Global Information Assurance Certification Paper

Cryptography: At Work in the Business

A SANS GSEC Certification Paper
Version 2.3

Kendrick Conner

November 4, 2004

“Good cryptographic algorithms are proven through time; great cryptographic algorithms are never proven.”

Abstract

Cryptography has always been a hard field to understand. It often deals with complex mathematics that few of us ever learned or have forgotten about many years ago. So often in the IT industry we learn about certain technologies and maybe how they will benefit us. Since the beginning of the security push we have added on new ways to increase our level of security. This paper is intended to give a big picture view of the technologies involved with Cryptography as a whole in business. I will briefly cover the major concepts in cryptography, how they are used in the overall C.I.A implementation and how it works in business. Every attempt will be made clarify the technology in use and how that impacts the business decision.

Why we use cryptography

Cryptography and C.I.A.

The main goal for using cryptography is to address the C and I in C.I.A. (Confidentiality, Integrity and Availability). Maintaining the C and I are often necessary to obtain the Availability. Data that is owned by corporations, government and other organizations can be critical in nature such as patient records, top secret military records and trade secrets. The government mandates how certain information is stored and transmitted. In a global marketplace, counties have different levels of requirements for securing data. The European Union holds privacy of the individual to a higher standard. For example internet service providers are required to prevent, "unauthorized access to communications in order to protect the confidentiality of communications..."[1] Here in the United States the most visible form of government mandated data storage and transmission is Health Insurance Portability and Accountability Act (HIPAA). HIPAA started back in 1996, in general is a law requiring certain privacy protections, security and digital signature standards along with the portability of health records.

Availability

HIPPA is a great example of how in order to satisfy availability you need confidentiality and integrity. In order to make private documents available you must have a confidential way of holding and transmitting those documents and verifying they have not been changed through un-authorized means. HIPPA requires that health organizations of all types guarantee that "the integrity, confidentiality, and availability of electronic protected health information they collect, maintain, use, or transmit
is protected."[2]

[1] European Parliament, Directive 2002/58/EC.
[2] Federal Register Online, p. 8335.

Confidentiality and Integrity

Government laws are not the only reason to use encryption. These days personal information, where not protected by law, is being demanded by consumers. An FTC report states:

> Surveys have shown that increasing numbers of consumers are concerned about how their personal information is used in the electronic marketplace. These findings suggest that consumers will continue to distrust online companies and will remain wary of engaging in electronic commerce until meaningful and effective consumer privacy protections are implemented in the online marketplace.[3]

Statements likes these show the importance of maintaining confidentiality, not only in the internet marketplace but also in the more traditional brick and mortar companies. In these last few years there has been a number of reported break-ins into databases holding personal and credit card information. This loss of confidentiality has cost those businesses both in terms of public trust and in the cost associated with resolving the issue. If a company becomes the victim of information theft, the amount of value a statement such as "while the information was stolen it was encrypted with RSA and thus of no value to the thieves" goes a long way towards assuring consumers that their data is secure.

Cryptography in the workplace

Keberos

Keberos is an authentication protocol using a trusted third party to authenticate entities on the network. In order to run Kerberos you need to install Kerberos software on both the server and the client. The individual enters their password which is combined with ultimate destination information and encrypted using a separate encryption algorithm. This info is given to the Key Distribution Server (KDC) which is the middleman negotiating for you. He grants you a ticket good for a particular amount of time. This is so old tickets cannot be used again by anyone who gets a hold of it including the original owner. Keberos is like going to the movies. You pay your money and are issued your ticket; you can leave and come back as much as you want as long as your ticket is still valid for the date and showing. It doesn't allow you to go see other showings; if you want to do that you have to get another ticket. And if you loose or throw your ticket away at the end of the show, even if someone picks it up, it is no longer valid and won't get you or anyone else in.

