdf (June 24, 2004).

Okiok Data. “Why Migrating to Triple DES is Not Easy.” 2002. URL: http://www.okiok.com/documents/white_papers/Why%20Migrating%20to%20Triple%20DES%20is%20Not%20Easy.pdf (June 27, 2004).

Star Systems. “Appendix J: Security and Compliance Guide.” Star Network Operating Rules. October 2001.

Star Systems. “How Does a PIN-Secured Debit Transaction Work?” URL: http://star.com/?go=retailers.PINSecuredGuide (June 25, 2004).

Tropical Software. “ DES Encryption: Overview.” URL: http://www.tropsoft.com/strongenc/des.htm (June 22, 2004).

Tropical Software. “Triple DES Encryption: Overview.” URL: http://www.tropsoft.com/strongenc/tripledes.htm (June 22, 2004).

Verifone. “Securing Trust in the Payment Industry.” URL: http://www.verifone.com/pdf/Security_white_paper.pdf (July 15, 2004).

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.