VPNs

Virtual Private Networks (VPNs) are said to be like data tunnels, where only you have access to the tunnel. I don't think this is a good description. VPNs are more like a speed train running at night. If you're close enough you can hear

[3] Federal Trade Commission, section. 2, par. B, subsection 2.

them but you still can't tell what kind of train it is, where it is going or what is on board nor can you get on board. Just knowing it is there doesn't help you to determine what is being carried on them. Along with the trains ability to secure travel trains can also use multiple paths to determine how to get to its destination and once the best route is chosen it sticks to that path. VPNs accomplish this communication path and anonymity by using special hardware and/or software that, acting like a proxy, hides the real destination of the data and then encrypts the data so that if it is seen you have no idea what it really is. For all intense and purposes, it is just noise on the line. Vendors use different methods for creating VPNs and they can vary widely. Usually you need to set up with the same system or vendor on both sides of your train track. At which point how you encrypt the data also can vary. One method is to use L2TP for setting up the session and IPsec between the gateways for encryption and there are numerous other ways.

Radius

Radius is one part to secure remote access. Radius encrypts only the authentication process, it has good logging capabilities and is great for tracking usage. However, it relies on IPSEC or some other encryption protocol to secure the data. It was originally used for dialup users such as customers of an ISP.

SSH

SSH was developed to allow secure communication to remote computers and looks very much like a telnet session. Both the client and the server need to have RSA or DSA public keys.[4] This allows SSH to encrypt all communication from the moment the session starts. SSH, now currently on version SSH2, can run on just about any operating system. Programs like Putty, a freeware program, allows anyone to easily set up a secure session to remote systems. Since Putty is available for multiple operating systems it has made SSH implementation to be relatively painless and SSH easy to use. This has helped to make SSH an integral part to system security.

IPsec

IPsec is probably the most widely used standard for encrypting communications. It provides for all three of the C.I.A. tenets. IPsec can utilize manual key or automatic key distribution with automatic key distribution almost mandatory for large implementations. The most well used automatic key distribution used with IPsec is Internet Key Exchange (IKE). IKE uses a very complex system for distribution and authentication of keys. Together, IPsec and IKE create two sets of keys for encryption. The first key is made so the second key can be shared securely which in turn is used for securing the actual data.

IPsec can be used in tunneling mode or transport mode. The biggest difference between the two is in the path they take in the communication process. In

[4] Cobb, p. 186

transport mode two computers communicate back and forth with only the data payload of each packet encrypted. If you were watching the traffic go back and forth you would know who was talking to whom but not know what they were saying. This is like watching from a distance someone whisper to another person. In tunneling mode, it's like two people whispering in the dark. You know people are talking, but you don't know who they are or what they are saying. In tunneling mode there is an intermediary such as a gateway acting as a proxy. The two gateways appear to only be communicating to themselves but, once they receive the traffic they pass it on to the person it belongs to at either end. The real source and destination addresses are encrypted with the rest of the packet.

PGP

Pretty Good Privacy (PGP) invented by Philip Zimmerman, is encryption for the masses.[5] Unlike PKI which generally uses a third party such as a certificate authority(CA) for issuing digital certificates, individuals sign their own digital certificate. The idea behind PGP was to create a way for the average person to use an encryption program or encryption for the masses. While it can encrypt files, hard disk volumes and even network communications it is most widely used for the encryption and signing of email. A note on PGP: PGP is still free to individuals however requires a businesses or anyone using it for commercial purposes to purchase licenses for its use.

Cryptography and E-commerce

Business on the internet has grown substantially in the last several years and with it, has come the need to protect the information transferred back and forth during those transactions and sometimes even where that information is stored. Several large and public disclosures have been made when unauthorized access has been gained to private information such as credit card numbers. In February of 2003 Fred Katayam wrote an article titled Hacker hits up to 8M credit cards in which possible access to 8 million credit card accounts had been made. In the article they estimated the cost, "If 8 million card accounts were affected, and all those cards were canceled with new cards issued in their place, it would cost the credit card companies an estimated $200 million, according to credit card experts." In September 2002 a security flaw in Microsoft's CryptoAPI was discovered. "The flaw could let a Web site with a valid certificate issue a second; invalid one, which could enable unauthorized access to a computer as well as, among other things, the theft of user passwords or credit card numbers." Wilcox's Credit card theft feared in Windows flaw.

In order to stop this from happening, many technologies have been developed and implemented. Most of these are built around cryptography. In order to provide secure transactions, verify someone or something is what it says it is and secure data in possession, several technologies have stood out from the pack.

[5] Krutz and Vines, p. 218

SSL/TLS

Netscape was one of the first companies to realize and implement a secure way to transmit e-commerce data. They did this through incorporating verification and transaction security into their browser via SSL. SSL version 3 has become the standard and the Transaction Layer Security (TLS) standard is directly derived from that by the Internet Engineering Task Force (IETF). SSL/TLS is indicated in the browser in two ways; one is the use of HTTPS:// and the other is by a little lock or key icon located in the browser itself. SSL/TLS are standards that allow you to use an encryption algorithm to encrypt your data passing between you and the server. SSL/TLS is used like IPsec to encrypt data for transmission across public network. The difference is that SSL/TLS is mainly used for Web/HTTP communications across public networks and IPsec for business communications across public networks. Another difference is IPsec usually requires extensive hardware on both sides of the communication. SSL/TLS relies on the client machine and the server to do the encryption

XML

XML is a language that looks similar to HTML but, it is really used to categorize data. What this allows you to do is put your data into a form that other systems can recognize, transmit and store. How this works in the security realm is simple. By using XML on a web page you can set different levels of security for that web page and set what needs to be encrypted and what does not. This alone can decrease your transmission overhead by only encrypting what needs to be encrypted and at what level. XML is not just limited to web page use; many documents that define categories use it. For example, URL filtering list categories that are allowed or blocked from access. You may have the categories gambling and sports as blocked, and allow news. You can set up the XML file to put certain sites into the gambling category and put the gambling category into the blocked category. Instead of writing up a script or special program to do this, you can use XML and define your categories, what the variables are and what goes in them.

Digital Signature

In M. Hellman's article, An Overview of Public Key Cryptography a digital signature is a number that can only be created by the sender, appended to a message, that allows the receiver to validate the message was made by the sender. Digital signatures usually use your private key to only encrypt a message hash. This keeps the length of what needs to be encrypted with your private key to a minimum.

Digital Certificate

Digital certificates are used to validate a person or entity against who they claim to be and contain their public keys. There are different levels of certificates from personal to enterprise and the requirements for each vary on the level. Certificates are handed out by certificate authorities (CA). Certificate authorities are usually third parties who are trusted to issue certificates. However, large

companies can set themselves up to be a CA for internal use. Digital certificates are a fundamental part of the public key infrastructure (PKI). PKI uses digital certificates to share public keys that are digitally signed. In essence the digital certificate is a combination of the public key and the digital certificate, which allows you to both validate and securely communicate.

Making e-commerce secure: Putting it all together

So now that you know some of the technologies used to make e-commerce secure let's see how it all works together. You start out as consumer going to a business's website with your SSL/TSL enabled browser and your personal digital certificate. This site likely has been issued a certificate from a certificate authority such as Verisign. You log into the site with you personal username and password. Depending on the site you may now be encrypting the data you send and receive. After selecting your items to be purchased you move to check out, if you weren't already encrypting data back and forth at this point you most likely are now and should see a locked icon in the lower left of the browser window. If you hover over the lock icon you will see what type of encryption and level it is using. For example my communication with Amazon.com uses SSL at 128 bit strength. At this point you sent and also have also received a digital certificate. In some browsers you can view the website's certificate by right clicking on the web page and selecting properties. You will see a button that says "Certificates." In the certificate you will find the encryption type and some detailed information In Amazon.com's case, the certificate shows that the digital signature is using the SHA1 RSA algorithm to verify that the public key using RSA algorithm with 1024 bit encryption is genuinely sent from them. You also have your own digital certificate which you can view but, details will not be provided here. Via this process your pc and the e-commerce website have established their identities and have determined how they are to communicate securely. This process also allows for non-repudiation which means you can not send something such as a request to purchase a book and then say you never did. Because you sent a digital certificate you essentially signed your order.

Concepts of Cryptography

Hashes

Hashing is used for the integrity of data. Hashing is an algorithm that creates what's called a message digest. The message digest is a string of numbers and character that together are unique to an individual message. Given the same hashing algorithm and the exact same message you will get the same message digest. However, make one change to that message, even as small as adding a space to the text, and it will completely change the message digest. This allows you to compare the message digest to the message digest you created and verify the message has not been changed. By using the same hashing algorithm on the message you received you should get the same message digest; if you don't then the message has likely been changed in transit. You need to encrypt

the message digest or the message itself so that it is impossible to create a new message digest that would work with an altered message.

Asymmetric vs. Symmetric

Symmetric

Symmetric keys are where both the sender and receiver have a copy of the same key. These keys require less computation power to encrypt and decrypt than asymmetric keys do. The problem with symmetric keys has always been how to distribute them.

Asymmetric

The idea that we could encrypt something with one key using a one way function, but only decrypt it with another started back around the time of WWII. One way functions is easy to compute in one direction but hard to compute from the other direction. The easiest version of a one way function is through the use of prime numbers. Using prime numbers is one form of one way functions, there are other mathematical concepts that can be used such as elliptic curve computations, which are also one way. The meaning of one way is not exactly correct as the numbers "can" be calculated but it is exponentially harder to go backwards. Take for example the numbers 23 and 65 if you multiply those you get 1495. Now, take the number 676 and tell me what two numbers I used to create it. You can see that it is easy to create a number then to figure out how it was created. Did I use 2 x 338, 4x169, 13x52 or 26x26? Those aren't the only possibilities either. This is the concept behind one way computations or Asymmetric encryption. If you would like to see more of how the concept works please see appendix A.

Block Cipher vs. Stream Cipher

Block Cipher

Block cipher separate the data into chunks and then encrypt the blocks seperately. For example, if you had a 128 bit document and you used a 64 bit block cipher to encrypt the document you would actually end up with 2 encrypted pieces that together make up the whole document. Each piece is encrypted with the same key and since the key can only handle 64 bit chucks of data, it breaks the document in half and encrypts each piece separately. If the document was of an uneven length say 120 bit then the algorithm used to encrypt would pad the last block with 8 bits of random bits to make the whole message a total of 128 bits. Since it is highly possible for 2 64 bit data chunks to be of the exact same data, I.E. a statement in the document is repeated twice, you need to be able to identify and encrypt them separately. To do this an initialization vector (IV) is added. The IV adds some randomness to every block of data. The quality of the IV can vary and if poorly implemented can make things worse.

Stream Cipher

Stream ciphers are similar to block ciphers in some sense. However, instead of encrypting large chunks of data they encrypt the data in smaller amounts such as a byte. A stream cipher also uses just one large key to encrypt. This key needs to be as large as the data being encrypted. Stream ciphers can be much quicker to process then block ciphers. Their weakness is that you must know the data size before encrypting or the process is stopped to re-key which pretty much negates any processing gain you receive from using a stream cipher. For these reasons stream ciphers are not as popular as block ciphers.

Bit size matters

The bit size of an encryption key can affect the time it takes to break and also the computational power required to use it. To illustrate purposes I'll use a really small key such as 5, 1, 6. In order for someone to figure out this combination they must do 10^3 calculations as there are 10 possibilities for each of my three numbers 10x10x10=1000 possible combinations. You can see how quickly a computer could try all these combinations. Since keys are set in bits meaning your choices are either a 0 or a 1 then you would have 2^3 possible combinations or 2x2x2=8. By increasing your possible combinations by just 3 times as many numbers or 2^9, you have 2x2x2x2x2x2x2x2x2=512 possible combinations. You can quickly see how a number such as 2^{56} can significantly increase the number of possible solutions to 72057594037927936. Keys with the length of 2^{56} have long since been replaced with 1024 bit keys or 2^{1024} and many to 2^{2048}. However, bit size is not the only factor as the algorithm used to create the actual key can affect this. A poorly designed algorithm might not truly randomize the key, making it easier to crack

Asymmetric vs. Symmetric

Because symmetric keys are inherently more secure then asymmetric keys, meaning they are harder to brute force than the factoring difficulty of the asymmetric key, employing them both means choosing the right size key. It is generally considered that a 64 bit symmetric key has the same strength as a 512 bit asymmetric key. There are comparison charts available for this and should be used when deciding on the size of each. Another difference it asymmetric keys require quite a bit more computational power to encrypt and decrypt. This is why asymmetric keys are primarily used to encrypt things like the symmetric key that is used for the actual message or data.

Main Algorithms of Cryptography

Asymmetric Algorithms

Diffie-Hellman

Whitfield Diffie and Martin E. Hellman were some of the first private researchers to come up with a way to provide secure key exchange and digital signatures in

their land mark paper called New Directions in Cryptography. In this paper Diffie and Hellman provided a mathematical one way function that could be used to set up a private and public key system and how using those keys you could digitally sign a document. It is important to keep in mind that Diffie and Hellman's algorithm was designed for key exchange and not for message encryption. It wasn't until the RSA algorithm was invented that public key exchange along with encryption was made possible.

RSA

RSA named after the three inventors has proven to be one most well used of the asymmetric algorithms for public key distribution and encryption. RSA was invented in 1977 by Ron Rivest, Adi Shamir and Len Adleman shortly after the Diffie-Hellman paper was published. The algorithm uses large prime numbers to create the private and public keys. Since asymmetrical encryption requires much more processing power than symmetric encryption, the use of symmetric algorithms is used for encryption of data and asymmetric keys are used to encrypt the symmetric key and hashes. RSA is typically combined with one of the symmetric keys such as 3DES, AES etc. RSA has many features that together can provide a complete package of public key encryption (PKE); this can include digital signatures and certificates, public and private key creation and public key distribution.

Elliptical Curve Cryptography

Elliptical curve cryptography (ECC) is the bright star in asymmetrical encryption. Unlike other asymmetrical encryption algorithms such as RSA, ECC's computational power needed for encrypting and decrypting is much lower. Also, ECC is more secure as the computational power to break it is much higher. Instead of using prime numbers and factoring, ECC uses the principal of elliptic curve discreet logarithm.[6] ECC can be used with existing algorithms; you simply replace whatever mathematical formula, such as the prime numbers with the elliptic curve. One thing to watch out for, and maybe the reason ECC hasn't been implemented more, is that ECC needs more time to be proven.

Symmetric Algorithms

MD5 and SHA-1

These are really hashes rather than symmetric algorithms since the result is not decrypted. MD5 and SHA-1 are widely used hashes and until recently were considered quite secure. There have been reported vulnerabilities in both SHA implementations and MD5. However, there is not much need for concern and the breaking of a hash will not provide much use to data already transmitted. Because a hash is only used for a particular message or key, once that message has successfully been sent the hash is no longer needed. There are other

[6] Krutz and Vines, p. 207

implications such as their use with digital certificates but time is still available to design a new hash before MD5 and SHA-1 become obsolete.

Triple DES

In 1997 DES was cracked in 96 hours.[7] Before DES was cracked there was an effort to create the next iteration which was named 3DES. 3DES also known as triple DES originally used 3 separate keys. The first one encrypting the plain text, the second one encrypting the cipher text, and the third re-encrypting the cipher text. Triple DES is a fill in standard until the next symmetric encryption standard called AES was approved.

AES

The National Institute of Standards and Technology (NIST) announced in 1997 a competition to come up with a new encryption standard to replace DES. This new standard was to be known as the Advanced Encryption Standard (AES). The algorithm *Rijndael* was chosen for use in AES. As with all encryption algorithms it must be proven over time. So, AES is not widely used yet and there is still some distrust but, momentum is gaining. In the mean time DES and 3DES along with a few other encryption algorithms such as IDEA are still being used.

Putting it all together

PKI

Public key infrastructure (PKI) takes many of the technologies described above and puts them all together as we saw earlier in the e-commerce example. The "I" in PKI stands for infrastructure. This is the total infrastructure needed to set up the creation, distribution, maintenance, verification and revocation of keys. This infrastructure includes the hardware and software necessary to provide all the above functions and can include a third party for the certificate authority. Heavily used for e-commerce, PKI is also used in businesses internally. Setting up a PKI, usually in large businesses, can prove to be a good way to secure data, communications and access on the company's network. While setting up a PKI and an internal CA can be a complicated process, it does allow for some of the most secure access and verification procedures. Instead of relying on a user directory to setup access and permissions, you can issue certificates that allow employees to access system all over the network. For example one certificate can be used to secure and verify a session to a Linux server a Windows server, a UNIX server or whatever kind of server you are running. In addition these certificates can be used to identify systems, their roles and policies. Since certificates are not just used for entity or person identification, supplier and buyer systems can be automatically set up to talk with each other. If you set up a new firewall at your location, that requires connection to your customers firewall, the connection can be automatically verified, communications encrypted, and the

[7] Mel and Baker, p. 41

session up and running quickly, without the need for a lot of manual configurations.

PGP

As mentioned earlier, PGP can be used for encrypting emails, any attachments and digitally signing the email among other things. This comes in handy when you email back and forth to clients, business partners, customers etc. Not only does it allow you to encrypt your communications that go outside of your company, but your business partner who receives a digitally signed email can be secure in knowing it is actually from you and not some email worm. PGP is also sometime required by your customers. If the information you are sending them is confidential in nature or contractual, you and they will want to be assured that the correspondence is signed and the integrity verified by both parties.

Data Encryption

What's the purpose of encrypting your communications if, once the data arrives, it can be accessed in its unencrypted form? We all know how often crackers attempt to and do gain access to the servers we have tried so hard to protect. Encryption of data is to give you that extra layer of protection or defense in depth if your server is ever compromised. California passed a law requiring all personal information to be protected. This law also applies to companies out side of California if they retain California resident's information. This means just about every e-commerce website and most brick and mortar companies are bound to this law. No company wants their proprietary or private information out in the public. What if an internal memo was accessed and published publicly? What if that memo indicated your company's process for contract negotiation with your customers? You might loose a lot of your ability to negotiate contracts, loose customers and now the competition knows how your process works. Encryption can also help with the "need to know." By encrypting data only the people that should be able to see it can. So, administrators that need to work with the data, such as backups and moving it around can still do their job, but do not have the ability to actually look at the data. By hashing the data file you can also make sure that it maintains its integrity. This can be very important when trying to maintain a single source file used in generating other files or documents. You'd want to know if someone changed your customer contract wouldn't you?

Summary

Hopefully I've been successful in explaining the various technologies used in cryptography, how it's used together and why it's so important in business. We have seen that not only do laws require certain data be protected but there is also a business need to ensure information integrity, verification and confidentiality. Additionally laws such a HIPPA require certain information to be securely made available to other entities and people. The information that businesses hold is vital to their business and if that information is released, can cause quite a bit of damage financially if not also in public opinion. Data

encryption can be your last line of defense against all the different people who want access to it from crackers, competition and un-authorized employees.

Appendix A

Note: This description is only for an illustrative point and may not be how asymmetrical encryption actually works.

Creation of a Public key:
First decide on your private key (PK). Next develop a table of numbers each number, call these your special numbers (SN). Here I have 33, 61, 27, 22, 13 and 55. However, no three combinations of special numbers can equal the same number. For example 33+61+27=121, no other combination of SNs can = 121. Each one the special numbers times your private key equals a unique number (UN). You can now distribute these unique numbers to the public.

PK*SN1=UN1	PK*SN2=UN2	PK*SN3=UN3
57*33 = 1881	57*61 = 3477	57*27 = 1539
PK*SN4=UN4	PK*SN5=UN5	PK*SN6=UN6
57*22 = 1254	57*13 = 741	57*55 = 3135

Creation of Secret Key:
In the creation of the Secret key sender chooses any 3 of the UNs, sum them up and that gives them the secret key (SK)
UN2 + UN3 + UN5 = SK
3477+ 1539 + 741 = 5757

Figuring out the secret key
In order for you to decrypt you need to know what was used for the three UNs. To figure this out you take the summed UNs and divide by your PN this gives you the sum of the three SNs
5757 / 57 = 101
Now you have the sum of three SNs. Since you know the numbers used (33, 61, 27, 22, 13 and 55) and you know that only one set of three SNs can equal 101 you can now figure out what three UNs where used.
61 + 27 + 13 = 101
Multiply the three numbers individually by your PK and you have
57*61 = 3477
57*27 = 1539
57*13 = 741

Decryption of Secret Key:
Using the three UNs you can now decrypt the Secret Key.

Notes:
Because the unique numbers are all that is made public, it would take quite a while to figure out which ones made up the large number. Since you know both the secret numbers and your private key it is much simpler for you to figure out through a few computations what three numbers where used. This example is an over simplification and in fact would be quite easy to break. However since most people do not have knowledge of the higher math that real cryptography uses this example was simplified. The weakest area in this example is that of the unique numbers. Since the unique numbers are known you could easily run a program to compute all the number to determine the three unique numbers used. By increasing the size and number of the unique numbers you increase the time it would take to compute the unique numbers used.

Appendix B

Technology	What it works with	Provides
Kerberos	Encryption Algorithm such as DES	End-to-end authentication and communication security in a trusted network.
Radius	Encryption Tunneling protocol such as IPsec	Secure remote dial-up access with utilization tracking.
SSH	Public key exchange such as RSA	Secure “telnet” like session to remote computers.
VPN (Virtual Private Network)	Encryption Tunneling protocol such as IPsec	Secure communications over public networks.
IPsec	Key Exchange	Encryption for network traffic
SSL	Public key exchange such as RSA	Secure Web/HTTP communications
RSA	Symmetric Keys such as 3DES	Verification of sender and encrypts data
PGP	Public key exchange such as RSA	A way to Creates and distributes public keys, encrypt emails and data
XML	Various encryption technologies among other things	Ways to categorize data security, usually used on websites
Digital Signature	Hashing programs such as MD5 and SHA-1	Non-repudiation
Digital Certificate	Public keys and hashes	Verification of entities such as a company or person
Hashes	algorithms such as MD5 and SHA-1,2	Data integrity
Diffie-Hellman	Public keys and digital signatures	Way to set up private, public key and digital signature system

List of References:

Bruce Schneier, Applied Cryptography (Indianapolis: Wiley, 1996)

Chey Cobb, Cryptography for Dummies (Indianapolis: Wiley, 2004)

Diffie, W. and Hellman, M. "New Directions in Cryptography." *IEEE Trans. Info. Th.* **22**, 644-654, 1976.

European Parliament. "Directive 2002/58/EC." 12 July 2002. European Parliament and the Council of the European Union. <http://europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexplus!prod!CELEXnumdoc&lg=en&numdoc=32002L0058&model=guichett>

Federal Register Online. "Health Insurance Reform: Security Standards. Health Insurance Portability and Accountability Act of 1996" 20 February 2003. (Volume 68, Number 34) Page 8335 <http://edocket.access.gpo.gov/2003/03-3877.htm>

Federal Trade Commission. "Public Workshop on Consumer Privacy on the Global Information Infrastructure." October 1996. section. 2 par. B subsection 2 <http://www.ftc.gov/reports/privacy3/history.htm#Privacy%20Concerns>

H. X. Mel and Doris Baker, Cryptography Decrypted (Boston et. Al: Addison-Wesley, 2004)

Katayama, Fred. "Hacker hits up to 8M credit cards." 27 February 2003. CNNmoney. <http://money.cnn.com/2003/02/18/technology/creditcards/>

Martin E. Hellman. "An Overview of Public Key Cryptography." May 2002. IEEE Communications Magazine. <http://www.comsoc.org/livepubs/ci1/public/anniv/pdfs/hellman.pdf>

Ronald L. Krutz, and Russell Dean Vines, The CISSP Prep Guide Gold Edition (Indianapolis: Wiley, 2003)

Wilcox, Joe. "Credit card theft feared in Windows flaw." News.com 6 September 2002. <http://news.com.com/2100-1001-956729.html>

Last Updated: July 21st, 2017

Upcoming Training

SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Trenton SEC401	Trenton, NJ	Aug 21, 2017 - Aug 26, 2017	Community SANS
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS