

401.4

Cryptography and Risk Management

SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

401.4 Cryptography and Risk Management

SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

Copyright © 2017, Secure Anchor Consulting. All rights reserved to Secure Anchor Consulting and/or SANS Institute.

PLEASE READ THE TERMS AND CONDITIONS OF THIS COURSEWARE LICENSE AGREEMENT ("CLA") CAREFULLY BEFORE USING ANY OF THE COURSEWARE ASSOCIATED WITH THE SANS COURSE. THIS IS A LEGAL AND ENFORCEABLE CONTRACT BETWEEN YOU (THE "USER") AND THE SANS INSTITUTE FOR THE COURSEWARE. YOU AGREE THAT THIS AGREEMENT IS ENFORCEABLE LIKE ANY WRITTEN NEGOTIATED AGREEMENT SIGNED BY YOU.

With the CLA, the SANS Institute hereby grants User a personal, non-exclusive license to use the Courseware subject to the terms of this agreement. Courseware includes all printed materials, including course books and lab workbooks, as well as any digital or other media, virtual machines, and/or data sets distributed by the SANS Institute to the User for use in the SANS class associated with the Courseware. User agrees that the CLA is the complete and exclusive statement of agreement between The SANS Institute and you and that this CLA supersedes any oral or written proposal, agreement or other communication relating to the subject matter of this CLA.

BY ACCEPTING THIS COURSEWARE, YOU AGREE TO BE BOUND BY THE TERMS OF THIS CLA. BY ACCEPTING THIS SOFTWARE, YOU AGREE THAT ANY BREACH OF THE TERMS OF THIS CLA MAY CAUSE IRREPARABLE HARM AND SIGNIFICANT INJURY TO THE SANS INSTITUTE, AND THAT THE SANS INSTITUTE MAY ENFORCE THESE PROVISIONS BY INJUNCTION (WITHOUT THE NECESSITY OF POSTING BOND), SPECIFIC PERFORMANCE, OR OTHER EQUITABLE RELIEF.

If you do not agree, you may return the Courseware to the SANS Institute for a full refund, if applicable.

User may not copy, reproduce, re-publish, distribute, display, modify or create derivative works based upon all or any portion of the Courseware, in any medium whether printed, electronic or otherwise, for any purpose, without the express prior written consent of the SANS Institute. Additionally, User may not sell, rent, lease, trade, or otherwise transfer the Courseware in any way, shape, or form without the express written consent of the SANS Institute.

If any provision of this CLA is declared unenforceable in any jurisdiction, then such provision shall be deemed to be severable from this CLA and shall not affect the remainder thereof. An amendment or addendum to this CLA may accompany this courseware.

SANS acknowledges that any and all software and/or tools, graphics, images, tables, charts or graphs presented in this courseware are the sole property of their respective trademark/registered/copyright owners, including:

AirDrop, AirPort, AirPort Time Capsule, Apple, Apple Remote Desktop, Apple TV, App Nap, Back to My Mac, Boot Camp, Cocoa, FaceTime, FileVault, Finder, FireWire, FireWire logo, iCal, iChat, iLife, iMac, iMessage, iPad, iPad Air, iPad Mini, iPhone, iPhoto, iPod, iPod classic, iPod shuffle, iPod nano, iPod touch, iTunes, iTunes logo, iWork, Keychain, Keynote, Mac, Mac Logo, MacBook, MacBook Air, MacBook Pro, Macintosh, Mac OS, Mac Pro, Numbers, OS X, Pages, Passbook, Retina, Safari, Siri, Spaces, Spotlight, There's an app for that, Time Capsule, Time Machine, Touch ID, Xcode, Xserve, App Store, and iCloud are registered trademarks of Apple Inc.

Governing Law: This Agreement shall be governed by the laws of the State of Maryland, USA.



SANS

SECURITY 401

SANS Security Essentials

© 2017 Secure Anchor Consulting
All Rights Reserved
Version C02_02

This page intentionally left blank.

SANS

Day 4

Cryptography and Risk Management

© 2017 Secure Anchor Consulting
All Rights Reserved

This page intentionally left blank.

SEC401 Day 4

- Cryptography
 - *Lab – Stego*
 - Cryptography Algorithms and Deployment
 - Applying Cryptography
 - *Lab – GPG*
- Incident Handling and Response
 - *Lab - Hashing*
 - Contingency Planning – BCP/DRP
 - IT Risk Management

Outline for Day 4 of SEC401.

SANS

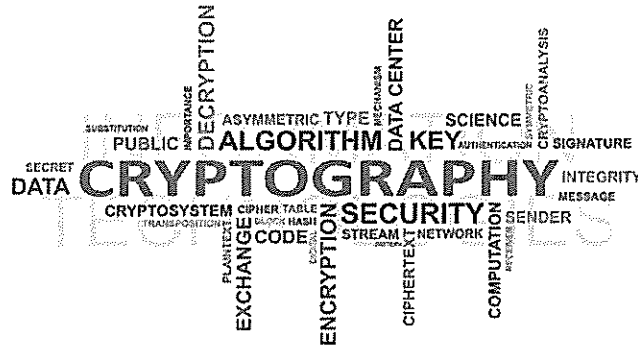
Module 18: Cryptography

Module 18: Encryption Overview

This page intentionally left blank.

Objectives

- Cryptosystem fundamentals
- General types of cryptosystems
 - Symmetric
 - Asymmetric
 - Hash



Objectives

We begin with some examples that illustrate the importance of sound cryptographic practices. We then closely look at effective ways of deploying cryptography, some of the common mistakes that are made and how to avoid them. Finally, we dive into the technical material with a discussion of how it all works, building a foundation for the cryptosystems covered in the next module.

Reference

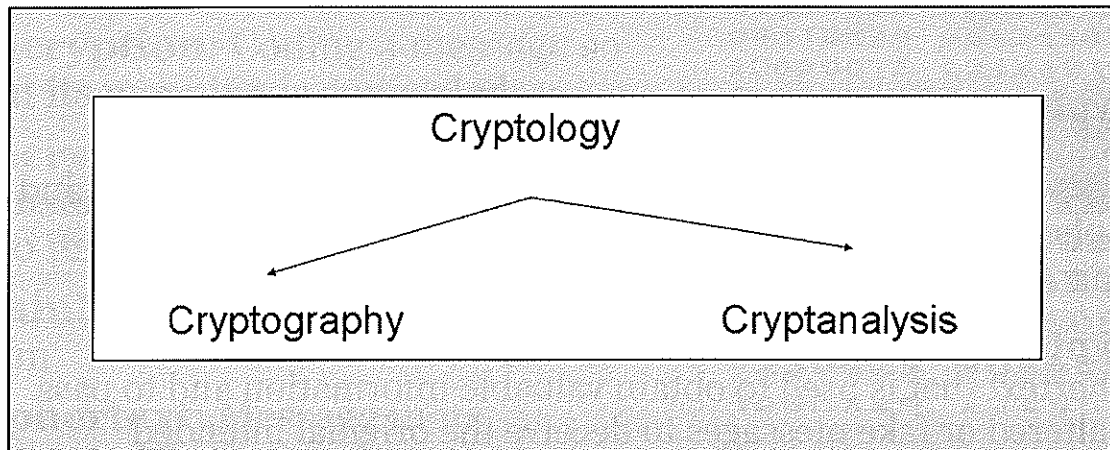
1. Sample Encryption and Decryption, <http://www.ictassociatie.com/themablog-bitcoin-blokchain-and-cryptocurrency/>

Crypto Fundamentals

The student will have a basic understanding of the fundamental concepts of cryptography

Cryptography, the science of secret writing, helps us communicate without revealing the meaning of information to adversaries. It also potentially validates to whom we are communicating. It can protect any kind of data, from very sensitive information, such as Internet-based commerce and banking transactions, to harmless messages you would prefer that no one else knew about, such as a letter to a friend. Cryptography (abbreviated as crypto) can provide a great deal of confidentiality and integrity checks for information. However, it is not a silver bullet, and it can lead to a tremendous false sense of security unless used properly and implemented correctly. Cryptography should always be a part of a larger defense-in-depth strategy, providing just one layer of the security onion.

Cryptography and Cryptology: Overview



Who creates these encryption algorithms? Computer scientists called “cryptographers,” who are well trained in several different fields of mathematics and who usually work in groups, take many years to invent and refine ciphers. But, with so much depending on cryptography, individuals called “cryptanalysts” dedicate their lives to breaking ciphers. Some cryptanalysts work for the military and governments; others are simply interested in the study of ciphers and want to find weaknesses in ciphers to ensure that they cannot be broken by others. The generic term for the study of both cryptography and cryptanalysis is called “cryptology.”

Cryptography and Cryptology: Key Terms

Cryptology: Encompasses cryptography and cryptanalysis

Cryptography: Art and science of hiding the meaning of a communication from unintended recipients; the word “cryptography” comes from the Greek words *kryptos* (hidden) and *graphein* (to write)

Cryptanalysis: Act of obtaining the plaintext or key from ciphertext that is used to obtain valuable information and to pass on altered or fake messages to deceive the original intended recipient

The following key terms relate to cryptography and cryptology:

Block cipher: Obtained by segregating plaintext into blocks of *n* characters or bits and applying the identical encryption algorithm and key to each block.

Cipher: A cryptographic transformation that operates on characters or bits.

Ciphertext or cryptogram: An unintelligible message.

Clustering: Situation in which a plaintext message generates identical ciphertext messages using the same transformation algorithm, but with different crypto variables or keys.

Codes: A cryptographic transformation that operates at the level of words or phrases.

Cryptanalysis: Act of obtaining the plaintext or key from ciphertext that is used to obtain valuable information and to pass on altered or fake messages to deceive the original intended recipient.

Cryptographic algorithm: A step-by-step procedure used to encipher plaintext and decipher ciphertext.

Plaintext: A message in cleartext, readable form.

Core Components of Cryptography

Cryptography: "Hidden writing"

Encryption: Coding a message so that its meaning is concealed

Decryption: Process of transforming an encrypted message into its original form

Plaintext: Message in its original form

Ciphertext: Message in its encrypted form



00312E30	00424301
1C020076	024E4E4F
21B2C809	8833P0CC
2B3EE8EF	DF0
14143B75	4FF
57C659E	820EE07
D7	9A36DD29
1	9A54E072
34	8986092
F130429	90A60B9
18	4A57266

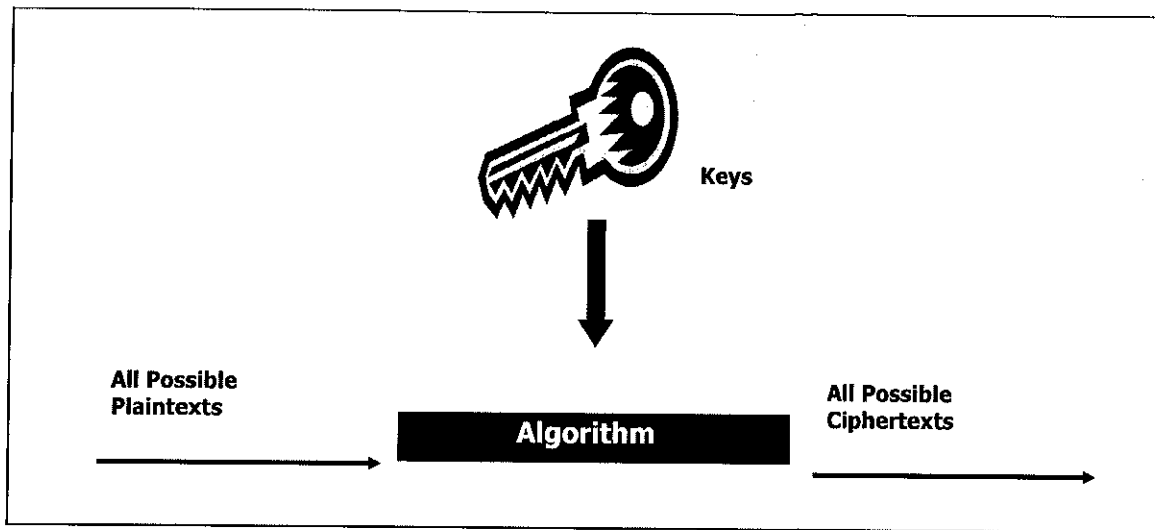
Cryptography is vitally important to information security. One of the main goals of cryptography is to protect information from unauthorized disclosure. The idea is that communicating over any kind of medium has the inherent risk that an unauthorized third party could be listening, and we want to minimize or eliminate that risk. So, in its most basic form, cryptography garbles text in such a way that anyone who intercepts the message cannot understand it.

Nearly every cryptographic algorithm performs two distinct operations: encryption and decryption. Encryption is the practice of coding a message in such a way that its meaning is concealed. How the message is transformed depends on a mathematical formula called an encryption algorithm or a cipher. After a message is transformed with a cipher, the resulting message is called ciphertext. Because ciphertext contains the message in its encrypted form and not its native form, it is unintelligible (has no meaning). For the recipient of the ciphertext to read the message, the recipient must decrypt it. Decryption is the process of transforming an encrypted message back into its original plaintext or cleartext form. Note that a plaintext message refers to any type of message in its unencrypted form. A plaintext message is not just an ASCII text message; an executable is also considered a plaintext message if it is not encrypted.

Reference

1. Encryption and Cryptography Standards, <https://webstore ansi.org/software/Encryption-Cryptography.aspx>

Cryptosystems



SANS

SEC401 | Security Essentials Bootcamp Style 10

A cryptosystem is the collection of all possible inputs and all possible outputs, in addition to the algorithm and keys. But, don't forget about the humans. Good cryptography is very strong and usually hard to break, but humans are still involved in managing and controlling the key. If you were an attacker, which would you attack? A strong algorithm or the users? Which is weaker? Never forget that humans are a critical aspect of a cryptosystem. They must be trained properly in using the system and protecting their keys. Lose the keys (by tricking a user into giving them up) and the entire security of the cryptosystem collapses.

We know what you are thinking: What are "all possible plaintexts?" Imagine any form of data or any kind of message you can think up. "I went to the store" is a valid inclusion into all possible plaintexts, just as "Bob was here," or an mp3 file. The resultant cryptographic transformations of all possible plaintexts are "all possible ciphertexts."

Keys

- Keys permit the existence of unrestricted algorithms
- Keys might be any one of a large number of values
- The strength of a cryptosystem rests with the strength of its keys
- Keyspace matters

40 bits of protection

```
0011101010010101010110101010101000110
```

128 bits of protection

```
0011101010010101010110101010101000110 0011101010010101010110101010101000110 0011101010010101010110101010101000110
```

128-bit keys offer approximately a trillion times more protection than 40-bit keys

Keys

Cryptographic keys are simply values used to initialize a particular algorithm. The important aspect of keys in regard to cryptosystems is that only the key, not the algorithm, needs to be protected. This means that algorithms might be widely distributed and their internal workings publicly documented. Only the key must be protected from thievery by communicating entities.

Keys are critical components of a cryptosystem. These keys are similar to a key to your house or safety-deposit box insofar as cryptographic keys provide access to protected resources, namely information intended to be kept secret.

The uniqueness of cryptographic keys is just as important as the keys themselves. Most airline travelers are probably familiar with the keys that come with luggage locks. These keys are far from unique. Purchase one set of lock and keys, and you have a fairly good chance of opening just about any randomly selected luggage lock you come across. Luggage-lock keys are a great example of a lack of uniqueness or of an "insufficient keyspace."

Keyspace is a critical concept concerning cryptographic keys. The larger the keyspace, the less likely an attacker is to discover a given key through brute force. A brute-force attack on a key involves trying every possible key until finding one that works. For example, the Caesar Cipher had a very small key space, which is trivial to exhaust or brute force.

Contrast luggage keys against car keys, for example. Car keys need to be more unique than luggage keys because of the importance and expense of automobiles.

Car keys can be said to have a larger keyspace and, therefore, offer more protection because the possible combinations of ridges and valleys on a car key is large. Because it is practically impossible for an attacker to guess the correct sequence of ridges and valleys that match your original car key (a brute-force attack), the attacker needs to either steal your keys or make an illicit copy to gain unauthorized access to your car.

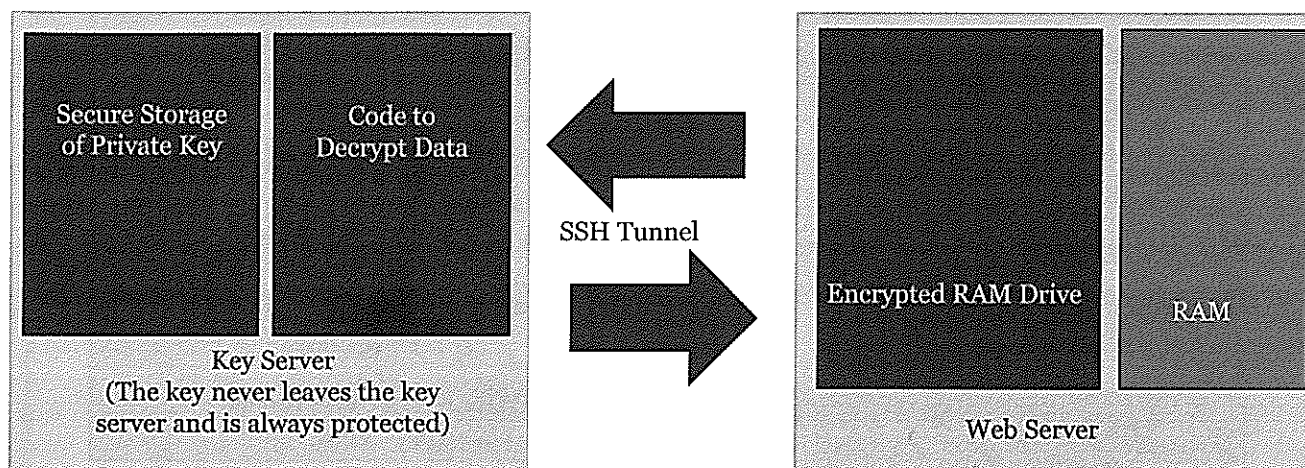
Conceptually, cryptographic keys should, at a minimum, provide the same level of uniqueness required for car keys, an abundant combination of ridges and valleys. In reality, of course, cryptographic keys need a much larger keyspace than that used for car keys. It should be impossible for an attacker to guess a cryptographic key that matches the one used to encrypt correspondence. Let's say the total number of possible unique car keys is approximately 200,000. Although that might not be accurate, even if the number is 10 million, the total number of possible unique cryptographic keys for a given cryptosystem needs to be a billion times larger simply to afford the keyspace protection against guessing an encryption key through brute force. Why? It's far easier to use a computer to iterate through a billion cryptographic keys than it is to physically recreate a million car keys. In short, a cryptographic keyspace must be absolutely enormous to afford sufficient protection.

What should this mean to you? Keyspace matters. The bigger, the better.

In the case of cryptographic keys, the length of the key correlates to the amount of protection the key provides. A 128-bit key offers about one trillion times more protection than a 40-bit key. Although it is obvious that a 128-bit key is longer than a 40-bit key, the difference in the amount of protection is exponential, not linear. Go ahead, prove it to yourself: 2^{40} (1.1×10^{12}) compared to 2^{128} (3.4×10^{38}).

Managing Keys Using Separate Key Servers

Protection of the private key on the web server



SANS

SEC401 | Security Essentials Bootcamp Style 13

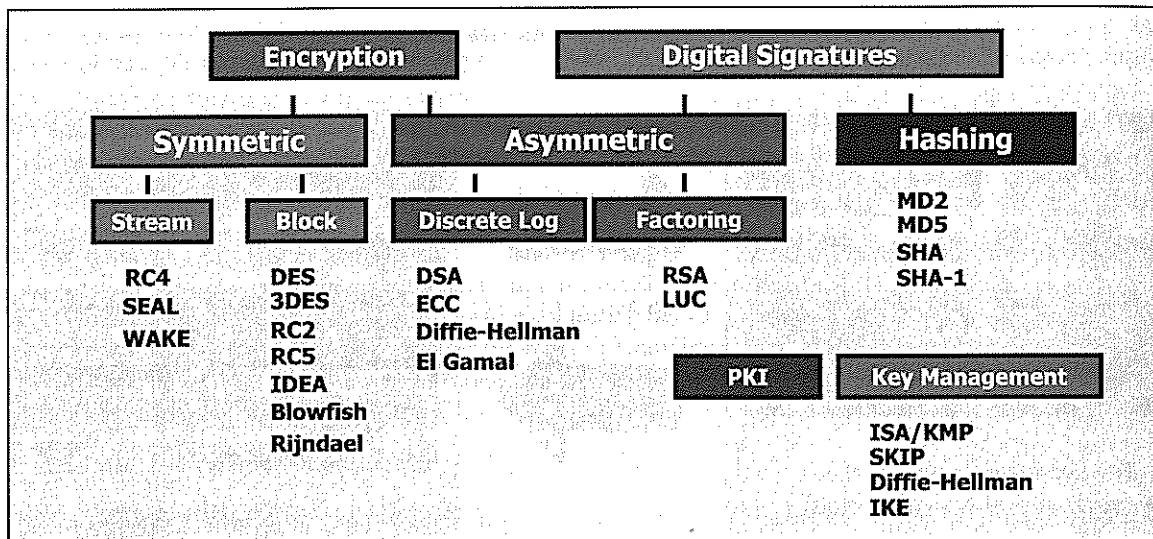
The uniqueness of a cryptographic key directly correlates to the amount of protection the key provides. In the previous discussion of car keys, the total number of combinations of ridges and valleys constitutes the size of the keyspace; the larger the number of combinations, the more likely the key is unique (making it more difficult for an attacker to guess a matching key).

The limitations of car keys are their length and their engraving depth; most are only 1.5 to 3 inches long, depending on the manufacturer. Keys can be engraved only so deeply before weakening the substance of which they are made (fragile keys are little use). These physical traits limit the number of keys that can be made, therefore limiting the total keyspace. If, however, car keys were 12 or even 20 inches long, the total keyspace would dramatically increase because they could support larger combinations of ridges and valleys. (Not too many consumers would want to carry around a 20-inch car key, however.)

Conceptually, the length of cryptographic keys is similar to the length of car keys; the longer the key, the more opportunity for uniqueness and the more difficult it is for an attacker to guess a corresponding key.

While keyspace matters, it is important to remember that the key must also be protected. You can have the largest key length possible, but if someone steals the key it becomes irrelevant. Therefore it is critical to remember that the keys must be protected at all times. This is done by always storing the key on a separate key server and the key never leaves the key server.

Enterprise Crypto: The Big Picture



SANS

SEC401 | Security Essentials Bootcamp Style 14

Some people tend to feel dizzy when we talk about cryptography. Not only is the mathematics of cryptography fairly esoteric and convoluted, but there are also many different kinds of cryptographic systems. We discuss each of these types of crypto in more detail in this course, but this slide provides an overview.

Symmetric stream ciphers are fast, and asymmetric factoring algorithms are slow. Diffie-Hellman is great for secure key exchanges, but not necessarily optimal for encryption. What does this all mean? Do not fret, we explain. For now, you need to know that the use of cryptography in the enterprise is a multifaceted endeavor. Different types of cryptography are used for different types of situations and, often, cryptographic systems are employed in concert.

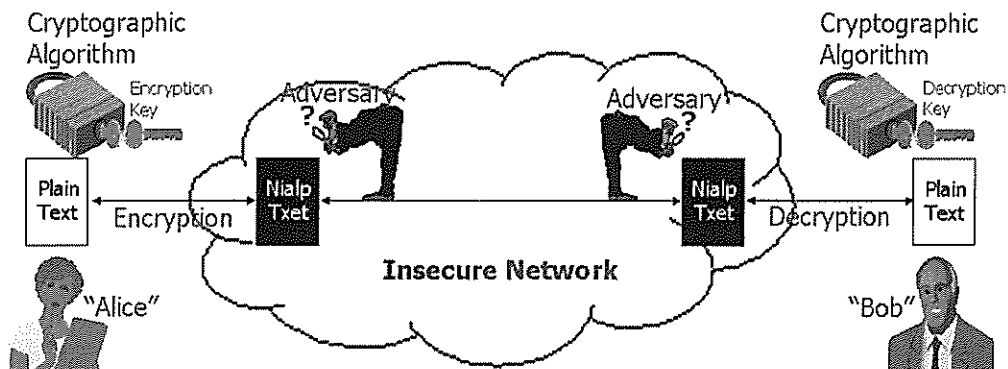
For example, to encrypt a message, we might choose either symmetric algorithms, such as RC4 or Blowfish, or asymmetric algorithms, such as RSA or ECC, but not any of the hashing algorithms, such as SHA.

However, to digitally sign a message (that is, give some type of "digital proof" as to the signer's identity), we might choose RSA or ECC with a hashing algorithm, but not any of the symmetric algorithms.

Finally, if we need high-speed encryption with the advantage of digital signatures, we might choose Diffie-Hellman to exchange a symmetric key, hash our message using SHA, digitally sign the hash using RSA, and encrypt the message and hash for transmission using Rijndael.

As we can see, cryptography in the enterprise promises to be a challenging topic. After completing this module, readers have a rudimentary understanding of these topics.

The Challenge



Communications in the presence of adversaries...

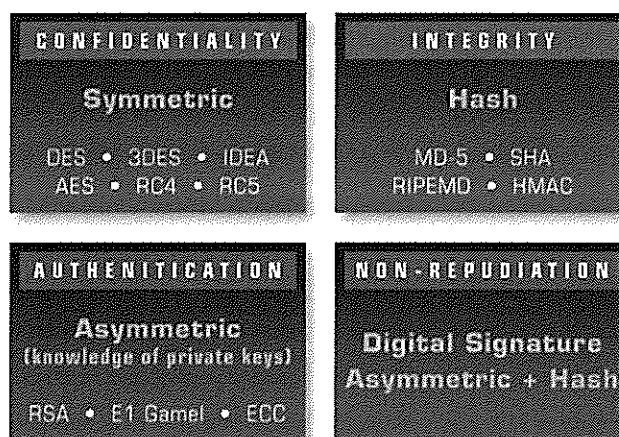
Confidentiality ★ Integrity ★ Authentication ★ Non-repudiation

The slide's image portrays the challenge of communicating over an insecure network. Alice and Bob want to exchange information securely. Their cipher is built on basic transformations, permutations, and substitutions. The result of the cipher is that the message is transformed so that, without knowledge of the key used in the system, the message is unreadable. Remember that even if someone knows how the algorithm works, without the key, he should still be unable to decipher the message.

There are three core components that we must address in implementing cryptography. First, we must make sure that the information is protected at rest. Second, we must make sure that the information is protected at transit. Finally, we must make sure that the keys are properly protected and managed. If any one of these three areas is neglected, the effectiveness of our crypto deployment is degraded and we are leaving the door open for the adversary.

Goals of Cryptography

The goals of a cryptosystem are

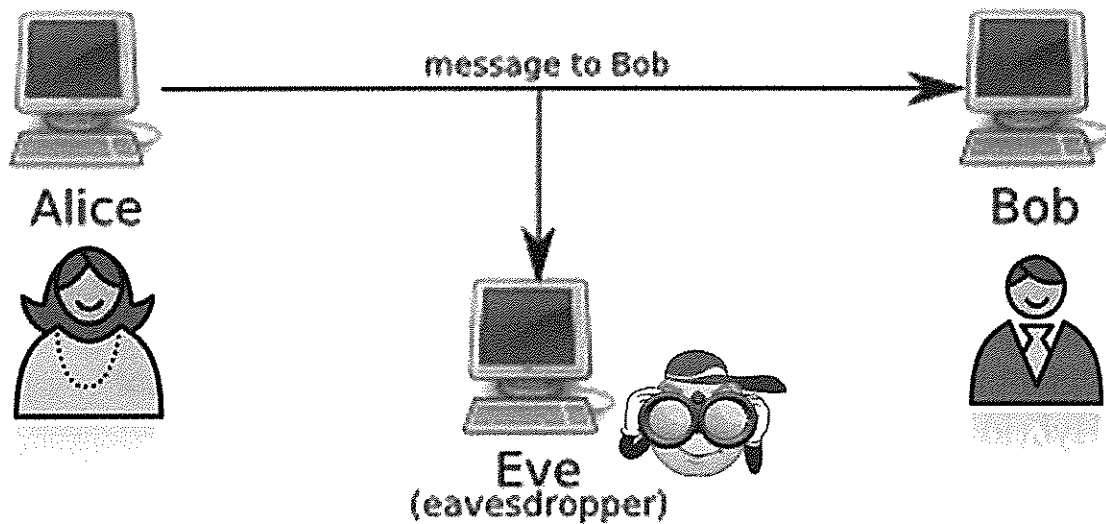


Like many of us, Alice does not care how the cryptography works, as long as it works. She needs to send a message to Bob with the same level of integrity she would have if she walked up and handed it to him. In addition to being unreadable by adversaries (confidentiality), we might have the following requirements:

- **Authentication:** If Alice walks up to Bob and hands him a message, he positively knows the message is from Alice; Alice might require the cryptosystem to provide an equivalent service for her, validating the authenticity of the person with which she is communicating
- **Integrity:** It should be possible to prove the message has not been tampered with—that this message is exactly the same as the one Alice sent to Bob
- **Non-repudiation:** The system should provide validation so someone is able to prove in a court of law that Alice, and only Alice, sent the message

The technology to do this is available, but for this system to work in practice, the non-technical issues are also important. Alice and all users of the system must be trained in its use and its limitations and have access to the keys, yet keep them protected and current. Processes must be as foolproof as practical. Think about social engineering, human error, and operator efficiency, accuracy, and understanding.

Meet the Player



In this module, we followed the convention of assigning human names to the participants in secure communications. We give the names "Alice" and "Bob" to two communicating parties.

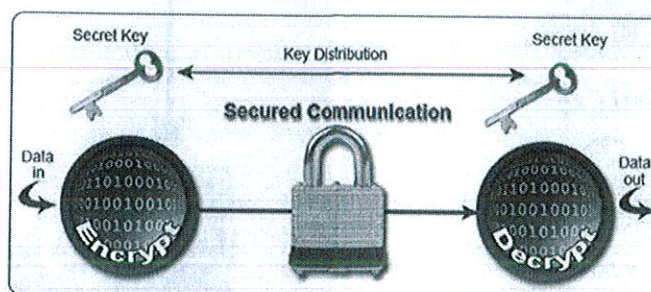
It is also common practice to use the name "Eve" as the person who is trying to break the encryption or read Alice and Bob's message. Although these names personalize our situations involving crypto, we need to remember that they are just metaphors. Although we might say, "Alice decides to use crypto algorithm X," keep in mind that users of crypto rarely make these kinds of deliberate, conscious choices. Alice probably bought some crypto product that selects a cipher from a set of available ones. The point is that users are generally not encumbered with the details of the cryptography.

General Symmetric Encryption Techniques

The goal is to garble the original message so that its meaning is concealed

Basic techniques

- Substitution
 - XOR
 - Rotation
 - Arbitrary substitution
- Permutation
- Hybrid



These techniques are used by symmetric key systems

Now we turn our attention from Alice and Bob to bits and bytes. Cryptography is a mathematical specialty that includes aspects of probability theory, information theory, complexity theory, number theory, abstract algebra, and more. Our discussion of crypto, however, does not require delving into these fields. Nevertheless, a few mathematical operations are necessary to understand our subsequent discussion, namely the OR, exclusive OR (XOR), and modulo functions.

The main goal of encryption is to garble text so that a third party cannot understand it. Two basic methods of encrypting or garbling text are substitution and permutation. A third approach is actually a hybrid, which is a mixture of both.

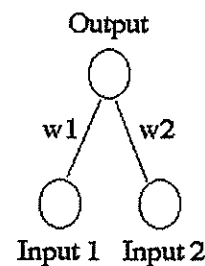
Reference

1. Cryptography and Encryption Background, http://www.nucrypt.net/overview_encryption.html

XOR

The Boolean Exclusive OR (XOR) function is one of the fundamental operations used in cryptography. The output of an XOR is TRUE if exactly one of the inputs is TRUE; otherwise, the output is FALSE.

Input 1	Input 2	Output
1	1	0
1	0	1
0	1	1
0	0	0



George Boole, a mathematician in the late 1800s, invented a form of logic algebra that provides the basis for electronic computers and microprocessor chips. His logical operations were a set of truth tables in which each of the inputs and outputs was either TRUE or FALSE.

The Boolean Exclusive OR (XOR) function is one of the fundamental operations used in cryptography. The output of an XOR is TRUE if exactly one of the inputs is TRUE; otherwise, the output is FALSE.

Computations require numbers, so we use 0 and 1 instead of TRUE and FALSE. The output of an XOR operation is 0 if both inputs are the same, and the output is a 1 if the two inputs differ.

These properties of XOR make it useful to cryptographers for two reasons. First, any value XORed with itself is 0 ($0 + 0 = 0$, $1 + 1 = 0$). Second, any value XORed with 0 is just itself ($0 + 0 = 0$, $1 + 0 = 1$).

Rotation Substitution

- It uses a one-to-one substitution of characters
- It "rotates" the alphabet by X characters, where x is the key

Easy to remember; for example

- Plaintext: A B C D E
 - Ciphertext: D E F G H
 - So "CAB" becomes "FDE"
- Caesar Cipher was ROT-3
 - Usenet uses ROT-13 (symmetric)

Symmetric encryption is based on simple mathematics that utilizes a single key. Whatever can be encrypted with one key can be decrypted only with the same key. Arbitrary substitution requires a mapping for every character in the alphabet. An alternate substitution method that does not require mapping is rotation. In this type of substitution, we shift every character a set number of spaces. For example, if we shift A three spaces, it becomes D, B becomes E, and so on.

The Caesar Cipher, invented by Julius Caesar to encode messages to his generals, is a famous rotation cipher. If Alice were using this "ROT-3" scheme, she would encrypt her message as "FDE." In its day (roughly 50 to 60 B.C.), the Caesar Cipher was considered good enough to fool almost anyone because few people could read, even fewer could write, and couriers would rather kill a snooper than let him capture a message. Caesar was no fool, however; he did not use just one encryption tool. He also transliterated Latin into Greek and used other forms of subterfuge.

Though many people believe the Caesar cipher is the earliest cipher, cryptography actually goes back nearly 2,000 years earlier (to ancient Egypt and China).

Although character rotation is a trivial scheme, rotation ciphers came back into vogue with newsgroups, primarily in the form of ROT-13. Shortly after USENET newsgroups and electronic mailing lists became popular, subscribers realized they did not always want to see the contents of a message.

Some messages contained jokes that might offend some subscribers. Other messages might contain riddles or puzzles complete with answers that the recipients might not have wanted to see before reading the riddle or puzzle.

The answer was to encrypt (or obscure) jokes and answers using ROT-13. ROT-13 was never meant to be a strong cipher—it is trivial to break. The point was for the reader to make a deliberate effort to decipher the message. No one could later claim accidental discovery, nor could anyone ruin a puzzle by accidentally

)
)
)
)
)
)
)
glimpsing at the solution. ROT-13 eventually became part of newsreader software and a common function of the UNIX operating system. ROT-13 had another nice feature. Because there are 26 letters in the English alphabet, ROT-13 is a symmetric operation; the same implementation both encodes plaintext and decodes ciphertext. This is because performing ROT-13 followed by ROT-13 is actually ROT-26, which takes you back to the original letter you started with.

Also note that, with rotation, if you figure out the mapping for one character, then you've discovered the entire key. Another flaw with substitution encryption is its predictability. If you use only one set of substitution rules, the encrypted message is easy to crack. Cryptographers responded by inventing more complicated substitution schemes.

Arbitrary Substitution

- It uses a one-to-one substitution of arbitrary characters
- Given one character mapping, you cannot determine the key, as with rotation substitution

For example:

Plaintext:	A	B	C	D	E
Ciphertext:	W	K	M	P	D

So "CAB" becomes "MWK"

It is easy to break using character frequency analysis

SANS

SEC401 | Security Essentials Bootcamp Style 22

Substitution involves exchanging one character (or byte) for another. Simple substitution schemes use mapping; therefore, one character is substituted with another character to encrypt a message, with decryption being the inverse action. The mapping function is the key; that is, anyone who knows how the characters were mapped to encrypt the message can decrypt the message.

Consider a simple example. Suppose we define the following map (only a portion of the alphabet is shown):

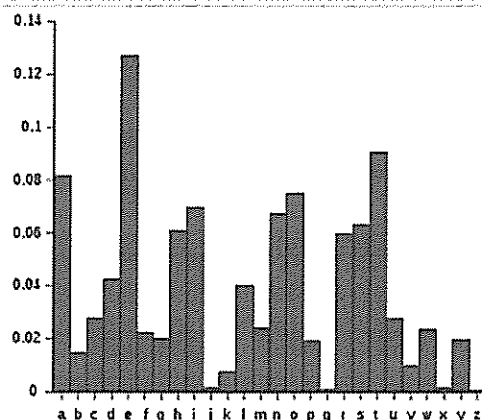
Plaintext: A B C D E...
Ciphertext: W K M P D...

To encrypt the word "CAB," Alice would substitute characters and send the string "MWK." In turn, Bob would reverse the substitution to recover the plaintext.

For substitution to work, there must be a unique one-to-one mapping from plaintext character to ciphertext character. A many-to-one or one-to-many mapping would make decryption difficult or impossible. For example, if W replaced both A and C, you would still be able to encrypt the message; therefore, CAB would become WWK. But if we tried to decrypt it now, we would not know whether the W should be an A or a C because they are both mapped to the same letter.

Frequency Analysis

Monolithic ciphers or one to one substitution ciphers can often be broken with frequency analysis



A typical distribution of letters in English language text. Weak ciphers do not sufficiently mask the distribution, and this might be exploited by a cryptanalyst to read the message.

One-to-one character substitution is weak because it can be defeated with frequency analysis. Long ago, cryptanalysts made tables showing the relative frequency with which letters, letter pairs (bigrams), and letter triples (trigrams) appear in a variety of languages. In all character-based languages, some letters occur with a greater frequency than others. In the English language, the letter E occurs approximately 13% of the time, and the letter T occurs approximately 9.3% of the time. So, by looking at the enciphered message, we can see which letter appears more often than most, and assume that the enciphered letter is an E. The next most frequently occurring letter would probably be a T, and so on. By looking at letter pairs (instead of just single letters), we can achieve an even more accurate guess.

Reference

1. Frequency Analysis, https://en.wikipedia.org/wiki/Frequency_analysis

Permutation

- Keeps the same letters, but changes the position within the text
- Changes the order from xyz to zxy

For example

- Change 1 2 3 4 5 to 3 5 2 1 4
- So ORDER becomes DRROE
- Very easy to break
- Substitution and permutation can be combined together

Permutation, also called transposition, shuffles the order in which characters (or bytes) appear rather than substituting one for another. The letters in the ciphertext are the same as the plaintext, they are just in a different order. Consider this simple example: Suppose that Alice and Bob chose the word "ORDER" as their plaintext message. The letter O is in the 1 position, R is in the 2 position, D is in the 3 position, E is in the 4 position and R is in the 5 position. If we use the key 3 5 2 1 4, the ciphertext becomes DRROE.

Unfortunately, permutation is also relatively easy to break. Remember that a few thousand or million combinations are nothing for a computer; it can defeat an adversary using pencil and paper. Today's computer-based methods still use substitution and permutation, but in combination applied many times.

General Types of Cryptosystems

The student will have a high-level understanding of the major types of cryptosystems

General Types of Cryptosystems

This page intentionally left blank.

Types of Cryptosystems

3 general types of crypto algorithms:

Symmetric

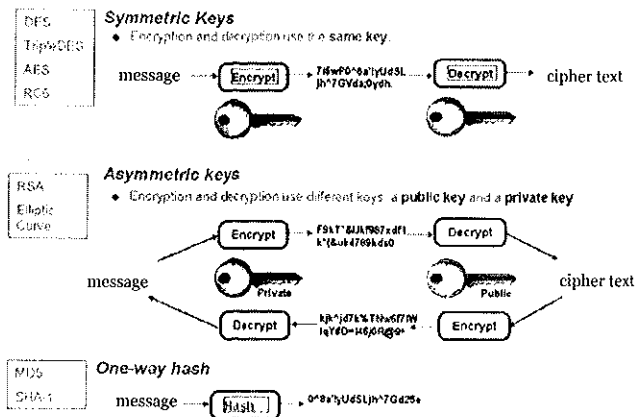
- Secret key
- Single or 1-key encryption

Asymmetric

- Public key
- Dual or 2-key encryption

Hash

- One-way transformation



Today, there are three general types of crypto algorithms: secret key or symmetric, public key or asymmetric, and hash. Each is used because it provides a different function from other cryptosystems. These schemes are usually distinguished from one another by the number of keys employed. The remainder of this module discusses these different types of algorithms.

Reference

1. Importance of Hashing Algorithms and Why should we Ask Ourselves if Cryptography and Encryption are Safe from the latest development of Quantum Computers, <https://steemit.com/science/@steemitguide/importance-of-hashing-algorithms-and-why-should-we-should-ask-ourselves-if-cryptography-and-encryption-are-safe-from-the-latest>

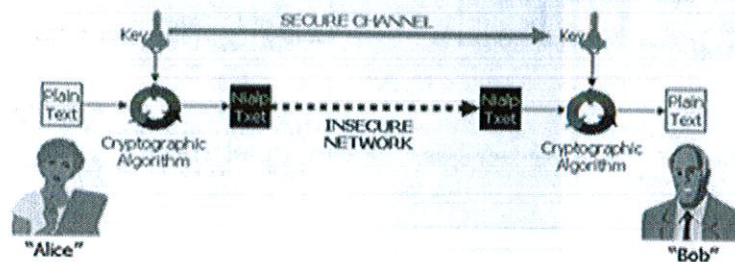
Symmetric Key Cryptosystems

"Secret key" encryption

- Fast! Single key for encryption and decryption
- Requires secure key distribution channel (scalability)
- No technical non-repudiation

Requires a secure channel

- Pre-shared key
- Asymmetric encryption
- Diffie-Hellman key exchange



SANS

SEC401 | Security Essentials Bootcamp Style 27

Symmetric key cryptography uses a single key for both encryption and decryption; this key is the shared secret between sender and receiver. Because symmetric key encryption uses only one key for both encryption and decryption, the key must be kept secret and is referred to as secret key encryption. The primary application of symmetric encryption is privacy, where only the parties with the key can encrypt and decrypt messages for each other.

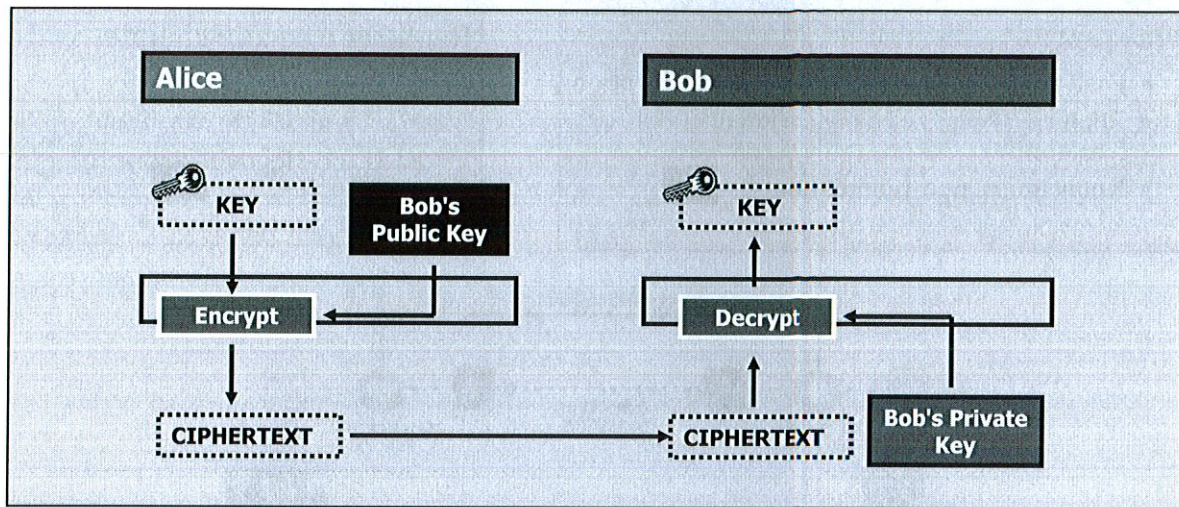
The big issue with secret keys is managing the key creation and exchange to avoid key compromise. Also, the greater the number of parties that share the secret key, the greater the exposure of the key.

The bottom line is this: Because symmetric key cryptosystems are so much faster than asymmetric key systems but lack the latter's key management and digital signatures, the two are often combined to achieve the best of both worlds.

There are a number of symmetric encryption schemes in common use today, all believed to be mathematically strong. If a cryptanalyst cannot defeat the ciphers by finding a weakness in the mathematical algorithms, then the remaining approach is a brute-force attack to guess all possible keys. Key size does matter, as explained in a paper by Matt Blaze, Whitfield Diffie, Ron Rivest, Bruce Schneier, and others in the cryptographic community. The paper, *Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security* (<http://www.counterpane.com/keylength.html>), describes brute-force attacks that are within the cost and computing means of a variety of attackers, and the key lengths necessary to keep such attackers at bay.

Examples of symmetric encryption schemes in common use today are the Advanced Encryption Standard (AES), Blowfish, and the International Data Encryption Algorithm (IDEA).

Secure Channel for the Secret Key Using: Asymmetric Key Exchange



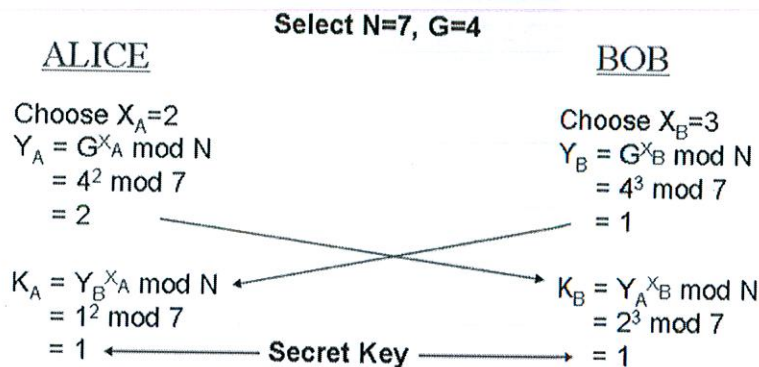
Using public-key technology to encrypt messages is rather expensive in terms of computational resources. In basic terms, asymmetric is really slow and can take a lot of time to encrypt a message depending on the message's size.

If you are going to encrypt a single message that is relatively small, slow performance might not be a concern; however, if you are going to frequently encrypt many large messages, it might be best to use public-key technology to exchange a symmetric key and use a symmetric algorithm to encrypt all those messages. Why? We discussed previously that symmetric algorithms are much faster than asymmetric algorithms, so when encrypting many large messages frequently, symmetric algorithms make more sense performance wise.

For example, to encrypt a 128-bit symmetric key using an asymmetric algorithm takes considerable less time than to encrypt a message that is perhaps 167 kilobytes, or 1,336,000 bits. To put these sizes in context, the symmetric key is more than 10,000 times smaller than the message. Decidedly, performance wise, it is far better to use public-key technology to encrypt a relatively small item, such as a symmetric key, as opposed to encrypting a document.

Secure Channel for the Secret Key Using: Diffie-Hellman Key Exchange

Alice and Bob agree on the value of a large prime number, N and a generator, G . Each calculates a private key (X) and public key (Y). The secret key (K) is derived from X and the other person's Y .



SANS

SEC401 | Security Essentials Bootcamp Style 29

Diffie and Hellman first published the concept of two-key crypto in 1976, but some time later they developed the Diffie-Hellman asymmetric algorithm, which is referred to today as the "Diffie-Hellman" and is used only for key exchange. This method provides a mechanism whereby Alice and Bob can determine the same secret key, even on a network with someone observing all of their communications. Essentially, it allows two parties to exchange a secret key in the presence of an adversary over a nonsecure network.

Alice and Bob start by agreeing on a large prime number, N . They then choose a generator number, G , where $G < N$, and G also meets some other conditions. Alice and Bob then each follow these same steps:

1. Each chooses a large, random number, $X < N$. X is the private key.
2. Each calculates the value $Y = G^X \bmod N$. Y is the public key and is sent to the other party.
3. Each computes the secret key $K = Y'^X \bmod N$, where Y' is the other party's public key.

Note that each party's Y is openly shared, but X is kept secret; these are the public and private keys, respectively. For that matter, N and G might also be well known. This scheme works because the secret key values (K) that Alice and Bob compute independently are the same; namely, $K = G^{XX'} \bmod N$, where X is their own private key and X' is the other party's private key (derived from the value of Y'). Because both X values are private, an eavesdropper cannot discover K except by brute-force methods. If N is large enough, this cannot be accomplished in a reasonable amount of time.

The figure shows a Diffie-Hellman example where $N=7$ and $G=4$. As shown, Alice and Bob choose private key (X) values of 2 and 3, respectively, from which they calculate public key (Y) values of 2 and 1, respectively. After swapping their Y values, both independently compute a secret key (K) value of 1.

This scheme works because Alice and Bob are using the same computation to calculate the secret key, namely

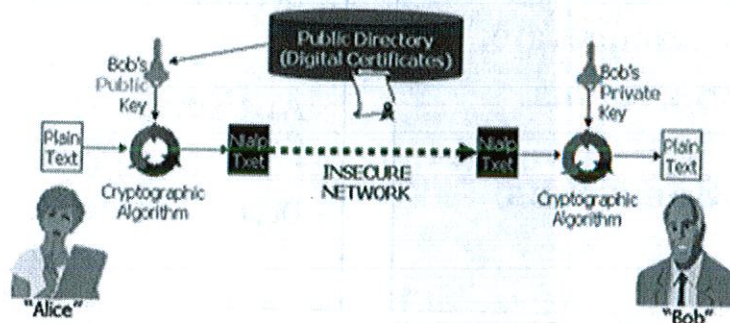
$$\begin{aligned}
 K &= Y'^X \bmod N \\
 &= (G^{X'} \bmod N)^X \bmod N \\
 &= G^{XX'} \bmod N
 \end{aligned}$$

Here, X and Y are their own private and public keys, and Y' is the other party's public key. In this example, Alice and Bob used $N=7$, $G=4$, and two private keys with the values 2 and 3; therefore, their shared secret key should be $4^{2 \cdot 3} \bmod 7 = 4096 \bmod 7 = 1$, which is exactly what they both calculated.

Asymmetric Key Cryptosystems

"Public key" encryption:

- Slow! Public/private key pair
- Public keys widely distributed within digital certificates
- Used as a secure channel for symmetric key exchange
- Technical non-repudiation via digital signatures



SANS

SEC401 | Security Essentials Bootcamp Style 31

The management problems associated with symmetric keys are so overwhelming that they virtually preclude their use by themselves in e-commerce. But, we can use public key computation to develop a shared message key. Also, algorithms like Diffie-Hellman can be used to exchange a secret key. Again, the general idea is to exchange keys securely, perhaps only once, to secure a given session, such as a visit to a web page to execute a credit-card transaction.

Public key cryptography or asymmetric encryption methods have two keys: one used for encryption and the other for decryption. From a mathematical standpoint, anything that is encrypted with one of the keys can be decrypted only with the other key. Asymmetric encryption has many applications, but the primary ones today are key exchange (for symmetric encryption), authentication, and non-repudiation.

Stanford University professor Martin Hellman and graduate student Whitfield Diffie first described modern asymmetric encryption publicly in 1976. Their paper described a two-key cryptosystem in which two parties could engage in a secure communication over a non-secure communications channel without sharing a secret key.

In the real world, how are these asymmetric key systems used? They are typically used to perform key exchange for symmetric key algorithms.

Bottom line: Despite being much slower than symmetric-key cryptosystems, asymmetric key systems are widely used because of their powerful key management and digital signatures—often in concert with symmetric key systems to attain the best of both worlds.

Asymmetric Encryption

The mathematics behind asymmetric encryption depends on the existence of so-called trapdoor functions

Two common examples are:

Multiplication versus factorization

Exponentiation versus logarithms

Use of Asymmetric Keys

Secure Channel

Encrypt with recipients public key

Decrypt with recipients private key

Authentication

Encrypt with senders private key

Decrypt with senders public key

The mathematical trick of asymmetric encryption depends on the existence of so-called trapdoor functions, or mathematical functions that are easy to calculate, whereas their inverse is difficult to calculate. Here are two simple examples:

- **Multiplication versus factorization:** Multiplication is easy; given the two numbers 9 and 16, it takes almost no time to calculate the product of 144. But, factoring is more difficult; it takes longer to find all of the pairs of integer factors of 144 and then to determine the correct pair that was actually used.
- **Exponentiation versus logarithms:** It is easy to calculate (for example, the number 3 to the 6th power to find the value 729). But, given the number 729, it is much more difficult to find the set of integer pairs, x and y , so that $\log_x y = 729$ and then, again, to determine that pair was actually used.

The previous examples are trivial, but they are examples of the concept; namely, the ease of multiplication and exponentiation versus the relative difficulty of factoring and calculating logarithms, respectively. Actual asymmetric encryption algorithms use integers that are prime and can be several hundred digits in length. Multiplying two 300-digit primes, for example, yields a 600-digit product; finding the two prime factors of a 600-digit number is beyond the capabilities of today's known methods. In this case, then, factoring is said to be intractable because of the difficulty of solving the problem in a timely fashion.

Keys are derived in pairs and are mathematically related, although knowledge of one key by a third party does not yield knowledge of the other key. One key is used to encrypt the plaintext, and the other key is used to decrypt the ciphertext; it does not matter which key is applied first, but both keys are required for the process to work.

One of the keys is designated as the public key and may be advertised as widely as the owner wants. The other key is designated as the private key and is never revealed. If Alice wants to send Bob a message, she merely encrypts the plaintext using Bob's public key; Bob decrypts the ciphertext using his private key.

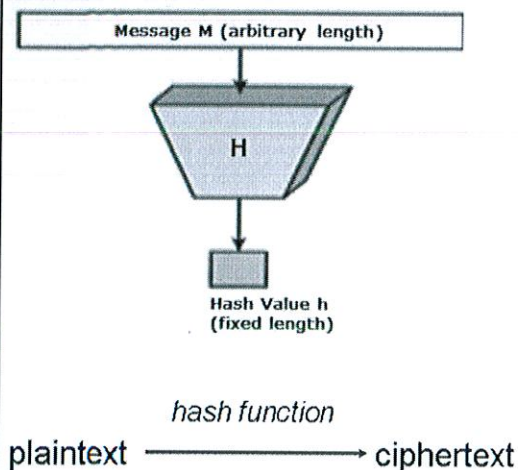
This two-key scheme can also be used to prove who sent a message. If Alice, for example, encrypts some plaintext with her private key, Bob (or anyone else) can decrypt the ciphertext using Alice's public key. The benefit here is that Bob (or whoever successfully decrypts the ciphertext) knows for sure that Alice encrypted the message (authentication), and Alice cannot subsequently deny having sent the message (non-repudiation).

Hash Functions

No key used during encryption

- Irreversible one-way transformation
- Key length is the hash length
- Plaintext (and length of plaintext) is not recoverable from the ciphertext
- Examples: MD2, MD4, MD5, RIPEMD-160, SHA-1, and SHA-2
 - Some algorithms have issues with predictable collisions
- Also called “message digests” or “one-way encryption”

Primary use: Message integrity



There are three types of cryptography algorithms: secret key, public key, and hash functions. Unlike secret key and public key algorithms, hash functions, also called message digests or one-way encryption, have no key used in the transformation. Instead, a fixed-length hash value is computed based on the plaintext that makes it impossible for either the contents or length of the plaintext to be recovered. The fixed-length output is what is often referred to as the key length of a hash function.

The primary application of hash functions in cryptography is message integrity. The hash value provides a digital fingerprint of a message's contents, which ensures that the message has not been altered by an intruder, virus, or by other means. Hash algorithms are effective because of the extremely low probability that two different plaintext messages yield the same hash value.

There are several well-known hash functions in use today:

- **Message Digest 2 (MD2):** Byte-oriented, produces a 128-bit hash value from an arbitrary-length message, designed for smart cards.
- **MD4:** Similar to MD2, designed specifically for fast processing in software.
- **MD5:** Similar to MD4 but slower because the data is manipulated more. Developed after potential weaknesses were reported in MD4.
- **Secure Hash Algorithm (SHA):** Modeled after MD4 and proposed by NIST for the Secure Hash Standard (SHS), produces a 160-bit hash value.

Reference

1. Cryptography Hash Functions,
https://www.tutorialspoint.com/cryptography/cryptography_hash_functions.htm

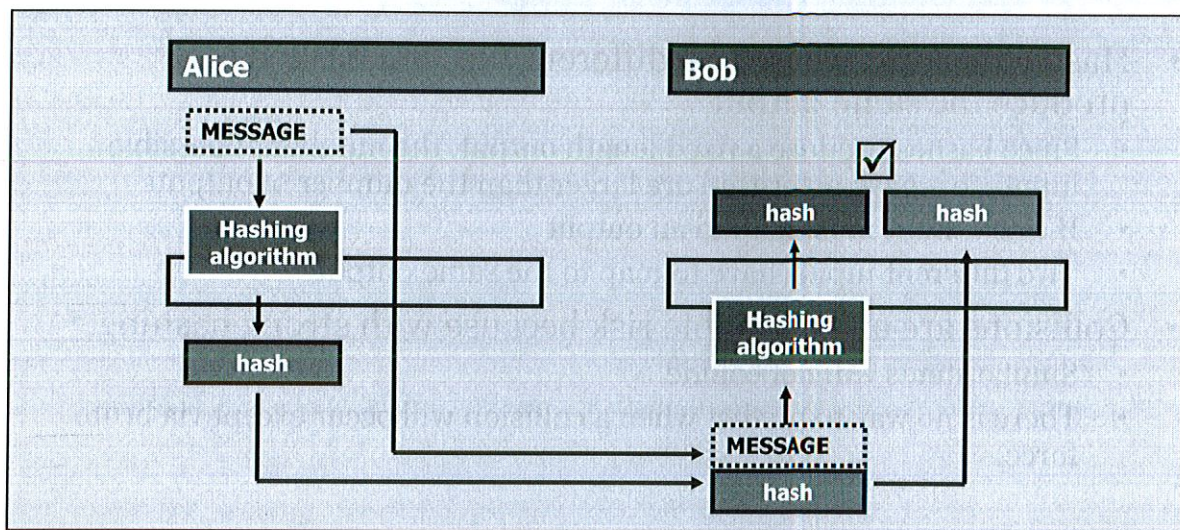
Collisions in Hashing

- Hash collisions: where two different files are hashed and produce the same output
 - Since hashes produce a fixed length output, the number of possible inputs to a hash algorithm are larger than the number of outputs
 - If every input must map to an output
 - Two different inputs have to map to the same output
- Collisions are an acceptable risk because with strong hashing:
 - Similar items will not collide
 - There is no way to predict when a collision will occur except via brute force

Hash collisions are where two different files are submitted through the same hashing algorithm and the same supposedly unique hash is generated. This is important because the purpose behind hashing a file is to create numeric representation of a file where the representation is unique every time. Hashing is one of the reasons why document signing is possible. Public key cryptography allows us to prove it was the recipient who created a document, but the hash proves the document has not changed since the recipient created it.

With hash collisions, it is possible to have two different files having very different content, end up with the same hash value when hashed. The reason this can occur is due to the types of transformations the hash algorithm is doing and the bit length of the hash function. The bit length makes the largest difference in the number of collisions possible. The larger the bit length the less likely there will be a collision because there are more bits to represent the data within a file.

Integrity



In the diagram, Alice intends to send a message to Bob. Alice's computer is represented by an orange box bearing Alice's name, whereas Bob's computer is represented by a green box bearing his name. In this scenario, Alice is concerned only with the message's integrity. Alice is not concerned with third-party eavesdropping.

Alice inputs the message into a hashing algorithm of her choice. The hashing algorithm outputs a hash of the message. Alice transmits both the original message and the fingerprint to Bob. Again, message integrity is Alice's primary concern, not message confidentiality. Finally, Alice informs Bob of the hashing algorithm she used. Informing Bob of the hashing algorithm is important so Bob can confirm message integrity.

Upon receiving the message and hash, Bob employs the same hashing algorithm as Alice. Bob inputs the message to the hashing algorithm, which generates a fingerprint. Bob then compares this hash against the received fingerprint. If both hashes are identical, Bob confirms Alice's message was unaltered in transit. If the hashes do not match, Bob knows something occurred during transmission.

Digital Signatures

- Digital signatures use public key cryptography to "sign" documents
- The signatures are nonrepudiable
- They "sign" a document by encrypting a one-way hash with a private key



SANS

SEC401 | Security Essentials Bootcamp Style 37

Semantically, digital signatures are equivalent to signatures affixed to documents with pen and ink: The signature is meant to identify the signer uniquely.

Because pen and ink are useless in an electronic environment, cryptography, specifically asymmetric algorithms, are used to provide the required uniqueness. Handwritten signatures have long held protected legal status as an official recognition of approval on a paper document (despite the fact that handwritten signatures are notoriously easy to forge). In this digital age, it seems only fair that we have a method of signing electronic documents that is unique as well as difficult to forge.

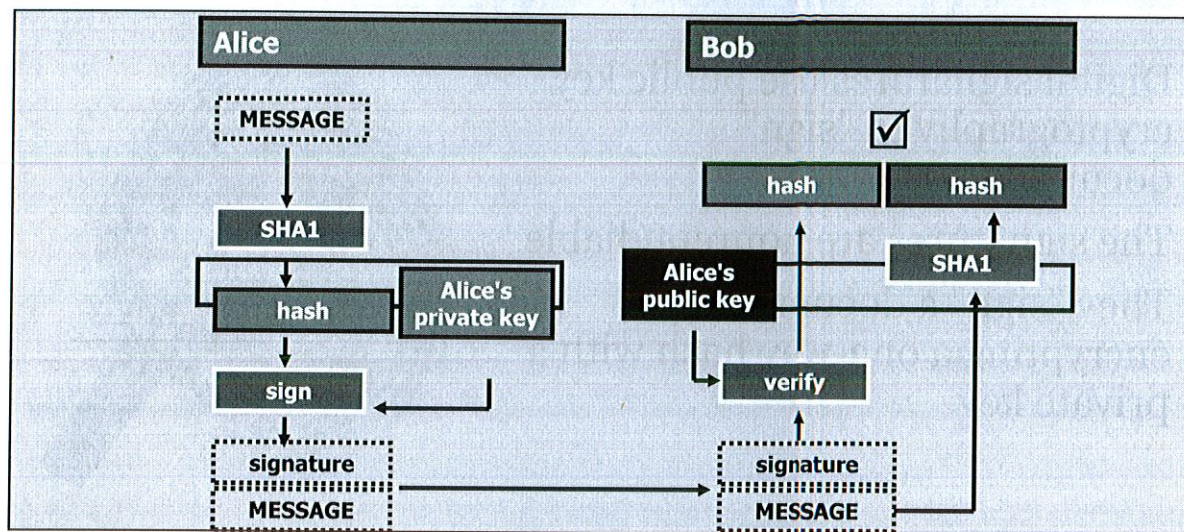
Public key cryptography allows users to employ their private key to encrypt data, in effect signing the data. Because a given private key is intended for one and only one owner, use of the private key in encryption unmistakably associates the user's identity with the encrypted data. This is semantically equivalent to the user hand-signing the data.

Recipients of the signed data employ the user's public key to decrypt and, therefore, verify the sender's signature. In short, by leveraging asymmetric algorithms, users can establish a person's digital signature is authentic. In addition, users can also establish authenticity even if the sender denies having signed the data. This is called non-repudiation.

Reference

1. Adding Digital Signature to Your PDF, <https://plugpdf.com/adding-digital-signatures-to-your-pdfs-on-android-and-ios/>

Digital Signature Example



In this illustration, Alice intends to send a message to Bob. An orange box bearing her name represents Alice's computer and a green box also bearing his name represents Bob. As in the hashing algorithm example, Alice creates a unique fingerprint of the message she intends to send to Bob. In this case, Alice chooses SHA-1 as the hashing algorithm. Alice also wants to digitally sign the document for two reasons:

- To protect the hash from modification during transmission
- To prove to Bob that she sent the message

To digitally sign the hash, Alice employs her private key to encrypt the hash. Because Alice is supposed to be the only owner of her private key, use of the private key binds her identity to the hash. Alice then sends both the plaintext message and the signature to Bob. As in the hashing algorithm example, Alice's concern is for the integrity of the message, not necessarily the message's confidentiality.

To mitigate an attacker who is intercepting, altering, and creating a new hash of her message, Alice signs the hash—that is, cryptographically safeguards it from modification.

Upon receiving the message and Alice's digital signature, Bob employs Alice's public key to decrypt the signature. By decrypting the signature, Bob retrieves the hash of the message generated by Alice.

Bob then generates a fingerprint of the received message and compares it to the received hash. If both hashes are identical, then Bob knows two things:

- The message did not change in transmission
- The message was sent by Alice

In addition, Alice cannot deny she sent the message because only her public key could decrypt the signature—assuming, of course, that no one stole her private key.

Steganography Overview

The student will understand what steganography is
and how it differs from cryptography

SANS

SEC401 | Security Essentials Bootcamp Style 39

Steganography Overview

This page intentionally left blank.

Steganography (Stego)

- Data hiding (steganography means "covered writing")
- Involves concealing the fact that you are sending "sensitive" information
- Dates to Ancient Greece, modern awareness relatively new
- Can hide in a variety of formats
 - Images (bmp, gif, jpg)
 - Word documents
 - Text documents
 - Machine-generated images (fractals)

Steganography (stego) is a means of hiding data in a carrier medium. Steganography means "covered writing." In concept, it dates back to ancient Greece. However, as a means of hiding data electronically, it is a new concept.

The modern form of stego can take many forms, although all involve hiding data in something else called a carrier file. This can be hiding a document in an image, hiding a short message in a document, and even hiding an image in a sound file! The applications are only limited by the tool being used, the carrier file, and the imagination of the sender.

Stego can be used for a variety of reasons but, most often, it is used to conceal the fact that sensitive information is being sent or stored. It can also be used to disguise encrypted data. This helps prevent attacks on encrypted data, or in scenarios where encrypted data is inappropriate for transmissions (for example, in countries where encryption is against the law).

Crypto Versus Stego

- Cryptography (crypto) provides confidentiality but not secrecy
- It is fairly easy to detect that someone is sending an encrypted message; it is just difficult for someone to read it
- With stego, you might not even know someone is sending a message; the true intent is hidden

Cryptography (crypto) is a tool to protect confidentiality and integrity and provide non-repudiation for the senders of data. However, despite all these benefits, crypto does not guarantee the secrecy of your data. Scrambling the data into unintelligible ciphertext can prevent others from reading the file, but it does not keep them from realizing that the data is there. It is easy to detect an encrypted message; it is difficult to read one.

One unwanted side effect of using encryption is that it can mark a user's most important and confidential files. It is similar to keeping valuable items in a bank vault, or an armored car. Encryption keeps the content safe, but when attackers are in hot pursuit, they know what to target for the valuables.

An encrypted conversation can also raise suspicions. If two parties suspected of a crime had suddenly started trading extensive encrypted messages the week before the crime occurred, even though we might not know what they were saying, it would definitely raise some flags and concerns. This is known as an inference attacks. You do not know all of the facts, but you can infer what is happening.

When handling extremely confidential data, it would be ideal to obfuscate the information and keep it as undetectable as possible. Secrecy keeps an attacker from even trying to subvert the encryption on these files. The person can see the image or sound files, yet he has no idea that they are also carriers of encrypted data.

Steganography Doesn't Guarantee Safety

One important thing to keep in mind when using stego is, that even though the secrecy provided by stego is great, the data's protection still relies on the encryption algorithm that is being used. Some stego programs use weak or untested encryption algorithms, or in some cases no encryption at all! Some stego tools have a choice of encryption methods of varying effectiveness that require you to choose between. Users are often duped into a false sense of security while using a stego tool. They think that if the data is hidden, it is safe. However, if stego is detected, the safety of your hidden message is only as good as the encryption that is used to protect it. If the confidentiality of your data is important to you, always verify the stego tool that you are using has a proven encryption algorithm. If it doesn't, or you are unsure, encrypt the data with a tool using a proven algorithm (such as PGP) before running it through the steganographic process.

Detecting Cryptography

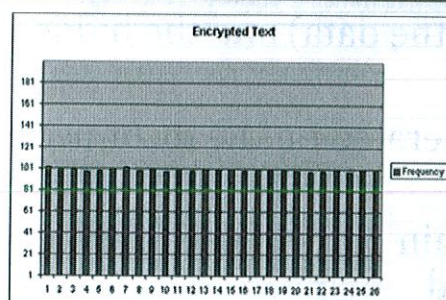
It is easy for both humans and computers to detect that a message is encrypted. For example, "test" becomes

```
eJrMIedoDcgYmK7/XwY6Q+7RAeuPDSeoFziMLDU1GyUhcoWPcat
AaIpw+UrcoMUXl257b1q11gFZN4SorXwAKg2Tzqn9ois7+1pJHO
dxI2fH9LCQmxtRBpZ79oFh+wFwcuPV3wW4Mgoh1HL2JQ7Sarr
JuZixgRoV+IW/HtoWx2Mvop+4CACHtTxbv8SjchhNFLaQNVQA
1o0oUgR+m7bJh42bWfR5cdGBYkVTzglbu5QXzFodk3PmtG+ghq
NCz2CZ5VZv3H581bSeydcM5zjK7DUD4OZEDSa9kF+9xKdyDMC
fvFW5DyhlJkOBUVo8jvQMn/3nO8vGcx/5CcDvv6MF4xh5hPbV6
NfP2OaOyNVXcHwn9n6/swH4OnrBciX8MCgFJCyXrwnlYl1GK7R
BO67zwoimUkBABfAqc+Jwnbv2HJAAUoNDC+Vd+d9I4UZN6QJd
7RN821ID1oScXelDNiqCq8hxXHJM8qaP5gQp5iC2ExoPfFPI8KRsb
OKcK5XPP57T
```

A human can infer that, because this is unreadable, it may be encrypted.

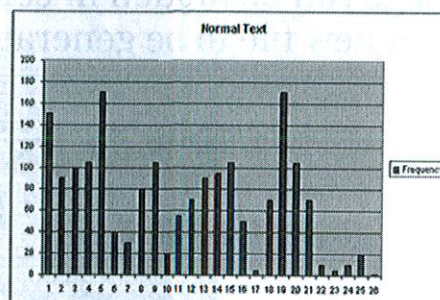
Both humans and computers can easily detect encryption because encryption increases file size and mathematically normalizes the occurrence of data. When viewing a document or e-mail comprised of garbled characters, one might infer that it is actually encrypted.

Histograms



The histogram for "normal" text is very non-uniform and easy for an automated program to distinguish between encrypted and unencrypted information.

The histogram for encrypted text is very flat and easy for an automated program to detect.



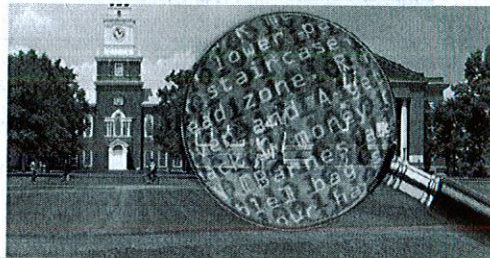
SANS

SEC401 | Security Essentials Bootcamp Style 43

Histograms are graphical representations of the number of occurrences of data in a given distribution of such data. For example, a histogram of a text document would show the number of occurrences of each character that appears in the document. A normal text document would generate a histogram that shows that the frequency of characters varies greatly. In a histogram for an encrypted document, the frequency of characters is normalized because good crypto always produces random ciphertext. The same factor that helps prevent encryption from being interpreted makes it easier to detect.

How Steganography Works

- Stego requires a host (to carry the data) and the hidden message
- Host (usually a file) can be generated on the fly or use existing data
- Message can be hidden in certain parts of existing file or can cause a new file to be generated



SANS

SEC401 | Security Essentials Bootcamp Style 44

The principle behind steganography is simple: hiding data within data. This can be done in many different ways. The only limiter is the steganographer's creativity. Despite the seemingly endless possibilities for stego, some commonalities can be found in its operation. There are several basic components that are common to all stego and several general types of operations that all stego can be categorized into. In the following sections, we explore these tenants of basic steganography.

Components of Stego

There are two general components of standard steganography. The first is the carrier or host file. This is the medium used to hold the hidden data. The carrier can be almost any type of file imaginable. Some popular examples of such hosts are:

- Images: bmp, gif, and jpeg
- Word documents
- Sound files
- Movies: mpeg
- Text documents
- Machine-generated images: fractals
- HTML files

Reference

1. What is Steganography? <http://smartechverse.blogspot.com/2015/07/steganography.html>

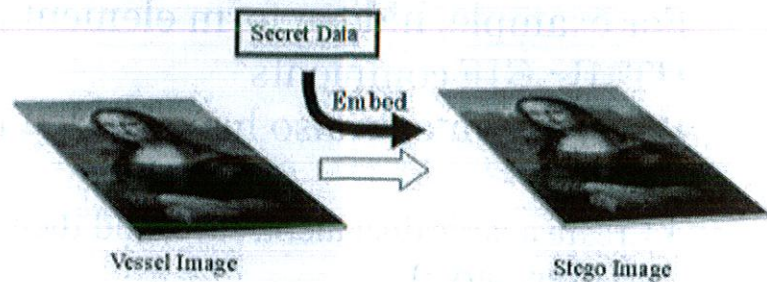
For the purposes of this section, we focus on steganography as it relates to files. Other non-file carriers are possible with stego including data streams and correlative messages.

General Types of Stego

There are many ways to hide information; lesson in creativity

- General methods

- Injection
- Substitution
- File generation



SANS

SEC401 | Security Essentials Bootcamp Style

45

The second component of standard steganography is the hidden message. Information can be hidden in many ways. In ancient times, information was placed on wooden tablets that were then covered with wax to hide the message. Messages were also tattooed onto messengers' bare heads. (Hair growth covered the message and their head needed to be shaved so the recipient could read it.) In more recent history, messages were written with invisible inks that appeared only after they were heated. Messages were written with these inks in the margins or between lines in false documents to hide the fact that a hidden message existed.

In the information age, there are many new creative ways to hide information in an electronic carrier. Most of the techniques can be summed up in one of three general stego types:

- Injection
- Substitution
- File generation

Reference

1. Steganography, <http://datahide.org/BPCSe/steg-scheme-e.html>

Injection

- Most file types have ways of including information that will be "ignored"
- For example, hidden form elements or comments in HTML; GIF comments
- Word documents also have hidden information—holes in the data
 - Create a large document, save, and then remove data (file size is still very large)
- Increases the size of the file but no theoretical limit on how much data can be hidden

With most file types, there are ways to include information within them that is ignored when the file is processed. This is the basis for injection stego. We place the information into "holes," or unused areas of the file. For example, with HTML, informational tags that tell how it should be processed must precede all characters. Web browsers ignore data that is formatted with certain HTML tags. However, if you examine the same HTML file with a text or HTML editor, the added characters are fully visible. Another example is the comments that can be inserted in files, such as those that can be placed in a GIF image or MP3 sound file. These comments do not appear when you view or play the file, although they still physically exist in the body of the file if you know what to look for.

Even Microsoft Word documents contain areas (or holes) where information can be hidden. This can be demonstrated by creating a large document, saving it, and then by "cutting" a large portion of the document out. Even after the data is removed, the file size is still very large. The slack that is left in the document can also have data inserted into it.

The greatest problem with the injection type of stego is that as data is added, the file size of the carrier increases. This makes detection easy if the original file can be found, or if the size is increased outside of the norm for its type. For example, if an MP3 file was injected into a document file, the increased size of the document will most likely be noticed.

When comparing the original document and the stego carrier with most text editors, it is impossible to tell the two apart visually. However, viewing the same document through a file-comparison utility or hex editor shows how different the files actually are.

Substitution

- Data in a file can be replaced or substituted with hidden text
- Depending on the type of file and/or the amount of data, it could result in degradation of the file
- It usually replaces insignificant data in the host file
- Since data is overwritten there are limits to how much data can be hidden

Substitution is the most popular stego method used to hide data in a host file. The concept is that elements are replaced on a bit-by-bit basis with information that is being hidden in the host document. Because the information is substituted in place of existing information, the file size of the carrier remains the same. However, noticeable file degradation can occur depending on the amount of information placed in the document. The goal with this technique is that only insignificant data should be overwritten to prevent degradation. It is important to have a suitably large carrier file when great amounts of information are being concealed. Typically, insignificant data is replaced with the information to be hidden. This insignificant data can take many forms, but one of the most common forms is the least significant bits (LSB) in the color table of a graphic.

Generate a New File

- The hidden data can also be used to generate a new file
- No host file is needed
- For example, the input text can be used to generate fractals or "human-like" text

Another method of stego that is growing in popularity is the actual generation of a new file from the data to be hidden. This is the only form of stego where a carrier isn't needed beforehand. A carrier file is needed, but it is generated on the fly by the stego program. The carrier file is actually created from the source information to be concealed. This can be used to generate such output as readable text or fractals. With each unique input file, a completely new and unique output file is generated.

Detecting Steganography

- Many ways for steganography to work
- Detection typically works on a tool by tool basis
- Common way with images is use of the least significant bit
- Tools to detect Stego
 - Stegexpose
 - StegSecret
- No universal way to detect steganography

In order to detect encrypted files, you can run a histogram on the files. If the histogram of the file shows that there is a high entropy, this indicates that it is an encrypted file. With a clear text file, it has a very staggered non-uniform histogram. Detecting steganography in files is a bit harder, but the modifications to files do give some indications that steganography is in use but the modifications are different depending on the tool/technique that is being used. Since each tool utilizes a different technique, there is no universal method to detect steganography.

Data can be hidden in just about any file type once you know how the file type is constructed. The most common way to utilize steganography is to hide data inside images. This is done by using the least significant bits (LSB) in the image to store the actual data. These bits are in use by the image to dictate the color of pixels, but the least significant bits are the bits which have the least impact on shifting the color represented on the screen for the image. As a result, these bits can be overwritten with data and it will cause nearly no recognizable change to the picture. Two file types where this is the case is the PNG and BMP image formats.

Knowing this is how some tools store data inside an image, one could write a program such as Stegexpose or StegSecret, to analyze pictures and look for deviations in the least significant bit. Normally, the least significant bits should be aligned, but the data hiding alters the properties, indicating that data has been hidden in the file.

Stegexpose and StegSecret allow for the detection of steganography within images. It is important for the detection of steganography in cases where malicious actors are attempting to communicate without being noticed, or in the event of data exfiltration, where data which appears to be unclassified leaving the network actually has sensitive data hidden within it.

Reference

1. StegSecret, <http://www.brianur.info/stegsecret-herramienta-para-estegoanalisis/>

Summary

Encryption plays a critical part in the protection of information

- Most systems that employ encryption use all three types of encryption
 - Symmetric
 - Asymmetric
 - Hash



Cryptography, the science of secret writing, is an essential component of computer and network security at all levels. Information security professionals must be comfortable with at least the basic terms and concepts associated with this field so that they can understand products, services, and vendor claims.

Although the crypto methods used today are vastly stronger and more complex than algorithms used even 30 years ago, the same two fundamental operations still form the basis of symmetric encryption schemes used to encrypt messages for privacy: substitution and permutation. Substitution is the method of replacing, or substituting, characters in a message with other characters, whereas permutation (transposition) moves characters around within the message. Today's algorithms tend to employ many rounds of both.

A crypto key governs the transformation of the plaintext into ciphertext. Modern crypto algorithms can be broadly classified into three categories based on the number of keys employed and the goals they accomplish. Each of these methods is used for specific applications. The categories are

- Symmetric encryption algorithms use a single key for both encryption and decryption.
- Asymmetric encryption algorithms use a pair of very large, mathematically related keys. Asymmetric encryption uses a two-key system, whereby one of the keys is used to encrypt data, and the other is used for decryption. This depends on the existence of so-called trapdoor functions that are easy to calculate, whereas the inverse function is difficult (intractable). With trapdoor functions, one key does not yield knowledge of the other key. One of the keys, therefore, can be widely distributed and is called the *public key*; the other key is kept secret and is called the *private key*.
- Hash functions are one-way encryption; they employ no key, and the hash operation cannot be reversed to recover the original plaintext from the hash value.

In today's environment, it is rare to find only one of these algorithms in use; it is far more common to find a set of these protocols used together to form a cryptosystem.

Although cryptography is necessary for security, it is not sufficient by itself. There are bad crypto schemes, bad implementations of good crypto schemes, and misuse of good implementations. Just as security is a process, so is the management and use of crypto; thus, security administrators—and users—need to be trained in the art of cryptography.

Reference

1. Current Trend in Cryptography, <https://mondayblogger.com/cryptography-current-trends-2016-2>

SANS

Lab 4.1 – Image Steganography

Stego tools allow us to both encrypt files for confidentiality and also hide the encrypted files in a host file for secrecy. For this lab, we use the tool “Image Steganography,” which was written by “cpascoe” and is maintained at <https://imagesteganography.codeplex.com/>. Many stego tools were written in the late 1990s and early 2000s. The last update to this tool was in 2011 with Release 1.5.2. Image Steganography allows you to hide text and files inside of PNG image files. It supports a couple of different methods for hiding or embedding data, including Difference, Enlarge, and Embed. Each of these methods makes different changes to the output host file in order to accommodate the hiding of data. A good list of steganography tools can be found at <http://www.securityfocus.com/tools/category/55>.

Lab 4.1 – Image Steganography

Purpose

- Learn how to utilize steganography programs
- Understand the operations of data hiding program

Duration

- 20 minutes

Objectives

- Introduction to Image Steganography and its Interface
- Hiding text with Image Steganography
- Hiding files with Image Steganography

SANS |

SEC401 | Security Essentials Bootcamp Style 53

Purpose

- Learn how to utilize steganography programs
- Understand the operations of data hiding program

Duration

- 20 minutes
- The estimated duration of this lab is based on the average amount of time required to make it through to the end. The duration estimate of this lab can decrease or increase depending on various factors, such as the booting of virtual machines, the speed and amount of RAM on your computer, and the time you take to read through and perform each step. All labs are repeatable both inside and outside of the classroom, and it is strongly recommended that you take the time to repeat the labs both for further learning and practice toward the GIAC Security Essentials Certification (GSEC).

Objectives

- Introduction to Image Steganography and its Interface
- Hiding text with Image Steganography
- Hiding files with Image Steganography

Lab 4.1 – Overview

You use your Windows 10 VM for this lab. Your objective is to use the Image Steganography tool to hide both text and data in a PNG file. You first hide text inside of a file and then extract the text. You then encrypt and hide a file inside of a PNG image, as well as extract the file.

You use your Windows 10 VM for this lab. Your objective is to use the Image Steganography tool to hide both text and data in a PNG file. You first hide text inside of a file and then extract the text. You then encrypt and hide a file inside of a PNG image, as well as extract the file.

SANS

**NOTE: Please open the
separate Lab Workbook
and turn to Lab 4.1**

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.

Lab 4.1 – Exercise Takeaways

In this lab, you completed the following tasks:

- ✓ Introduction to Image Steganography and its interface
- ✓ Hiding text with Image Steganography
- ✓ Hiding files with Image Steganography

In this lab, you completed the following tasks:

- ✓ Introduction to Image Steganography and its Interface
- ✓ Hiding text with Image Steganography
- ✓ Hiding files with Image Steganography

In this lab, you used the Image Steganography tool to encrypt and hide both text data and a PNG image inside of a host file. You also extracted and decrypted the embedded data out of the host file. As you can see, detecting that a file was steganographically altered is different without at least having the original file along with the altered file for comparison. Each mode uses different techniques to hide the data, such as the embed option, which hides data in the smallest bits of each pixel.

SANS

Lab 4.1 is now complete

This page intentionally left blank.

SANS

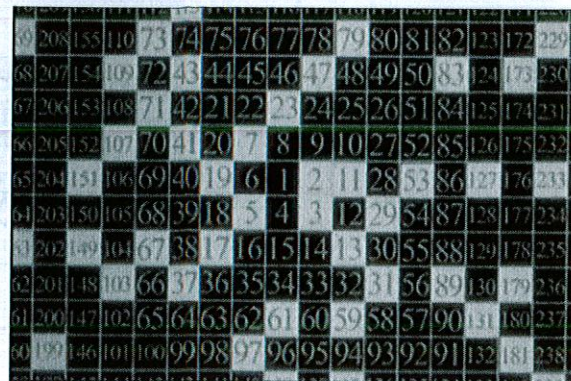
Module 19: Cryptography Algorithms and Deployment

Module 19: Cryptography Algorithms and Deployment

This page intentionally left blank.

Objectives

- Crypto concepts
- Symmetric and asymmetric cryptosystems
- Crypto attacks



69	208	155	110	73	74	75	76	77	78	79	80	81	82	123	172	229
68	207	154	109	72	43	44	45	46	47	48	49	50	83	124	173	230
67	206	153	108	71	42	21	22	23	24	25	26	51	84	125	174	231
66	205	152	107	70	41	20	7	8	9	10	27	52	85	126	175	232
65	204	151	106	69	40	19	6	1	2	11	28	53	86	127	176	233
64	203	150	105	68	39	18	5	4	3	12	29	54	87	128	177	234
63	202	149	104	67	38	17	16	15	14	13	30	55	88	129	178	235
62	201	148	103	66	37	36	35	34	33	32	31	56	89	130	179	236
61	200	147	102	65	64	63	62	61	60	59	58	57	90	131	180	237
60	199	146	101	100	99	98	97	96	95	94	93	92	91	132	181	238

The previous module gave you a tour of some of the important issues and concepts in the field of cryptography. It showed that encryption is real, it is crucial, it is a foundation of much that happens in the world around us today, and most of all, it is mostly transparent to us.

One of this module's goals is to show you how some of the world's most popular ciphers, which are in constant use in many different sectors, operates. Along the way, we share some pragmatic lessons we learned the hard way, hoping our experience will help you in the future.

Reference

1. Journey in Cryptography, <https://www.khanacademy.org/computing/computer-science/cryptography>

Crypto Concepts

The student will have a high-level understanding of the mathematical concepts that contribute to modern cryptography

Crypto Concepts

This page intentionally left blank.

Concepts in Cryptography (I)



- Probability Theory
- Information Theory
- Complexity Theory
- Number Theory
- Abstract Algebra
- Finite Fields

What if...

- We can find a mathematical "problem" that exhibits characteristics of one-way functions (with trapdoors)? Or, as mathematicians would prefer to say, a problem that is "impossible" to solve in polynomial time?

Hmm...

- We could use it to build a new cryptosystem!

Confidentiality

Integrity of Data

Authentication

Non-repudiation

SANS

SEC401 | Security Essentials Bootcamp Style

61

The previous module defined the four main goals of a cryptosystem: confidentiality, integrity of data, authentication, and non-repudiation. However, how do we construct a cipher that enforces these characteristics? Mathematics has fields such as probability theory, information theory, complexity theory, number theory, abstract algebra, and finite fields that are all rich in ideas that can contribute to our cipher.

The previous section also introduced one-way mathematical functions. Such functions can have trapdoor properties that make them well-suited for public key cryptography, in which the trapdoor allows a message to be decrypted using a different key than the one used to encrypt the message. If the public key were used to encrypt the message, the trapdoor, in this case, is the corresponding private key.

One-way functions that are computationally hard—that is, impossible to solve in polynomial time—can make things difficult for an adversary eavesdropping on our communications, say over an insecure public network like the Internet. At the same time, the existence of a trapdoor can be used to provide an easy solution to the intractable problem for use by the sender or the recipient.

Concepts in Cryptography (2)

Computational Complexity deals with time and space requirements for the execution of algorithms.

Problems can be **classified** as tractable or intractable.



This is exactly the class of problems we are looking for!

Tractable Problems

"Easy" problems. Can be solved in polynomial time (i.e., "quickly") for certain inputs

Examples:

- constant problems
- linear problems
- quadratic problems
- cubic problems

Intractable Problems

"Hard" problems. Cannot be solved in polynomial time (i.e., "quickly")

Examples:

- exponential or super-polynomial problems
- factoring large integers into primes (RSA)
- solving the discrete logarithm problem (El Gamal)
- computing elliptic curves in a finite field (ECC)

Mathematics is filled with intractable problems. So, a cipher designer can start by just picking one and trying it out. Evaluating an algorithm's computational complexity reveals the time and space required to execute it and helps us classify the problem as either tractable (easy) or intractable (hard).

When computers are used to solve problems, we don't care about the exact number of operations—we are more interested in how the amount of input to the problem (or program) affects the number of operations it takes to solve (or execute). Big-O notation is used to give a general idea of how many operations a problem takes relative to the input size n . The big-O function isn't usually specifically defined; it is mostly used as a notational shorthand to indicate a problem's complexity.

Relatively easy problems (symmetric encryption) can be solved in polynomial time—that is, the relationship between the input size and the number of operations required to solve the problem is constant, linear, quadratic, cubic, and so on. Constant time, $O(1)$, means they take the same number of operations to solve regardless of the input size. Linear time, $O(n)$ means the number of operations increases linearly with the input size—when the input size is doubled, the problem takes twice as long to solve. Quadratic time is $O(n^2)$, cubic time is $O(n^3)$, and so on.

Problems are considered intractable (or hard) when they cannot be solved in polynomial time (asymmetric encryption). Examples are exponential, $O(2^n)$ and superpolynomial (somewhere between polynomial and exponential), which are considered very complex, as in "hard" or "intractable". A cubic-time algorithm might take thousands of years to solve, whereas an exponential-time algorithm might take longer than the universe is expected to last.

It can be difficult to prove whether a problem is intractable or not. Someone might prove a particular problem can be solved in superpolynomial time, only to have someone later discover it can be solved a different way in polynomial time. So, it is more accurate to state that the problems we use in cipher algorithms are believed to be intractable by most researchers in complexity theory.

There's always the highly unlikely chance that easier solutions have been overlooked or just haven't been discovered yet.

Three well-known examples of intractable problems include factoring large integers into their two prime factors (the basis for RSA), solving the discrete logarithm problem over finite fields (the basis for El Gamal), and computing elliptic curves over finite fields (the basis for Elliptic Curve Cryptosystems). Now, let's examine each of these three important classes of intractable problems in greater detail because each one of these forms the basis of important cryptosystems, which are widely used all over the world today.

Concepts in Cryptography (3)

An Example of an Intractable Problem...

Difficulty of factoring a large integer into its two prime factors

- A "hard" problem
- Years of intense public scrutiny suggest intractability
- No mathematical proof so far

Example: RSA

- based on difficulty of factoring a large integer into its prime factors
- ~1000 times slower than DES
- considered "secure"
- *de facto* standard
- patent expired in 2000

Factoring integers doesn't seem that hard. It doesn't take much thought to figure out that 15 can be factored into 1×15 and 3×5 . So, why is it on our list of intractable problems?

The operative word here is "large." The larger the integer, the harder it is to factor. In fact, there is no known recipe for factoring other than trial and error: keep multiplying primes together until you arrive at the number. Remember that even though most researchers in complexity theory believe factoring large integers is a hard problem, there is no unequivocal proof to that effect. It is only the years of public scrutiny of the problem that lead us to conclude the problem cannot be solved in polynomial time.

Perhaps the most popular public-key algorithm today, RSA, takes advantage of the intractability of the integer factorization problem. We discuss RSA later in this module.

Another Intractable Problem...

Difficulty of solving the discrete logarithm problem - for finite fields

- A "hard" problem.
- Years of intense public scrutiny suggest intractability.
- No mathematical proof so far.
- The discrete logarithm problem is as difficult as the problem of factoring a large integer into its prime factors.

Examples

- El Gamal encryption and signature schemes
- Diffie-Hellman key agreement scheme
- Schnorr signature scheme
- NIST's Digital Signature Algorithm (DSA)

Another intractable problem is the discrete logarithm problem for finite fields. The discrete logarithm is based on a statement of the form $a^x \bmod n = b$, where a , b , n , and x are integers and a and n are known. The mod operator just means we take the remainder of the first number (a^x) when divided by the second number (n). Finding b when we know x is easy, but not the other way around.

For example, it is easy to calculate $8^3 \bmod 7$ - because $8^3 = 512$ and the next lowest multiple of 7 is 511, the remainder must be $512 - 511 = 1$. But, it takes trial and error to discover that $8^x \bmod 7 = 1$ is satisfied only by $x = 3$. This problem is the discrete logarithm. Just as with prime factorization, the problem really gets hard when x is a hundred- or thousand-bit number.

Again, the notion that discrete logarithms are intractable is the consensus of computational complexity researchers, and there is no unequivocal proof that this problem cannot be solved easily. It is the years of public scrutiny of the problem that leads us to conclude that it is a hard problem that cannot be solved in polynomial time. You can prove an algorithm is not secure by breaking it; you just cannot prove an algorithm is secure. But how does it compare with the previous intractable problem we looked at—the factorization of large integers into two primes? Evidence shows the discrete logarithm problem is just as difficult.

We should be able to use the discrete logarithm problem in building a cipher. In fact, several ciphers in use today are built upon the intractability of the discrete logarithm problem over finite fields: the El Gamal encryption and signature schemes, the Diffie-Hellman key agreement scheme, and the Schnorr signature scheme.

Concepts in Cryptography (5)

Yet Another Intractable Problem...

Difficulty of solving the discrete logarithm problem
--as applied to elliptic curves

- A "hard" problem
- Years of intense public scrutiny suggest intractability
- No mathematical proof so far
- In general, elliptic curve cryptosystems (ECC) offer higher speed, lower power consumption, and tighter code.

Examples

- Elliptic curve El Gamal encryption and signature schemes
- Elliptic curve Diffie-Hellman key agreement scheme
- Elliptic curve Schnorr signature scheme
- Elliptic Curve Digital Signature Algorithm (ECDSA)

The ciphers named in the previous section use the discrete logarithm problem, but only for certain sets of numbers that belong to what are known as finite fields. This problem also makes for a good cipher algorithm when applied to elliptic curves.

This class of problem is considered every bit as intractable as the previous two. Plus, it lends some additional useful features to our algorithm: high security levels even at low key lengths, high-speed processing, and low power and storage requirements. These characteristics are useful in crypto-enabling the many new devices that are rapidly appearing in the marketplace (for example, mobile telephones, information appliances, smart cards, and an ATM).

Symmetric, Asymmetric, and Hashing Cryptosystems

The student will have a basic understanding of commonly used symmetric, asymmetric, and hashing cryptosystems

Symmetric, Asymmetric, and Hashing Cryptosystems

This page intentionally left blank.

DES

- Data Encryption Standard
- Released March 17, 1975
- Rather fast encryption algorithm
- Widely used; a de facto standard
- Symmetric key, 64-bit block cipher
- 56-bit key size: Small 2^{56} keyspace
- Today, DES is not considered secure

DES used to be one of the more commonly used encryption algorithms in the world, which is based on IBM's Lucifer cipher.

Because of the internal bit-oriented operations in the design of DES, software implementations are slow, whereas hardware implementations are faster. There are four different DES operation modes for use: electronic codebook (ECB) mode, cipher block chaining (CBC) mode, output feedback (OFB) mode, and cipher feedback (CFB) mode.

DES Weaknesses

- DES is considered non-secure for very sensitive encryption. It is crackable in a short period of time
- See the book, *Cracking DES* (O'Reilly)
- Multiple encryptions and key size increase the security
- Double DES is vulnerable to the meet-in-the-middle attack and only has an effective key length of 57 bits
- Triple DES is preferred

From the beginning, concerns were raised about the strength of DES, due to the rather small key length of 56 bits (a 64-bit ciphertext block minus 8 bits for parity), resulting in a keyspace containing only 2^{56} possible different keys. The effectiveness of attacks based on brute-force searches depends upon keyspace size. Because of DES's relatively small keyspace, brute-force attacks are feasible. DES was first (publicly) cracked in the RSA Challenge, a program that offers monetary rewards for breaking ciphers and solving computationally intensive mathematical problems. The DES challenge took only five months for the public to solve, and subsequent attempts are taking less and less time.

Consequently, DES is no longer considered secure because of its key size, and not because the algorithm has been broken. In fact, anyone can build a DES-cracking engine these days. All the information you need, including sample code, is available in a book called *Cracking DES*. However, with the global e-commerce infrastructure build-out proceeding at a furious pace, because of all the new e-business initiatives that are sprouting up all over the world, the need for a fast, symmetric block cipher is extremely urgent. If DES can no longer be considered to be secure, what can we do in the interim?

Again, DES was already widely deployed in both hardware and software products, and it had withstood unbridled cryptanalysis for decades. It didn't take long to realize what a great advantage it would be to somehow increase DES's key size and use the existing implementations until a new standard was built.

One way to effectively increase the key length is to perform the encryption more than once. That is, encrypt the cleartext, and then encrypt the resulting ciphertext, and so on. But, this only works if the cipher algorithm is not a group.

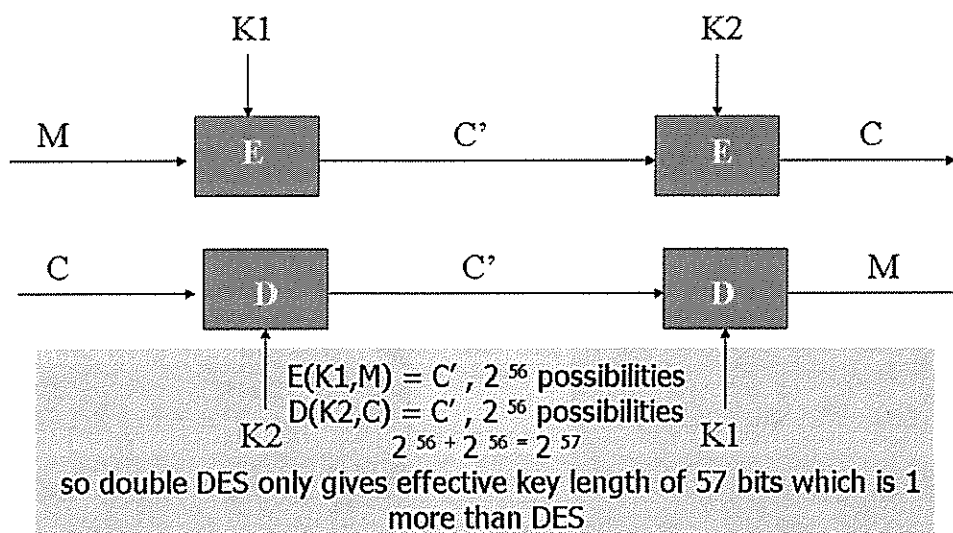
DES Is Not a Group

- If an algorithm is a group, then
 - $E(K_2, E(K, M)) = E(K_3, M)$
- In 1992, it was proven that DES is not a group. This means that multiple DES encryptions are not equivalent to a single encryption. **THIS IS A GOOD THING!**
- Because DES is not a group, multiple encryptions increase the security

The function E is a group if $E(K_2, E(K, M)) = E(K_3, M)$. In other words, encrypting once with key K and then again with key K_2 is equivalent to encrypting once with K_3 . Thus, if a cipher algorithm is a group, encrypting multiple times is no stronger than encrypting once.

Whether an algorithm is a group is an important statistical consideration. If it is a group, then applying the algorithm multiple times is a waste of time. In 1992, it was proven that DES is not a group, in fact, so encrypting multiple times with DES is not equivalent to encrypting once. That's good news; it means that encrypting more than once with DES can increase the security of the ciphertext.

Meet-in-the-Middle Attack



Encrypting twice with DES (Double DES) does not significantly increase the effective key size. If a cryptanalyst is able to obtain both a cleartext message (M) and its corresponding ciphertext (C), they can perform a meet-in-the-middle attack.

We already mentioned that brute-force attacks on DES are feasible, which means we can attempt to decrypt a message with every possible key until we find the one that gives us sensible cleartext. For a meet-in-the-middle attack, we first encrypt the cleartext M with every possible key K1:

$$C' = E(K1, M)$$

giving us 2^{56} values of intermediate ciphertext C'. Then, we decrypt the ciphertext C with every possible key K2:

$$C' = D(K2, C)$$

again giving us 2^{56} values of C'. The values of K1 and K2 that yield the same C' in the previous equations are the two keys used for the double DES encryption. The number of operations and, therefore, the resulting key length, is only $2^{56} + 2^{56} = 2^{57}$. This gives us an effective key length of only 57 bits, which is only 1 more bit than DES.

Triple DES

USAGE

Still utilized today even though the key length is 168 bits and not considered secure in some environments

Prefer Triple DES over DES (which is, officially, no longer considered to be secure).

VULNERABILITIES

Cracking Triple DES means examining all possible pairs of crypto-variables.

So far, there have been no public reports claiming to have cracked Triple DES.

Based on the key length Triple-DES is not considered secure

To thwart meet-in-the-middle attacks, Triple DES adds a third round of encryption. Thus, when performing the two steps of the meet-in-the-middle attack, a cryptanalyst ends up with two sets of ciphertext that won't be comparable—they are separated by another encryption step. Triple DES is well-known and widely implemented, and it has been intensely scrutinized by the global community of cryptologists. Furthermore, it uses the same tried and true DES algorithm, and all existing DES implementations can be used to perform Triple DES (3DES).

Triple DES can be configured to use either two or three unique keys, yielding a key strength of either 112 bit (2 Keys) or 168 bit (3 Keys).

Triple DES conducts three passes of the DES algorithm. There is a concept of a round that represents the number of iterations within the algorithm. Each encryption algorithm has its own specification regarding the number of rounds—DES uses 16 rounds. To appreciate the extra effort required to use Triple DES, the standard DES algorithm is executed 16 times (rounds), whereas the Triple DES is executed 48 times (rounds). Thus, Triple DES requires three times the amount of resources to perform the encryption and decryption.

AES Overview

Advanced Encryption Standard

- A new encryption algorithm(s) that is designed to be effective well into the 21st century

THE FIVE "AES" FINALISTS !

- **MARS** IBM
- **RC6™** RSA Laboratories
- **Rijndael** Joan Daemen, Vincent Rijmen
- **Serpent** Ross Anderson, Eli Biham, Lars Knudsen
- **Twofish** Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, Niels Ferguson

Significance

Developing "good" cryptographic algorithms that can be trusted is hard. The only practical way to develop such algorithms is to perform the development process in an open manner, and under intense public scrutiny of the global cryptographic community. Can you think of a recent example in which this was not followed?

Countdown to AES !

- 1/2/1997, the quest for AES begins...
- 8/9/1999, five finalist algorithms announced
- Announced winner – Rijndael
- 12/26/2001 – AES approved!

SANS

SEC401 | Security Essentials Bootcamp Style 73

On January 2, 1997, NIST announced the initiation of an effort to develop the Advanced Encryption Standard (AES). A formal call for algorithms was made on September 12, 1997. The call stipulated that AES must specify unclassified, publicly disclosed encryption algorithm(s), available royalty-free, worldwide. In addition, the algorithm(s) would implement symmetric key cryptography as a block cipher and (at a minimum) support a block size of 128 bits and key sizes of 128, 192, and 256 bits. The evaluation criteria were divided into three major categories: security, cost, and algorithm and implementation characteristics.

NIST selected the five AES finalists on August 9, 1999. In October 2000, Rijndael (pronounced "Rain Doll") was announced as the winner and was approved as the official AES cipher. The two Belgian researchers who developed Rijndael are Dr. Joan Daemen (YO-ahn DAH-mun) of Proton World International and Dr. Vincent Rijmen (RYE-mun) of Katholieke Universiteit Leuven.

On December 26, 2001, NIST announced the approval of FIPS 197, which describes AES as an official government standard, by the U.S. Secretary of Commerce. FIPS 197 became effective on May 26, 2002.

The AES has supplanted the inadequate 56-bit DES, which is to be used only in legacy systems. AES has three initial key sizes: 128-bit, 192-bit, and 256-bit. Testing of the algorithm was performed by NIST and the Canadian Communications Security Establishment (CSE).

Reference

1. THE ADVANCED ENCRYPTION STANDARD: A STATUS REPORT - NIST

Cipher	Developers:
MARS	IBM
RC6™	RSA Laboratories
Rijndael	Joan Daemen, Vincent Rijmen
Serpent	Ross Anderson, Eli Biham, Lars Knudsen
Twofish	Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, Niels Ferguson

AES Algorithm

```

Cipher (byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
  byte state[4,Nb]

  state = in

  AddRoundKey(state, w)

  for round = 1 step 1 to Nr-1
    SubBytes(state)
    ShiftRows(state)
    MixColumns(state)
    AddRoundKey(state, w+round*Nb)
  end for

  SubBytes(state)
  ShiftRows(state)
  AddRoundKey(state, w+Nr*Nb)

  out = state
end

```

AES algorithm employs four basic transformations:

- **AddRoundKey:** XOR Round Key with State
- **SubBytes:** Substitute bytes in State s to form State s' on a byte-for-byte basis using S-box
- **ShiftRows:** Left circular shift of rows 1-3 in State s by 1, 2, and 3 bytes, respectively
- **MixColumns:** Apply mathematical transformation to each column in State s to form State s'

AES Algorithm

The pseudo-code for the AES algorithm displays in the slide. Let's discuss it briefly:

- $in[]$ is an array containing the block of plaintext. In general, $in[]$ is $4 * Nb$ bytes in size—16 bytes in the current AES specification.
- $out[]$ is an array containing the ciphertext and is also $4 * Nb$ bytes in size.
- $w[]$ is the array containing the expanded key and is $Nb * (Nr + 1)$ words in size.

The code also shows one additional data structure: state – a two-dimensional array containing the current value of the transformed ciphertext. The transformations themselves are defined by four function calls: AddRoundKey(), SubBytes(), ShiftRows(), and MixColumns(). The specifics of these transformations are described in the next section.

The rest of the code is straightforward:

- The plaintext is moved into the state[] array
- The first round key is applied
- There are then $Nr - 1$ rounds that apply all four transformations
- The final round applies to all but the MixColumns() transformation
- The state[] array is moved into the ciphertext data structure

AES Basic Functions

The AES algorithm employs four basic transformations:

- AddRoundKey(): Takes the appropriate round key and performs a bit-by-bit XOR with the current state.
- SubBytes(): Using a substitution box (S-box) defined in the specification, substitutes each 8-bit quantity in the State array to a different 8-bit value.

- **ShiftRows()**: Circularly shifts left the contents of state array rows 1, 2, and 3 by 1, 2, and 3 bytes, respectively. A left circular shift of one byte, for example, means that the bytes in columns 1, 2, and 3 move to positions 0, 1, and 2, respectively, and the value in byte position 0 moves to position 3.
- **MixColumns()**: Another byte value substitution, but in this case, performed on a column (32-bit) basis; for example, rather than perform an S-box substitution on a per-byte basis, this transformation applies a polynomial transformation on four bytes at a time.

AES

USAGE

The AES algorithm has been developed to replace DES/3-DES, which is no longer officially considered to be secure.

DES/Triple DES is very widely used throughout the world today, and AES is just as popular...

VULNERABILITIES

No major vulnerabilities reported and viewed as a solid replacement for DES/3DES.

Only feasible attack is brute forcing the keys.

The AES development process has given us a splendid opportunity to see first-hand what it takes to develop a cryptographic algorithm. The process is inherently complex, and the only realistic way to reduce the risk of producing a weak algorithm is to open up the development activity to all interested parties and to the intense scrutiny by the global community of cryptologists.

RSA

USAGE

Widespread support in major web clients, such as Microsoft Internet Explorer.

Has become even more popular since the patent expired in 2000.

VULNERABILITIES

Cracking RSA generally means compromising poor implementations or those using small key lengths.

So far, there have been no public reports claiming to have compromised the RSA algorithm itself.

SANS

SEC401 | Security Essentials Bootcamp Style 77

The RSA algorithm has been widely implemented all over the world in all kinds of cryptography-enabled applications. It can be used to support both encryption and digital signature schemes. As a central part of the Secure Sockets Layer (SSL), it is also included in major web clients, such as Microsoft Internet Explorer.

Although there have been a large number of claims to having cracked the RSA algorithm, they have all turned out to be false. Vulnerabilities have been found in certain RSA implementations, however. Poor implementations of the RSA algorithm can be compromised, but as in the case of other cryptographic algorithms, it does not mean the algorithm itself has been cracked.

The working mechanism of most public key (asymmetric) cryptographic algorithms are generally openly published and widely known. The security of the cryptosystem comes from the secrecy and size of the private key and not from the secrecy of the algorithm itself. As for other cryptographic algorithms, it is important to ensure that the key size is not so small that brute-force attacks become feasible due to the small size of the resulting key space.

Elliptic Curve Cryptosystem

USAGE

Where high speed, low power consumption, low storage requirements, and high security at small key lengths is critical, e.g., in wireless communications, electronic cash, and ATMs.

Growing in popularity.

VULNERABILITIES

Cracking ECC generally means compromising poor implementations, or those using small key lengths.

So far, there have been no public reports claiming to have cracked the ECC algorithm itself ...

Elliptic Curve Cryptosystems (ECCs) are capable of supporting both an encryption/decryption scheme and a digital signature scheme. In addition, the ECC has some interesting characteristics: high security even at relatively small key lengths (that is, a higher strength per bit), high-speed implementations, low processing power requirements, and low storage requirements.

The previous properties make ECC a particularly attractive cryptographic option for use in resource-constrained computing environments such as mobile telephones, information appliances, and smart cards.

Through the efforts of Certicom and others, ECC has enjoyed strong acceptance over the last 5 years and has been included in SSL/TLS standards and NIST standards. We expect to see increasing deployments of ECC-enabled applications in our e-commerce-enabled environments.

Comparing Key Length

Security (Bits)	Symmetric	DSA/DH	RSA	ECC
80		1024	1024	160
112	3DES	2048	2048	224
128	AES-128	3072	3072	256
192	AES-192	7680	7680	384
256	AES-256	15360	15360	512

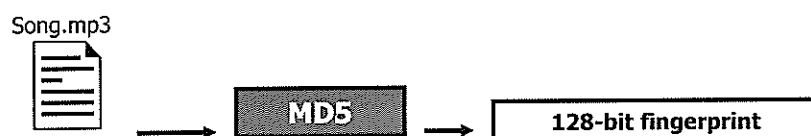
Typically within a given crypto algorithm, stronger ciphers result as the key size increases. When comparing key sizes across different algorithms, then the rule changes.

When comparing symmetric versus asymmetric cryptosystems not only are asymmetric algorithms more resource intensive (eat up your CPU time) than symmetric routines, but they are also less secure at the same key length. To compensate for this disparity and to increase the work factor for cracking an asymmetric key pair, asymmetric key sizes are typically much longer. Although symmetric keys range from 40 bit to 256 bit, asymmetric keys are typically 1,000 bits or longer.

The table in the slide, taken from NIST, provides a comparison of key lengths for various algorithms. We can see that using Triple DES with two keys (112 bit) is equivalent in strength and work factor to RSA using 2040-bit keys. AES using 192-bit keys is as strong as RSA with key sizes approaching 8,000 bits. ECC also requires longer key sizes than symmetric crypto algorithms, but only needs to be double the size.

MD5

MD5 takes variable-length input
Output is 128-bit unique fingerprint
Typically used with digital evidence



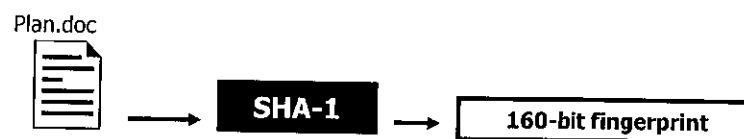
MD5 was created by Ron Rivest in 1991 and is more secure than its precursors, MD2 and MD4. MD5 is actually an extension of the MD4 algorithm and is more conservative in design. MD4 executed incredibly fast, but the cryptographic community felt MD4's collision protection was less than optimal. MD5's conservativeness increases its security but diminishes its execution speed. Most users don't mind sacrificing a little speed for the sake of enhanced security.

MD5 accepts arbitrary lengths of input and produces a fixed-length output that is 128 bits, which is referred to as the key length. A hashing algorithm's output might be referred to as a hash, digest, or fingerprint. In the illustration, the song.mp3 file is input into MD5, and a 128-bit unique fingerprint of the file is created. MD5 does not modify the original file in any manner whatsoever.

As an extra precaution, security-conscious users may choose to cryptographically sign the fingerprint to guard the fingerprint against inadvertent modification. Digitally signing a fingerprint cryptographically safeguards the integrity of the fingerprint.

SHA-1/SHA-2

SHA-1 Output is 160-bit unique fingerprint
SHA-2
SHA-256 Output is 256-bit unique fingerprint
SHA-512 Output is 512-bit unique fingerprint



NIST developed the Secure Hash Algorithm (SHA) in 1993. SHA-1 was released in 1994 to correct an unpublished flaw in the original release of SHA. SHA and SHA-1 are pronounced "shaw" and "shaw-one" respectively.

SHA-1 produces a 160-bit fingerprint compared to MD5's 128-bit fingerprint. Although SHA-1 is slower than MD5, SHA-1's larger fingerprint makes it more secure against collision attacks.

In the illustration, the plan.doc file is input into SHA-1; subsequently, SHA-1 produces a 160-bit fingerprint. As with MD5, security-conscious users would cryptographically sign the fingerprint to safeguard the fingerprint against inadvertent modification.

With both MD5 and SHA-1, security-conscious users must explicitly conduct a second, independent step to protect the fingerprint against modification. Hashing algorithms do not provide any intrinsic means of protecting the fingerprint once generated.

SHA-2 increases the output to 256 bits with SHA-256 and 512 bits with SHA-512.

Cryptanalysis

The student will be able to identify common attacks used to subvert cryptographic defenses



Cryptanalysis

This page intentionally left blank.

Cryptanalysis

Analytic

- Uses algorithms and mathematics to deduce key or reduce key space to be searched

Statistical

- Uses statistical characteristics of language or weaknesses in keys

Differential

- Analyzes resultant differences as related plaintexts are encrypted using a cryptographic key

Linear

- Linear analysis of pairs of plaintext and ciphertext

Differential linear

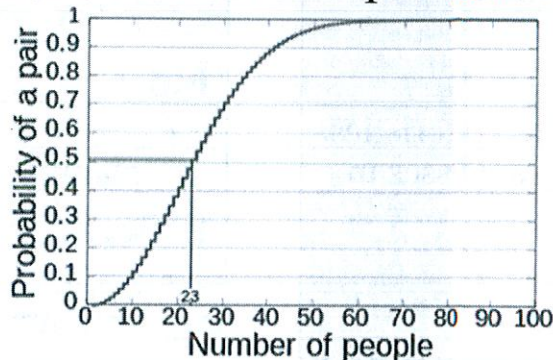
- Applies differential analysis with linear analysis

We now look at some general types of cryptanalysis:

- **Analytic:** Uses algorithms and mathematics to deduce key or reduce key space to be searched
- **Statistical:** Uses statistical characteristics of language or weaknesses in keys
- **Differential:** Analyzes resultant differences as related plaintexts are encrypted using a cryptographic key
- **Linear:** Linear analysis of pairs of plaintext and ciphertext
- **Differential linear:** Applies differential analysis with linear analysis

Birthday Attack

- When 23 people are put together, the odds are greater than 50% that 2 or more people share a birthday
- Hash collisions is related to that probability



Cryptanalysts can sometimes use a phenomenon known as the *birthday paradox* to attack hash signatures. People in large groups often find that at least two of them share the same birthday. They're usually astonished at the coincidence, thinking that the odds must be very slim that two people could be born on the same day of the year. It is true that it would be rather unusual to find a person with your exact birthday unless the group was very large. The odds of finding someone born on a particular day are 1 in 365 (assuming all days of the year are equally likely birthdays and nobody was born on February 29).

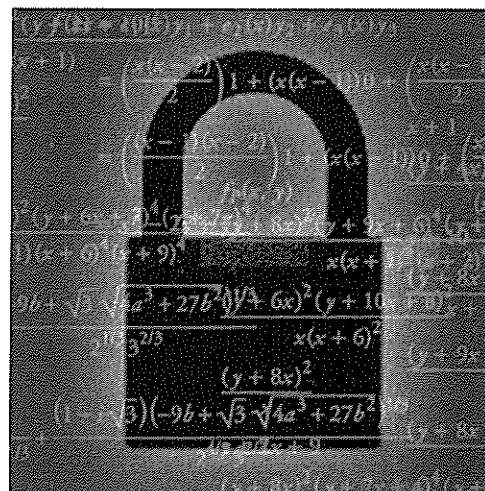
However, just specifying that any two people have the same birthday, without specifying whose birthday it is, improves the odds considerably. For a group as small as 23 people, the odds are greater than 50% that two or more of them share a birthday. If each of the 23 people compares birthdays with another, you'd have 253 comparisons. The odds, then, that none of the 23 have the same birthday are $(364/365)^{253} = 0.4995$. Thus, the odds that two of them share a birthday are $1 - 0.4995 = 0.5005$.

Just as pairs of people in a group might have the same birthday, pairs of messages might have the same hash signature. Of course, there are much more possibilities for hash signatures than birthdays, but the same logic applies. If an attacker can find any two messages that generate the same hash value, that is, a collision, they could substitute one message for the other at will. For example, maybe they have a list of password hashes but not the cleartext. If they can hash enough on their own generated cleartext to cause a collision, they have a password that works just as well as the real thing.

The entire attack is a statistical probability problem.

Summary

- Concepts in Cryptography
- Symmetric (Private) Key Systems
 - Triple DES
 - AES
- Asymmetric (Public) Key Systems
 - RSA
 - ECC
- Hashing



Cryptography is essential for e-commerce, military communications, and the privacy of individuals on the Internet and other networks. A cryptographic algorithm must provide confidentiality, integrity of data, authentication, and non-repudiation. To achieve these goals, mathematical problems with a suitable computational complexity are used to create a cipher that takes too long to break using brute-force methods to be practical. Such problems are considered intractable, and although they cannot usually be proven to be secure, years of research and scrutiny leads us to believe they are.

The Data Encryption Standard (DES), introduced in 1975, is too easy to brute force for serious use today, but Triple DES effectively lengthens that key size. Triple DES employs the already available DES implementations and takes advantage of DES's decades of public scrutiny.

A permanent replacement for DES was needed to last for decades to come, and that was chosen as the Advanced Encryption Standard (AES) in October 2000. Rijndael, chosen from one of five finalists, is a symmetric cipher with possible key sizes which include: 128-bit, 192-bit, and 256-bit.

Symmetric ciphers aren't right for every application. When two parties have never met, such as a merchant and a customer, secure key exchange is not easy. Public key cryptography solves this problem by allowing for separate keys for encryption and decryption. The famous RSA algorithm is one such cipher, and it is used by the Secure Sockets Layer (SSL) to provide secure communications on the Internet. Because symmetric cryptography is much faster, RSA is often used to exchange a session key for symmetric algorithms, which then encrypts the transaction.

Proposed in 1985, elliptic curve cryptosystems (ECCs) offer the possibility of strong cryptography with low overhead. Thus, ECC lends itself to embedded applications in which memory and processing speed are at a premium.

When attacking a cipher, a cryptanalyst takes advantage of what information they have. Plaintext, ciphertext, and relationships between keys can all be useful. Being able to choose the text that gets encrypted or decrypted

can be even more useful, because the cryptanalyst can deduce information based on his own input. For attacking hash algorithms, the birthday paradox makes it surprisingly likely to find collisions, two messages that hash to the same value.

Perhaps the most important lesson in this module is that ciphers should be developed in the open, taking advantage of the collective brainpower of cryptologists throughout the world. This kind of scrutiny reduces the likelihood that a weak algorithm is used and encourages cipher designers to place all of a cryptosystem's security in the key rather than the algorithm itself.

Reference

1. Cryptography and Security, <https://sites.northwestern.edu/cs101/cryptography-and-security/>

SANS

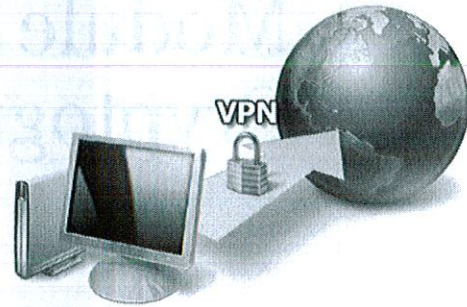
Module 20: Applying Cryptography

Module 20: Applying Cryptography

As was discussed in the previous two modules, cryptography has many applications in information security. You can consider it the Swiss Army knife of information security because it has so many useful purposes. The top purposes are confidentiality, integrity, authentication, and non-repudiation. By encrypting information with a key that only the rightful users of the information possess, it can be used to protect the information from prying eyes (confidentiality). It can also be used to detect information tampering (integrity) by cryptographically hashing the information and then encrypting the hash. If the information is tampered with, the hash will not match, proving that the information has been modified. Cryptography can be used to prove identity (authentication). This can be done by requesting that the user encrypts a test message. The encrypted test message is then decrypted using the user's stored key. If the message decrypts successfully, the user has proven his identity, or at least that he possesses the right key! Non-repudiation allows someone to prove in a court of law that someone agreed to a contractual relationship.

Objectives

- Data in transit
 - Virtual private networks (VPNs)
- Data at rest
 - Data encryption
 - Full disk encryption
 - GNU Privacy Guard (GPG)
- Key management
 - Public key infrastructure (PKI)
 - Digital certificates
 - Certificate authorities (CA)



The abilities that cryptography offers us are great, but how are they being used in real-world networks? Here, we discuss some of the practical applications of cryptography, including how cryptography can be used to protect communications across a network, protect information resting in storage, provide authentication services, and ensure the integrity of information.

Our discussion begins with one of the most common uses for cryptography: protecting information as it flows across a network. Information is exposed when it leaves your PC to travel across a network to another PC or server. At any point between the source and destination of a message, a man-in-the-middle can capture or modify the information contained within the message. What does this mean in the real world? Well, without encryption, an attacker might be able to capture your credit card details as you provide them to an online retailer.

Potentially more damaging, many network protocols do not encrypt their session information. These protocols transmit your username and password information "in the clear." An attacker who can listen in on your network conversations when you use one of these unencrypted protocols can impersonate you, gaining access to all the network resources you have access to through that protocol. For our last example, consider the damage an attacker could cause by simply modifying the contents of the right network conversation. Changing an account number used during a banking transfer or the ship-to address during an online purchase could defraud you of potentially large amounts of money. Cryptography provides a powerful method to protect against these information security risks.

A basic question when protecting network communications is where the protection should be performed. Should each application be responsible for protecting its own network communications, should cryptography be implemented as a service that applications can optionally use, or should it be included at the network level where all communication from and to particular locations can be protected? In practice, all of these methods are currently in use.

It is also important to not only protect the data in transit but also at rest. Various methods of data encryption will be discussed including the use of full disk encryption. A practical implementation of crypto called GNU Privacy Guard (GPG) will not only be discussed in the slides, but there will also be a lab covering its use.

Finally crypto is only as good as the use and protection of the keys. Key management to include digital certificates and certificate authorities will also be covered.

Virtual Private Networks (VPNs)

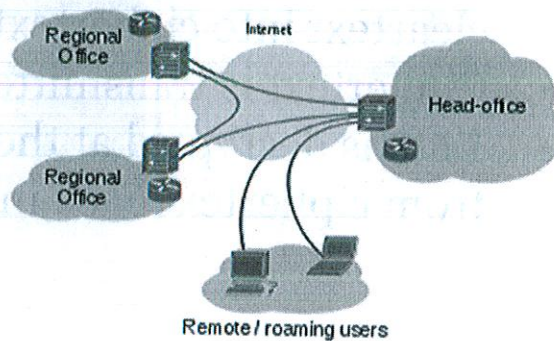
The student will have a high-level understanding of what VPNs are and how they operate

Network-layer encryption protects network conversations whether the application using the network supports cryptography or not. Network layer encryption sits in-between the transmitter and receiver. It accepts in clear-text information and then encrypts it prior to sending it out. At the receiving end, the information is decrypted and forwarded on to its final destination. This type of network encryption is called a virtual private network (VPN).

Confidentiality in Transit

Private network

- **Pro:** Dedicated lines and equipment are not shared by others
- **Con:** Dedicated lines are expensive, grow more so with distance, and are underutilized



SANS

SEC401 | Security Essentials Bootcamp Style 91

Prior to the popularization of VPNs, companies wanting to protect network conversations between different locations purchased dedicated leased lines, MPLS, ATM connections, or other types of private circuits that provided connectivity between the sites from a telecommunications company. They could be reasonably confident that their information could not be intercepted because these circuits allow only the two sides of the connection to exchange information. No third parties should be able to communicate over the private connection. This assumes that you trust the telecommunications provider, which might or might not be a good bet depending upon where in the world you are. Although secure, these circuits also tend to be slow and expensive and become more expensive as they get faster, or the distance increases between the sites that need to communicate. There is also a large lead-time between the decision to set up one of these connections and getting it running. It can take months for a telecommunications company to fulfill a new circuit order and typically you have to sign a contract for at least a year.

Reference

1. Virtual Private Network, https://en.wikipedia.org/wiki/Virtual_private_network

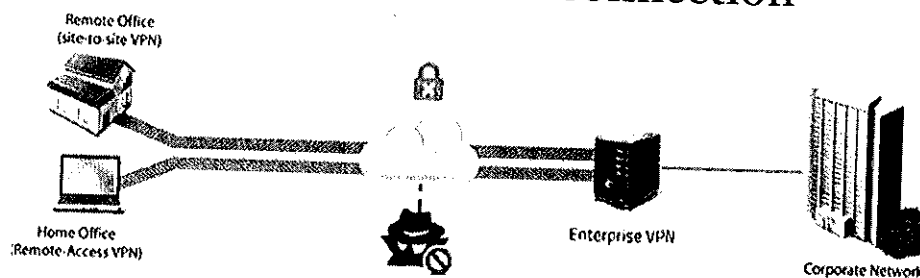
Virtual Private Network (VPN)

- Data is encrypted at one end of the VPN from cleartext into ciphertext
- Ciphertext is transmitted over the Internet
- Data is decrypted at the other end of the VPN from ciphertext back into the original cleartext

VPNs are a perfect alternative to costly, inflexible private circuits. They give companies the option of setting up virtual circuits across public networks, such as the Internet. Encryption provides the confidentiality needed as the private information flows across the public network. This capability allows VPNs to establish secure communication between different remote offices and can also be used to establish remote access to internal network resources by employees from their homes or while they are on travel.

VPN Advantage – Flexibility

- VPNs are flexible
- A VPN "tunnel" over the Internet can be set up rapidly; a frame circuit can take weeks
- All you need is an Internet connection



SANS

SEC401 | Security Essentials Bootcamp Style 93

One of the biggest benefits of VPN technology is its flexibility. If you need a secure channel between two hosts only for a day, or even an hour, then a VPN might fit the bill. After you have all the components to establish a VPN, setting one up requires only configuration. This makes the technology far more flexible than private circuits, which must be ordered far in advance of their use and might also require additional hardware. This flexibility lends itself to creating new business solutions. For example, it's not cost-effective to wire a T1 for every employee who works from home. It is practical, however, to load software on an employee's laptop and let them connect from their home office via a VPN over the Internet. This assumes that the home users already have connections to the Internet.

Reference

1. What is a Virtual Private Network (VPN)? <https://www.hotspotshield.com/resources/what-is-a-vpn/>

VPN Breakdown

- | | |
|--|---|
| <ul style="list-style-type: none">▪ Connect easily and cheaply to the Internet▪ Create two Internet connection points▪ Encrypt traffic over the Internet▪ Pro: A lot more bandwidth cheaper▪ Con: Don't get dedicated bandwidth across the Internet▪ Only 1-2 second delay compared to private network<ul style="list-style-type: none">▪ Can be critical for some operations | <ul style="list-style-type: none">▪ VPNs not ideal for financial, medical, and other real-time operations▪ VPNs are ideal for file transfers, e-mail, and so on▪ If time is not critical, VPNs can save a lot of money▪ If time is critical, dedicated lines are recommended |
|--|---|

Another significant advantage of using VPN technology is the cost savings. It is simple and affordable to get Internet access these days, and in many cases, that is all the connectivity that a VPN client needs. A VPN tunnel is then established between two Internet connection points, and the traffic between them is encrypted.

Generally speaking, this provides organizations with more bandwidth for less money (depending on the speed of the remote Internet connection). One downside to this, however, is that there is no way to offer dedicated bandwidth over the Internet. Numerous factors might influence a VPN client connection, most of which are out of the organization's control. Another possible negative aspect of VPN connections is a 1-2 second delay in communications compared to private networks. Although this is acceptable to most organizations, some critical operations might not be able to afford the delay.

There are also some disadvantages to VPNs, the main one being the lack of performance guarantees. Most private circuits, such as leased lines or ATM, have the capability to guarantee bandwidth and latency. Similar guarantees have been difficult to achieve with VPNs. TCP/IP, the networking protocol for the Internet, was not designed to provide Quality of Service (QoS) and improvements have been slow in coming. Providing QoS for VPNs is even more difficult because many QoS solutions require the service provider to look into the messages they are passing on to decide whether the message has higher priority than other messages. If the service provider cannot examine the information in a message (because of encryption), it makes it even more difficult to decide which network traffic should get priority.

There are solutions to these problems. Multiprotocol Label Switching (MPLS), an alternative over traditional layer three routing, is used to address these problems. It allows forwarding of messages across the Internet without requiring examination of the message contents. MPLS-based VPNs can be purchased from a wide variety of Internet service providers (ISPs), although they are more expensive than standard IP services.

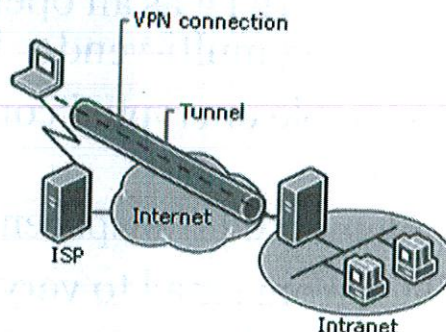
Types of Remote Access

Client-to-site VPN (transport)

- Example: Laptop connection to remote access server at HQ

Site-to-site (tunnel)

- Example: Sales office connection to HQ office location



There are two primary categories of VPNs to consider: client-to-site and site-to-site. Client-to-site VPNs provide remote access from a remote client, such as a traveling sales rep or telecommuting employee to the corporate network. Such VPNs are normally established between the client's computer and a gateway device located at the border of the corporate network. The client's computer runs VPN software that allows it to establish the connection to the VPN gateway.

Site-to-site VPNs provide connectivity to networks, such as headquarters and a remote office. In these connections, gateway devices are located in front of both networks. Information needing to flow between the sites is directed to the local gateway, which then encrypts the contents of the message and forwards it to the other site's gateway. The remote site's gateway decrypts the message then sends it onto its final destination.

There is a third, less common type of VPN, the client-to-client VPN. These VPNs establish a protected link between two specific computers. As such, they could be considered the most secure of the VPN types, because in the client-to-site and site-to-site VPNs, part of the path between the transmitter of a message and the receiver of the message is unencrypted. For example, in client-to-site VPN, the communication from the client's computer to the VPN gateway is protected, but the message travels unencrypted (and unprotected) from the VPN gateway to the internal corporate server the client is trying to communicate with. If an attacker inserts herself somewhere between the VPN gateway and this server, she would be able to eavesdrop or modify the contents of the message.

If client-to-client VPNs are more secure, why are they not used more often? The majority of the reason is the configuration required. Each pair of hosts wanting to communicate must be specifically configured to allow the communication. The most important part of this configuration is key installation. Each host must have a separate unique key that it can use to encrypt information to a particular destination host. Because of this, client-to-client VPNs between every two hosts would quickly become unmanageable as the number of hosts increases, if manual configuration is used. Public key infrastructure (PKI), which is discussed later in this module, is one way to address this key distribution problem.

References

1. VPN Service, <http://vpnserviceinfo.blogspot.com/>
2. How VPN Works, [https://technet.microsoft.com/en-us/library/cc779919\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc779919(v=ws.10).aspx)

IPsec Overview

- Issued by IETF as an open standard (RFC 2401), thus promoting multi-vendor interoperability
- Can enable encrypted communication between users and devices
- Implemented transparently into network infrastructure
- Scales from small to very large networks
- Commonly implemented: Most VPN devices and clients are IPsec-compliant

IP Security (IPsec) is an IETF standard for establishing virtual private networks. It is slowly replacing proprietary VPN protocols and becoming the industry standard. Many products on the market now support IPsec natively.

Like the application-level and transport-level techniques we discussed, IPsec provides data integrity, confidentiality, and authentication. IPsec also offers sophisticated replay attack prevention.

Attackers use replay attacks by copying a message as it goes across the network, then re-transmitting the copy to the destination. Even if the attacker cannot read the encrypted message, he can cause undesired results. For example, if the message was a request to transfer \$1,000, the replay might be able to cause an additional transfer making the total transferred \$2,000. IPsec includes specific mechanisms to detect and prevent replay. Replay attacks are often used to capture encrypted authentication sessions and replay them later to log on to a given system.

Types of IPsec Headers

Authentication Header (AH)

- Data integrity: No modification of data in transit
- No confidentiality
- Origin authentication: Identifies where data originated

Encapsulating Security Payload (ESP)

- Data integrity: No modification of data in transit
- Confidentiality: Data can be encrypted
- Origin authentication: Identifies where data originated

IPsec is actually a collection of protocols used singly or together to implement its various network security services. Primarily, IPsec is composed of two main modes: the Authentication Header (AH) protocol and the Encapsulated Security Payload (ESP) protocol. To understand how IPsec works, let's examine the abilities offered by each of these protocols.

Authentication Header (AH)

AH provides message integrity, anti-replay, and source authentication. It works by adding authentication information to each IP packet. To see how this works, we need to understand some of the information that goes into an IP packet.

IP packets are composed of many pieces of information, with each being important. One of the most important, from a security standpoint, is the source IP field. The source IP field is used to tell the recipient who sent the message. In a normal network conversation, the computer that is sending a message uses its own IP address as the source address. This is important to the security of the system because many firewall systems use source IP addresses to determine whether a message should be allowed into a network or not. If an attacker can choose to lie about his IP address, he could potentially use an address that the firewall does allow in, fooling the firewall into accepting a message that it should have denied. Without AH, there is nothing to prevent an attacker from lying about the source or any other field inside the packet.

To prevent this, AH adds a keyed hash to the packet. This hash is referred to as the Integrity Check Value (ICV). In the ICV computation, AH includes every field that does not change during its trip from source to destination. This includes the source address, destination address, length, and the data. This information is inserted into the packet after the regular IP header, but before the data.

To verify that the packet has not been tampered with, the recipient recomputes the ICV. If any of the hashed fields, including the source address, have been changed, even by a bit, the hash will be different and the integrity check fails. This provides both integrity checking and authentication. The integrity is guaranteed because the hash must match the ICV.

However, what about the authentication? Remember that this is a keyed hash. The key used is negotiated between the sender and recipient prior to the start of communications. You can compute only the hash if you know the right key. Thus, if a recipient can re-compute the hash using the key previously agreed upon with the sender, then the message has been authenticated as originating from that sender.

The algorithm used to create the ICV is configurable. The architects of the IPsec protocol endeavored to minimize any dependency between IPsec and the cryptographic algorithms that it relies upon. This is to prevent the standard from becoming out-of-date if a new cryptographic algorithm needs to be supported.

As mentioned previously, some fields have to be left out of the ICV computation because they change during transmission. An example of this is the Time to Live (TTL) field. The TTL field is used to limit how many different routers (or hops) a packet can pass through before it reaches its destination. Every time a packet arrives at a router, its TTL field is decremented. When it reaches zero, the packet is dropped and an error message is sent back to the source of the packet. You can see why this could never be included in the hash computation. This field is guaranteed to be different by the time it arrives at the recipient. The recipient's hash computation would always fail!

There is one last feature worth mentioning about AH, its anti-replay capabilities. AH uses the sequence number to determine whether a packet has been seen before. The way it works is straightforward. When an AH connection is first established, the value is set to zero. Every time a packet is sent out, the number is incremented. So, the first packet has a sequence number of zero, the next 1, and so on. To prevent replay, the receiving system must make sure that it never accepts two messages with the same sequence number.

There is an additional wrinkle to this. The sequence number is a 32-bit value. This allows for over 4 billion different sequence numbers. Although this might sound like a large number, it is not inconceivable, given enough time, for it to be exceeded.

When this happens, the protocol specifies that the current key in use should be renegotiated and that the sequence number value should be reset to zero.

Encapsulated Security Payload (ESP)

ESP is the companion protocol to AH. Like AH, it offers message integrity, anti-replay, and authentication features, but it also offers confidentiality by providing the capability to encrypt the contents of the message. Its implementation differs from AH in the area within the packet that it concentrates on. ESP does not pay any attention to the IP header of the packet. It concentrates instead on the message contents.

Just like AH, ESP is designed to minimize its dependency on any particular encryption algorithm. Each implementation must also include the NULL algorithm for both encryption and authentication. The reason for the NULL algorithm is explained shortly.

As stated previously, ESP provides confidentiality and authentication. You don't have to use both though. It is possible to use ESP to only perform authentication, or confidentiality, or both. Here's how.

When encryption is chosen, all the information in the packet above the network level is encrypted using the selected encryption algorithm. This includes the embedded protocol header (for example, TCP, UDP, and ICMP) and all of the message data. The packet is then rewritten by replacing all of the transport data with the payload field of the ESP message.

If you do not need the message to be confidential, you can turn encryption off by using the NULL algorithm. This algorithm, as you might guess from the name, does nothing to the message. When used, an ESP message is still generated and placed into the outgoing packet. The only difference is that the message data contained in the ESP payload is still in its original form (for example, cleartext).

Authentication is performed similarly to the AH protocol, by creating and then verifying an ICV. The difference is what information is included in the ICV calculation. ESP authentication includes only the information in the ESP message, so the source and destination of the packet do not enter into the calculation. It does not matter whether the payload of the ESP message is encrypted or not. The calculation is the same.

Just as with ESP confidentiality, a NULL algorithm is available for ESP authentication. This algorithm acts differently than the NULL confidentiality algorithm. When it is called, instead of returning the same message that it was presented, it returns nothing. This results in the authentication field of the ESP message being empty.

There is one caveat worth mentioning about these NULL algorithms. You can use one or the other, but not both. Using both would effectively disable ESP and for obvious reasons is not included in the standard.

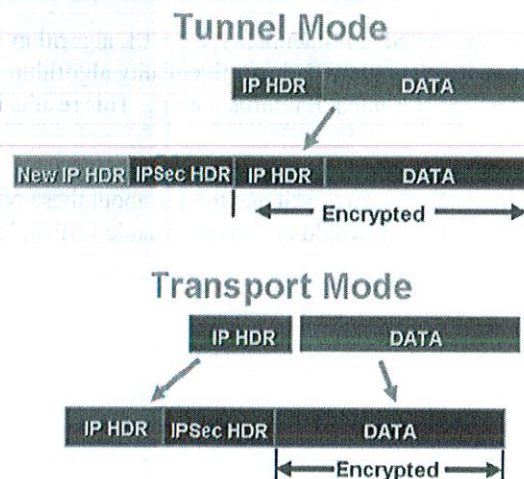
Types of IPsec Modes

Tunnel mode: Applied to an IP tunnel

- Outer IP header specifies IPsec-processing destination
- Inner IP header specifies ultimate packet destination

Transport mode: Between two hosts

- Header after IP header



Both AH and ESP can operate in two modes: transport mode or tunnel mode. Transport mode is used to protect a conversation between two specific hosts on a network. For example, two hosts using ESP in transport mode are establishing a client-to-client or a client-to-site VPN. Up to now, all of our IPsec examples have been based upon transport mode. Tunnel mode is used to establish a site-to-site VPN. Let's look at how tunnel mode differs from transport mode for both AH and ESP.

Tunnel mode, as the name implies, sets up virtual tunnels between gateways. Tunnel mode works by accepting an entire IP packet, which is then packaged in an IPsec packet. This new IPsec packet is not addressed to the destination of the packet it is carrying. Instead, its destination address is the address of the gateway system at the other side of the tunnel. When the destination gateway receives a tunnel packet, it unpackages it to get out the original packet. This packet is then routed onward to the host listed in its destination field. From this original packet's point of view, the trip across the tunnel represents just one hop, regardless of how many intermediate routers might have actually existed between the two gateways.

As in transport mode, AH provides authentication and integrity services for the packet. Implementation of tunneling mode AH is straightforward given our description of how tunneling works.

When a packet arrives at a gateway for passage across the tunnel, a new IP packet is created. This tunnel packet's header contains the source address of the gateway and the destination address of the remote side gateway. The data portion of the tunnel packet contains the original packet in its entirety.

Now, AH proceeds exactly the same as transport mode AH. An ICV is computed based upon the fields in the tunnel IP packet including the data field, which includes our original packet. The ICV is placed just after the new packet header and before the data field. When the packet arrives at the destination gateway, the ICV value is recomputed.

If it matches, it proves that the packet has not been tampered with while it traveled through the tunnel. This includes proving that the original packet has not changed, and that the fields of the tunnel packet are genuine. The gateway can now remove the original packet from the data field of the tunnel packet and send it on its way.

ESP tunnel mode works similarly to AH tunnel mode. When a new packet arrives at a gateway, it is packaged inside a tunnel packet that is addressed to the remote gateway. Encryption and authentication algorithms are then run on this new packet's data field, thus protecting the original packet. Note that this does not protect the header of the tunnel packet. The resulting tunnel packet includes the new IP header addressed to the remote gateway, and an ESP message, which includes the cipher-text and authentication data for the original packet.

There are many options available within IPsec. Before an IPsec connection can be created, the two sides of the connection must agree on what options they are going to use. In addition, many of the options require the exchange of other information, such as session keys and sequence numbers. Session establishment negotiates these details. The agreements from these negotiations are called Security Associations.

Security Associations (SAs) are a critical part of IPsec. They document the security services (called "transforms") that a particular IPsec connection is using. These details include the IPsec protocol being used (AH or ESP), the authentication mechanism that is going to be used, which cryptographic algorithm to employ, the length of the key used in the cryptographic algorithm, what security services are being applied (for example, authentication and confidentiality), and any other details necessary to fully describe the security services of the connection. Each IPsec connection must have an SA set up prior to beginning communication.

SAs are unidirectional. A single SA describes only transforms for one side of a network conversation. To establish a two-way conversation, two SAs are required: one to allow packets to be protected from point A to point B, and the second to allow packets to be protected from point B to point A. These SAs are normally set to use the same transforms, but this is not actually required.

Internet Key Exchange (IKE) is the protocol used by IPsec to negotiate the session details of a connection and then document them as SAs. IKE is a hybrid protocol composed of a key management framework and a key exchange protocol. These are the Internet Security Association and Key Management Protocol (ISAKMP) for key management and the Oakley Key Determination Protocol (Oakley) for key exchange. IKE is occasionally referred to as "ISAKMP/Oakley." Elements of a third protocol called "Secure Key Exchange Mechanism (SKEME)" are also used to extend the capabilities of Oakley.

The negotiation occurs in two phases. In phase one, a secure, authenticated connection is established to protect the conversations that occur next. This is extremely important because the security of all future conversations relies upon the capability of the two sides of the connection to privately exchange keys and other security details. Phase one provides this privacy. The results of phase one are recorded in a special SA (ISAKMP-SA) that is used only to protect ISAKMP conversations. In phase two, the security services and details for an SA are negotiated over the ISAKMP-SA.

Two methods can be used to accomplish phase one: main mode and aggressive mode. The difference between them is that main mode checks the identity of the participants, and aggressive mode does not. Identity protection sounds like a good thing and it is. So, why would we go without it? If public key cryptography is used to set up the ISAKMP-SA, identity can be inferred. If side A of a conversation can decrypt side B's messages using side B's public key, we can assume that the message was generated by B because only B should have B's private key. This provides the identity protection indirectly, making it unnecessary for ISAKMP to perform a special operation to check it.

Phase two also has multiple modes but the primary one is quick mode. This is the mode that is used to negotiate the security details for the ESP and AH SAs. This is also the mode that is used to re-key connections when the keys have been in use for too long.

SSL VPNs

- Fastest growing, have less operational problems than IPsec, cryptographically equivalent, but from an application perspective not quite as secure
- Ideal if you have multiple vendors and all you need is a browser for client side. Portal VPNs work with almost any browser. SSL Tunnel VPNs require modern browsers that can handle active content.
- Problems include opening firewall ports, application vulnerabilities, authentication, and the attack surface of the browser

If your organization is considering purchasing a VPN, don't ignore SSL-based technologies. In the past, IPsec VPNs were the overwhelmingly dominant standard. Today, SSL-based VPNs are more often the choice. They tend to have lower operational cost and fewer problems. An AES-based IPsec VPN is possibly stronger from an application security perspective; however, from a cryptographic standpoint, they are equivalent.

There are security issues related to SSL VPNs. These include the fact that you have to open ports in your firewall, probably 443 and 80. Because SSL VPNs use web technology, security weaknesses in web browsers and web servers could affect the VPN; however, if the primary purpose of the VPN is protection from casual eavesdropping, SSL is sufficient. SSL portal VPNs work with essentially any modern web browser. Specifically, they work with browsers whether or not the browsers allow (or support) active content. Thus, SSL portal VPNs are accessible to more users than SSL tunnel VPNs.

An SSL tunnel VPN allows a user to use a typical web browser to securely access multiple network services through a tunnel that is running under SSL. SSL tunnel VPNs require that the web browser be able to handle specific types of active content (for example, Java, JavaScript, Flash, or ActiveX) and that the user be able to run them.

The "tunnel" in an SSL tunnel VPN is both similar and quite different from the tunnels seen in typical IPsec VPNs. The two types of tunnels are similar in that almost all IP traffic is fully protected by the tunnel, giving the user full access to services on the network protected by the VPN gateway. The tunnels are quite different in that SSL/VPN tunnels are usually created in SSL using a non-standard tunneling method, whereas IPsec tunnels are created with methods described in the IPsec standard.

At the completion of each session, sensitive information might remain on the user's computer in temporary Internet files. If you are purchasing a commercial SSL VPN, make sure that it has the technology to clean up after sessions.

If you are considering the purchase of SSL VPNs, consider these procurement requirements:

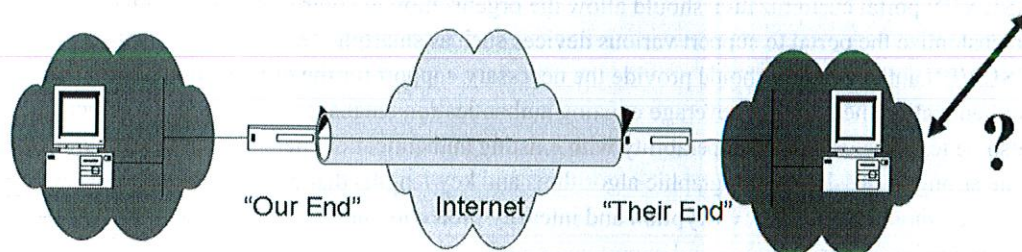
- SSL VPN manageability features such as status reporting, logging, and auditing should provide adequate capabilities for the organization to effectively operate and manage the SSL VPN and to extract detailed usage information.
- The SSL VPN high availability and scalability features should support the organization's requirements for failover, load balancing, and throughput. State and information sharing is recommended to keep the failover process transparent to the user.
- SSL VPN portal customization should allow the organization to control the look and feel of the portal and to customize the portal to support various devices such as smartphones.
- SSL VPN authentication should provide the necessary support for the organization's current and future authentication methods and leverage existing authentication databases. SSL VPN authentication should also be tested to ensure interoperability with existing authentication methods.
- The strongest possible cryptographic algorithms and key lengths that are considered secure for current practice should be used for encryption and integrity protection unless they are incompatible with interoperability, performance, and export constraints.
- SSL VPNs should be evaluated to ensure they provide the level of granularity needed for access controls. Access controls should be capable of applying permissions to users, groups, and resources, as well as of integrating with endpoint security controls.
- Implementation of endpoint security controls is often the most diverse service among SSL VPN products. Endpoint security should be evaluated to ensure it provides the necessary host integrity checking and security protection mechanisms required for the organization.
- Not all SSL VPNs have integrated intrusion prevention capabilities. Those that do should be evaluated to ensure they do not introduce an intolerable amount of latency into the network traffic.

Reference

1. Guide to SSL VPNs – NIST - <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-113.pdf>

Security Implications

- Be careful where encrypted tunnels are set up to avoid bypassing security devices



- Encryption not only stops an adversary from reading your information, but it stops you from reading an adversaries information

SANS

SEC401 | Security Essentials Bootcamp Style 104

Many sites assume that because they have established a VPN, they are secure. This is a naive assumption because VPNs bring their own special security concerns into your network. One frequent error made with VPNs is to overly trust the other side of a VPN connection.

With site-to-site VPNs, it is common to see the VPN connection allowed into the network without applying any security restrictions to it. This might be appropriate if the other side of the VPN belongs to the same organization and is controlled by the same security policies and procedures. If the other side of the connection is another organization, such as a business partner, access through the VPN should be restricted. Most VPN gateways include firewall abilities allowing them to limit network traffic across the VPN. It is a best practice to restrict this traffic to the minimum necessary to fulfill the business needs of the connection.

Another potential security problem VPNs introduce is caused by the encryption VPNs use to protect the messages they exchange. As mentioned previously, this encryption prevents an attacker from eavesdropping, but it also prevents intrusion detection systems and antivirus tools from examining the packets for malicious or inappropriate content. This reduces or eliminates the effectiveness of these security tools.

Last, client-to-site VPNs suffer from the trusted client problem. Many organizations have strict rules on the type of software allowed on corporate computers. Part of the reason for these controls is that unauthorized software might contain security vulnerabilities.

When allowing employees to use a VPN to access the corporate network, the organization might not be in the same position to dictate a tight configuration. In fact, most home computers are insecurely configured. If an attacker discovers the home computer and takes it over, he might be able to use his access to the computer to leverage access to the corporate network over the employee's VPN connection. For this reason, it is a good idea to recommend, or better yet, enforce the use of a personal firewall product and antivirus software prior to allowing remote users to access client-to-site VPNs.

GNU Privacy Guard (GPG)

The student will understand the functionality of the GPG cryptosystem and how they operate

SANS |

SEC401 | Security Essentials Bootcamp Style 105

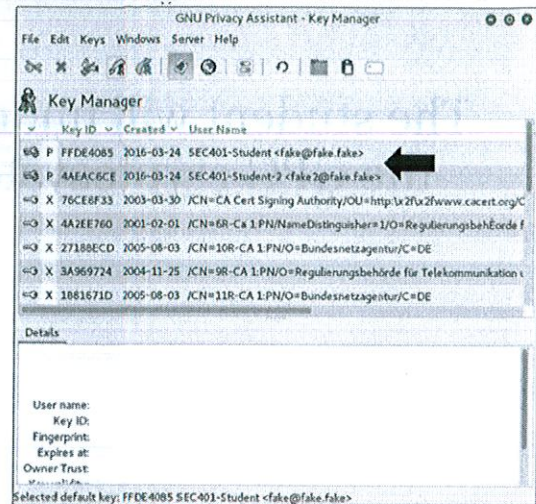
GNU Privacy Guard (GPG)

This page intentionally left blank.

Confidentiality in Storage

GNU Privacy Guard (GPG)

- Brings privacy to public communication medium via providing encryption for personal devices:
 - Protects files on hard drives
 - Protects files transferred via e-mail
 - Provides file/folder level encryption



GPG is a great example of an application-specific use of cryptography. The current version of GPG supports encryption and the creation of digital signatures for files.

GPG provides two main protections for e-mail. First, it supports strong encryption of the e-mail message. This encryption is implemented using a combination of public key and symmetric key cryptography. The second protection is digital signature of e-mail messages, providing non-repudiation and integrity verification.

GPG provides two security services for e-mail messages:

- Confidentiality through encryption
- Message integrity and source identification through digital signatures

A problem with using encryption for e-mail is that encryption requires some shared information between sender and receiver. Using symmetric key algorithms, both participants need to share a secret key. This key needs to be private to the two participants; otherwise, a third party would be able to decrypt the exchanges between them. Establishing a shared secret key prior to sending a message can be inconvenient when sending a message to someone you know, but can be impossible if you need to send a message to someone you might never have met. This makes a purely symmetric key system a bad choice for e-mail.

Public key is a better choice for this key exchange. Because public key systems separate the key into two pieces: a public piece, which you can safely distribute to the world, and a private piece, which you do not reveal, it becomes possible to exchange messages with anyone as long as both know each other's public key. Sounds better, but there is a major downside to public key cryptography. It's *slow*!

On-the-Fly Encryption (Full Disk Encryption)

- Encrypted files are decrypted to read, and then encrypted back to hard drive
- If system is turned off and computer is stolen, data is encrypted on the hard drive
- If computer is on and person is logged in, someone could potential decrypt your encrypted messages
- You should know what threat you are protecting against

When a user accesses an encrypted file, the file is decrypted for reading. When the user closes the file (in other words, it is no longer being accessed), the file is re-encrypted back to the hard drive of the system. What this means is that the system, and consequently the encryption routines employed on the system, is accessing the file based on the context of the user. If the system is turned on, and the user is logged in, the user's permissions dictate whether the file can be decrypted or not. If the user is authorized to decrypt the file, the file is readable to anyone who accesses the system while the user is logged in. If the system is turned off and gets stolen, data is encrypted on the hard drive.

Establishing a Key

Generate a public/private key pair:

- Diffie-Hellman/DSS or RSA
- Key length/size
- Key expiration

Most critical part of key generation

Use strong password principles:

- Many characters
- Mixed case, alphanumeric, special characters
- Easy to remember, difficult to guess

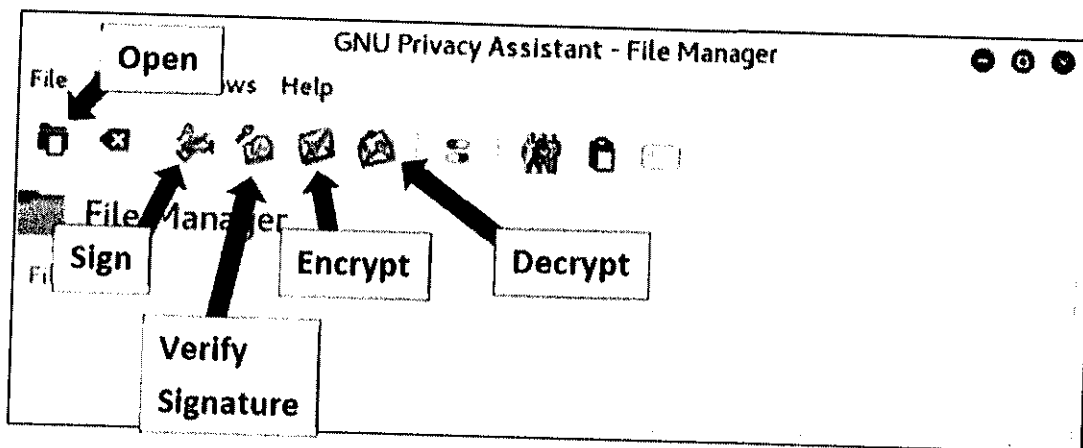
To avoid the performance penalty of public key cryptography, while still allowing its use, GPG takes a hybrid approach. It creates a random symmetric key that it uses to encrypt the message and then encrypts this key with the recipient's public key. Upon receipt, the recipient can decode the message by decrypting the symmetric key with his private key, and then use the symmetric key to decrypt the message. This provides a fast solution that allows easier establishment of trust between sender and receiver.

GPG can also digitally sign a message, verifying the integrity of the message, and the identity of who sent it. GPG digital signatures are created in a two-step process. In the first step, the information being signed is submitted to the SHA-1 cryptographic hash algorithm. The resulting hash is then encrypted using the sender's private key. The result is the digital signature, which can be sent with the original message allowing recipients to verify the validity of the message. Verification of the message is performed by decrypting the digital signature using the sender's public key to get the SHA-1 hash. A new hash is then computed on the received message and compared to the decrypted hash. If they match, then the message is genuine.

The last piece of information needed is a passphrase. The private portion of the key, which will be generated, needs to be stored on the disk of the computer that will send protected e-mails. Without additional protective measures, anyone with access to the computer would be able to copy it. Compromise of a user's private key would allow the compromiser to decrypt every message ever encrypted using the key. Because of this, GPG takes the extra step of protecting the private key, using the passphrase that you supply. The passphrase should be composed of letters, numbers, and symbols and should be fairly long. Take the time to choose a good passphrase, but take even more time to make sure you are not going to forget it. Without the passphrase, any data you have encrypted with your key will be inaccessible to everyone, including you!

Using GPG

To encrypt or sign content, it is as easy as clicking an icon



SANS

SEC401 | Security Essentials Bootcamp Style 109

GPG provides both a command line and GUI-based interface for performing encryption/decryption and signing/verification. It is an easy way to not only learn more about how encryption works but can also be used to properly protect and secure your information. Properly managing keys is also important if you are going to use GPG to communicate with other people.

GPG Functions

GPG provides 4 main functions:

- 1) Encrypting information
- 2) Decrypting information
- 3) Signing information
- 4) Verifying a signature

GPG also provides an interface for key management which is critical for performing these functions

GPG provides an easy and simple way to protect your information with encryption. It provides 4 main capabilities:

- 1) Encrypting information
- 2) Decrypting information
- 3) Signing information
- 4) Verifying a signature

All of these functions require that you have the proper keys in place. This not only includes generating your own public/private key pair but also receiving the public key from the recipient or recipients that you are going to communicate with.

Typically, when you encrypt a message, the system is generating a one-time secret symmetric key and encrypting it once with each of the recipient's public keys. This means you must have a copy of each recipient's public key that you would like to communicate with. The system will encrypt the message or file using symmetric encryption with the secret key and encrypt the secret key with each of the recipient's public keys.

To decrypt a message, the recipient would enter their passphrase to access their private key. They would then use their private key with asymmetric encryption to decrypt the secret key and the secret key with symmetric encryption to decrypt the message or file.

To digitally sign a message, the system would take a cryptographic hash of the message and encrypt it using asymmetric encryption with the sender's private key. Therefore, the sender would have to enter their passphrase to access the private key.

The recipient would verify the signature by using the sender's public key to decrypt the hash of the message or file. The system would compute a new hash and see if they match.

Public Key Infrastructure (PKI)

The student will have a high-level understanding
of how PKI cryptosystems are used for
secure communications

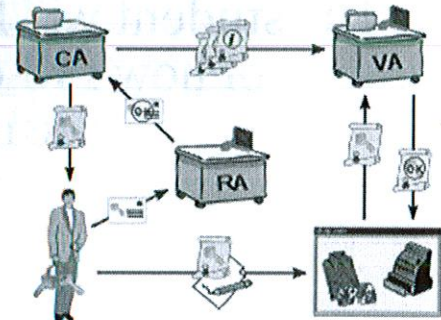
SANS |

SEC401 | Security Essentials Bootcamp Style 111

Public Key Infrastructure (PKI)
This page intentionally left blank.

What Is the Business Value of a Public Key Infrastructure?

- PKI provides a technical mechanism for encrypting an organization's data
- A hierarchy of infrastructure systems is used to create digital certificates
- Digital certificates contain the public key
- A PKI provides a managed infrastructure for
 - Creating certificates
 - Maintaining certificates
 - Revoking certificates



Public key infrastructure (PKI) is the tool most often used for e-commerce and business-to-business (B2B), and it allows users to exchange encrypted information over a public network. When you purchase goods on the Internet, you privately and securely exchange data and currency (like a credit card number), with an online vendor, through the use of a public and a private cryptographic key pair. That cryptographic public key is obtained and shared through a trusted authority.

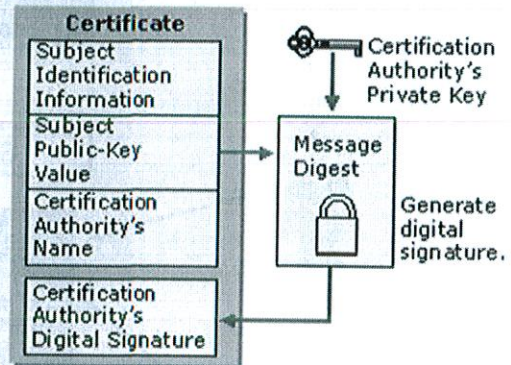
We are familiar with PKI on a public network like the Internet, but PKI can also be utilized inside of organizations. A PKI infrastructure allows an organization to create certificates to facilitate authorized access. A hierarchical certificate structure simplifies maintaining certificates and removing access when a user changes jobs or leaves an organization.

Reference

1. Public Key Infrastructure, https://en.wikipedia.org/wiki/Public_key_infrastructure

Certificates

- An essential part of PKI
- Digital document attesting the binding of an entity to a public key
- Unique to each entity
- Equivalent to a passport or driver's license
- Mitigates impersonation



The cornerstone of the public key technology is the capability to distribute public keys to large populations while conveying trust that the certificates are associated with a user and the user's public key. A certificate is the way by which trust is distributed appropriately throughout the environment.

The certificate itself is signed by a PKI authority, or certificate authority (CA), which "everyone" has agreed to trust. Everyone, as it is used with quotes in the previous sentence, means those individuals within a given operating domain, such as an industry, company, organization, or agency that have agreed upon a common law and trust system. Where there is no agreement, problems arise with interoperability of public key infrastructures, as discussed previously.

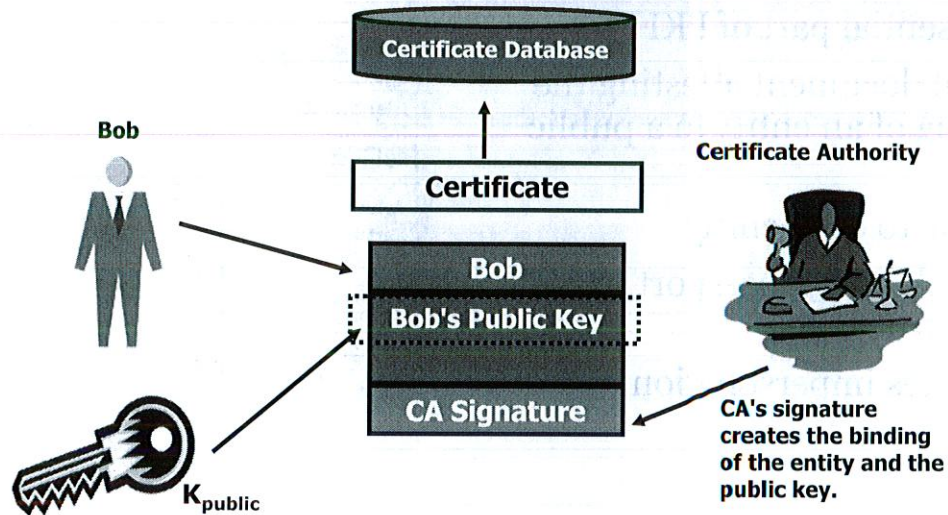
Certificates are meant to be equivalent to a passport or driver's license, at least in the domain for which it is issued. Passports are usually issued by nation states, which implies an overarching governing mechanism over a large geographic region. PKI currently does not share this type of overarching reach. In many ways, PKI represents islands of self-regulation where either a company or a collection of companies agrees on some sort of trust mechanism.

Certificates are intended to mitigate impersonation. A third party signs a user's certificate with its private key, in effect, stipulating that the third party has done a thorough background check of the entity to make sure she is who she says she is.

Reference

1. Digital Certificates, <https://technet.microsoft.com/en-us/library/cc962029.aspx>

Certificate – The Easy Picture

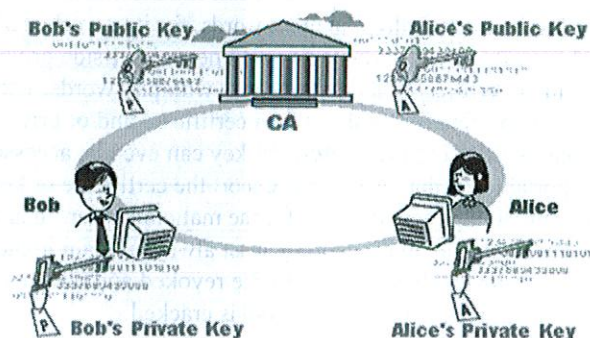


So far, we've been referring to a third party that signs the user's certificate. That third party is called a certificate authority (CA). A certificate authority has many responsibilities, not least of which is issuing certificates signed with the CA's private key. In the illustration, Bob and Bob's public key are included in a certificate. After the CA has established the validity of Bob's identity, it signs the binding, thus creating the public certificate. The certificate is typically stored in a publicly accessible directory.

Operational Goals of PKI

The traditional PKI certificate lifecycle includes

- Certificate registration
- Certificate creation
- Certificate distribution
- Certificate validation
- Certificate key recovery
- Certificate expiration
- Certificate revocation



SANS

SEC401 | Security Essentials Bootcamp Style 115

There are many steps along the lifetime of a certificate. Each is important to the maintenance of security within the PKI. On this slide, we cover

- Registration and initialization
- Creation
- Certification
- Storage
- Validation

Registration is the process that occurs before a certificate is issued. It involves the person or entity who wants the certificate providing his identification information in the form of a distinguished name (DN) and some definitive proof that he is indeed the person represented by the DN.

Next comes initialization. This step provides the person the details he needs to communicate with the PKI, including a copy of the root CA's certificate. Initialization is also where the client's public/private key-pair is generated. Depending on the policy being followed, this key generation might be performed by the person or by the CA. If performed by the person, the public key needs to be sent to the CA. If performed by the CA, the keying material (public and private) needs to be carefully sent to the person. In either case, the public key becomes associated with the person, and the person must be validated for the key to be valid.

Certification occurs when the CA actually issues the certificate, which includes the user's DN, public key, and certificate details such as validity period, protected by a signature generated by the CA. At this point, the certificate can be stored in a certificate server, such as an LDAP, or simply issued to the person to use and share as he wants. There are several facets of a key storage discussion:

- Public keys
- Private client-side keys
- Private server-side keys
- Private CA root and subordinate keys

First, public keys are just that—public. It is not only okay to share public keys, but it is encouraged. The public key can be used to determine the authenticity of messages, can serve as part of a non-repudiation scheme, and can be used to encrypt messages, which only the owner of the key can decrypt. The success of the entire infrastructure is based on the availability of the public keys, so they must be stored where everyone can get to them. It is important to note that a certificate ties a public key to an individual or a single entity so that the certificate/public key becomes an identifier. Public keys/certificates are, therefore, often stored in registries so that others can look up another user's certificate.

For PKI to be trustworthy—in other words, for it to work at all—adequately controlled secure key storage is critical for each client's private keys. As new client-side private keys are imported into a key store, users can protect their certificates and private keys with passwords. These passwords can be used simply to keep someone else from exporting (or stealing) their certificate and/or private key. Or the certificate can be stored in such a way that a password is required before the key can even be accessed and used. In fact, certificates can be stored as non-exportable, so that no one can export the certificate or key once it is installed. All of this makes compromise and masquerading more difficult for the malicious user. In any event, it is absolutely critical that a user's private keys be protected at all costs. They must always remain in the user's possession. Through revocation, discussed next, the victim can have her certificate revoked and obtain a new certificate so that the original is no longer useful to anyone, even if the password is cracked.

Users should be aware of changes in the local environment where their keys are stored. If anything causes a user to be concerned that her certificate or key might be compromised, the best solution is to have the certificate revoked and obtain a replacement certificate. Server-side private keys, such as those associated with SSL, must also be protected adequately to preserve the integrity of the messages between the client and the server. If the private key is known by another entity, it is possible for a man-in-the-middle attack to take place where the encrypted stream of data can be intercepted and decrypted by a third-party without the knowledge of the two parties who originated the conversation.

Perhaps the most important facet of key storage pertains to the private keys used to create the Root and Subordinate Certificates for the CA. The entire infrastructure becomes useless if these private keys are not carefully protected. If a CA's private keys are compromised, certificates created by that CA cannot be trusted. Anything signed by a compromised key is invalidated and unreliable.

Key escrow is the storage of keys with some trusted third-party for it to hold, in case the keys are needed but are otherwise inaccessible. This might also be referred to as “key backup.” Key escrow could also be requested by law enforcement so that it can access encrypted information as needed. Key escrow with law enforcement, therefore, is not a popular concept among civil libertarians because of the juxtaposition of public interests versus privacy and individual freedoms.

In the case of a certificate expiration, the CA need only issue a new certificate for the person. Certificate lifetime is often carefully managed and kept short. If we extend the lifetime, we increase the risk that the certificate could be compromised. By expiring certificates on a regular basis, we ensure that users who no longer need access to the data will not have that access after a specified time. In this way, the PKI system cleans up after itself in a mandatory way that cannot be altered or bypassed. Expired certificates are known by all PKI participants to be invalid because today's date is beyond the expiration date on the certificate. But what about certificates that need to be changed before the expiration date?

Certificates may be revoked for a number of reasons. Here are a few:

- User terminated from employment
- User moves to a new position no longer requiring the access provided by the certificate
- User changes e-mail address or name or other important information
- Suspected key compromise

To revoke a certificate, the CA maintains a Certificate Revocation List (CRL). The CRL consists of a list of the certificate serial numbers for all the certificates that have been revoked by the CA. This list needs to be regularly updated and sent to each of the PKI participants.

Key recovery is also an important part of many PKIs. Remember that if you lose your private key, all the information encrypted with that key is also lost. To prevent this, some CAs store a copy of the person's private key. Although this does somewhat undermine the non-repudiation of the key, it does allow the key to be recovered if the person loses it. Key recovery is particularly important in organizational settings where the information that is being protected is owned by the organization, not the individual. If the individual leaves the company, or is simply unavailable, the backup key can be used to recover the materials the individual was working on. Other reasons for key recovery include forgotten password for an encrypted file, death of an employee who has encrypted data, or someone attempting to hide criminal activity from law enforcement.

Reference

1. Enrolling for a Digital Certificate, <https://www.comodo.com/resources/small-business/digital-certificates5.php>

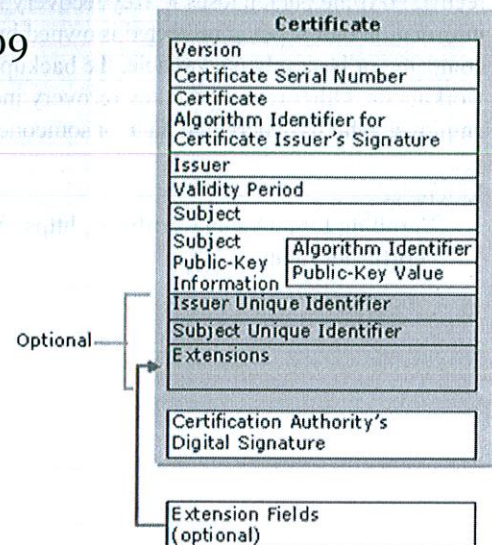
Digital Certificates

Standard for digital certificates is the x.509 certificate

Each certificate contains

- Demographic data
- Validity period
- Supported encryption algorithm
- Public/private key
- Signature by issuing CA

Public or private keys can be used for multiple forms of encryption



A digital certificate is a credential used to help someone decide whether a key is genuine. It works by binding a public key with identification information, such as name and e-mail address. This information is then signed by at least one third party. As long as you trust the opinion of one of the third parties that signed the certificate, you should be able to trust the validity of the certificate.

Digital certificates bind an individual's identity to the public key. With PKI systems, the purpose is the same, but the process used to produce the certificate is more formal. Most PKI systems do not allow the user to create certificates themselves like GPG does. Instead, a CA creates the certificate and issues it to the user. The care at which the CA performs this role directly affects how secure the overall PKI is. If the CA issues a digital certificate to anyone without requesting proof of identity, the confidence you should have in the certificate is low. If, instead, the CA requires that you show up, in person, with two forms of government-issued ID before issuing you a certificate, your confidence can be high in that CA's certificates.

Most current PKI systems produce certificates in the X.509 certificate format. This specification is published by the International Telecommunications Union (ITU), an international standards body. Most certificates follow the X.509 version 3 standard. Each X.509 certificate includes two sections: the data section and the signature section. The data section holds all the details associated with the certificate, including the following fields:

- X.509 version number
- Serial number
- Identity information of the certificate's owner in the form of a distinguished name (DN)
- Owner's public key and the algorithm used to generate it
- Period that the key is valid
- Identity information of the issuing CA

The certificate can also include other details, sometimes referred to as "certificate extensions," that are application dependent. An example is X.509 certificates used in SSL connections. With SSL, the X.509

Certificate Revocation List (CRL)

A Certificate Revocation List (CRL) is a list of revoked digital certificates

- Often due to private key compromise

CRLs have limitations

- The entire list must be downloaded each time it is updated
- CRL downloads can be network-intensive
- CRLs do not offer real-time notification of a revoked certificate

OCSP is designed to replace CRLs

A Certificate Revocation List (CRL) is, as its name implies, a list of revoked certificates. The primary limitation of a CRL is the “list” part: It is a flat document. Each time a CRL is changed, the entire list must be downloaded again in its entirety.

As a result, CRLs are not updated in real time. This opens a vector of attack: An attacker using a recently revoked certificate won’t be detected by systems using an out-of-date CRL.

Additionally, some malware blocks access to the CRL servers, as discussed next.

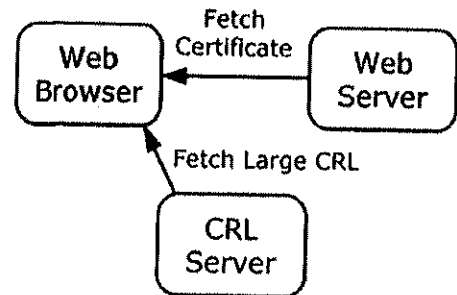
OCSP is a client/server protocol designed to overcome Certificate Revocation List limitations.

Online Certificate Status Protocol

Online Certificate Status Protocol (OCSP) is designed to overcome the limitations of CRLs

- Request status of an individual serial number
- Real-time notification of revoked certifications
- Lower bandwidth and storage requirements

OCSP is recommended by the IETF over CRL



The Online Certificate Status Protocol (OCSP) overcomes many CRL limitations. The hint is the word “online” in OCSP: It offers real-time notification of revoked certifications.

OCSP is widely supported on modern operating systems and browsers.

The Internet Engineering Task Force (IETF) recommends using OCSP instead of CRL.

Reference

1. Security Certificate Revocation Awareness, <https://www.grc.com/revocation/ocsp-must-staple.htm>

Secure Web Traffic (SSL)

- One use of PKI is to encrypt messages between a web server and a web browser
- This is accomplished by the use of either
 - Secure Sockets Layer (SSL)
 - Transport Layer Security (TLS)
- Client and server use a PKI certificate (asymmetric) to negotiate a session key (symmetric)
- PKI certificate is used for secure key exchange
- Session key is used to encrypt data between systems
- SSL/TLS is expanding today into more than websites

Cryptographic protocols can provide security and data integrity over TCP/IP networks. Two such protocols, TLS and SSL, encrypt the segments of network connections at the Transport Layer.

Secure Socket Layer (SSL)

The Secure Sockets Layer (SSL) protocol was developed by Netscape Communications in 1994 to provide application-independent secure communications over the Internet. SSL procedures are most commonly employed on the web with the Hypertext Transfer Protocol (HTTP) for e-commerce transactions, although SSL is not limited to HTTP. SSL uses cryptography to provide message privacy, message integrity, and client and server authentication, and operates on TCP port 443.

Transport Layer Security (TLS)

TLS protocol allows client/server applications to communicate across a network in a way designed to prevent eavesdropping/ tampering by providing endpoint authentication and communications confidentiality over the Internet.

Both SSL and TLS protocols provide for

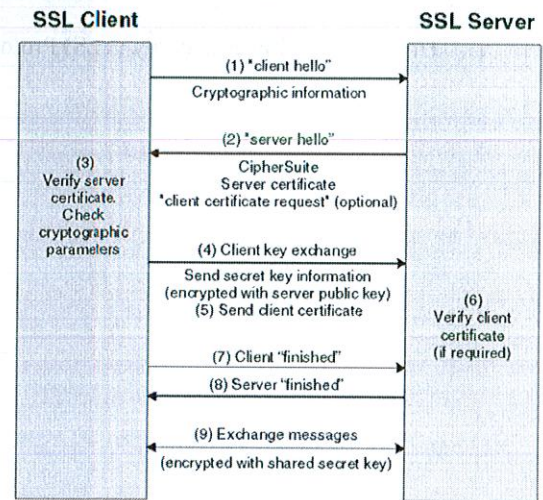
- Key Establishment
- Confidentiality with symmetric encryption
- Signature via asymmetric
- Integrity via hash

In most web-browsing situations, communications between the web browser and the server are unilateral, meaning that the client knows the server's identity, but not the other way around. The web-browser client is unauthenticated or anonymous. Server authentication means that the browser has validated the server's certificate (for example, checked the digital signatures of the server certificate's issuing CA-chain). When validated, the browser might display a security icon.

For true identification, an end user has to scrutinize the identification information contained in the server's certificate (and indeed its whole issuing CA-chain). Such a binding can be securely established only if the

PKI SSL Crypto: An Illustration

- 1) Client web request
- 2) Server responds
- 3) Client validates certificate and crypto
- 4) Client encrypts the session key and sends the session key to the server
- 5) Optional client certificate exchange
- 6) Server decrypts the session key
- 7 and 8) Key exchange finished
- 9) Encrypted messages are exchanged



At the beginning of an SSL session, an SSL handshake is performed. An HTTP-based SSL connection is always initiated by the client using a URL starting with `https://` instead of `http://`. This handshake produces the cryptographic parameters of the session.

1. **Client Web request:** "Hello, let me tell you about myself. I will tell you about my version of SSL, the cipher protocols I can support, and data compression methods I understand." The message also contains a 28-byte random number.
2. **Server responds:** "Hello there. I can understand many different cipher protocols. I will pick the one we both can understand, and a data compression method. I will also provide a session ID and another random number. I will also send you my public key, NOT my private key. You can't see that."
3. **Client validates certificate and crypto:** The client reviews the information sent by the server to authenticate the server. The client looks at the public key and sees the signature from the CA. If the server can be successfully authenticated, the client proceeds to step 4.
4. **Client encrypts the session key:** Using all data generated in the handshake to this point, the client (using cipher suggestion of the server) creates the pre-master secret for the session, encrypts it with the server's public key (obtained from the server's certificate, sent in step 2), and then sends the encrypted pre-master secret to the server.
5. **Optional client certificate exchange**
6. **Server decrypts the session key:** The server uses its private key to decrypt the session key that was generated by the client.
7. **Client ends key exchange**
8. **Server ends key exchange**
9. **Encrypted messages are exchanged:** The SSL handshake is now complete and the session begins. The client and the server use the session keys to encrypt and decrypt the data they send to each other and to validate its integrity.

Reference

1. Description of the Secure Sockets Layer (SSL) Handshake - <https://support.microsoft.com/en-us/help/257591/description-of-the-secure-sockets-layer-ssl-handshake>

Other Uses of PKI

PKI can be used for more than secure web traffic. It can also be used for

- Secure e-mail
- Partial or whole disk encryption
- Code and driver signing
- General user authentication
- IPsec and VPN authentication
- Wireless authentication
- Network Access Control/Protection (NAC/NAP)
- Digital signatures
- And much more

SANS

SEC401 | Security Essentials Bootcamp Style 125

In addition to uses for PKI such as e-mail encryption, web encryption via SSL, and disk-based encryption, there are many other uses of a standard PKI. Some of the other reasons why an organization might need to implement a PKI and issue certificates are in order to implement

- Code and driver signing
- General user authentication
- IPsec and VPN authentication
- Wireless authentication
- Network Access Control/Protection (NAC/NAP)
- Digital signatures

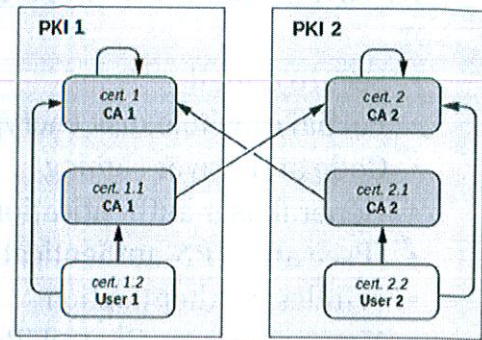
Again, as was mentioned previously, an organization needs to determine what its business goals are for implementing a PKI solution. Those business drivers are important for determining what type of PKI will be required, who will manage the PKI, and what types of certificates are issued by the PKI. Regardless of what your organization is using this for today, it should be assumed that there will be even more uses for it in the future, and it should be designed with flexibility and expansion in mind.

Reference

1. Uses for PKI, <http://www.slideshare.net/robinbasham/cryptographic-lifecycle-security-training>

Problems with PKI

- Competing/incomplete standards
- Certification of CAs
 - Important issue, but easy to overlook
- Cross-certification between CAs
- Do-it-yourself or outsource?
- Extensive planning requirement
- User education and/or perception



A basic question when establishing a PKI is who is going to use it. If only a small group is going to share a single CA, management can be relatively simple. Trying to establish a PKI for a large organization can be demanding. Extending it out to other organizations is even more so. As the number of people and groups who participate increases, so does the need for tight standardization and management, but these agreements are increasingly difficult to arrive at because there are more participants. Issues that still need to be addressed before wide-scale deployment of PKI include the following:

- **Competing standards, or standards still in flux:** Until most applications support a common PKI standard, interoperability continues to hamper large PKI deployments. Before a PKI can be useful, the applications that you rely on need to be able to make use of the PKI.
- **Certification of certificate authorities:** The policies that a particular CA uses, and how well those policies are enforced, directly affects how secure the entire PKI based on them will be. Especially when establishing common PKIs with other organizations, common certification standards need to be agreed to in order to make it possible to understand how trust between different groups should be maintained.
- **Cross-certification between CAs:** Standards for determining rules of conduct between cross-certifying CAs are still being worked out.
- **Do-it-yourself or outsource?:** It is a key question. Allowing a third party who specializes in PKI management to run your PKI infrastructure might be cost effective, but it is only possible if you completely trust the third party.
- **User education or perception:** Any large deployment of software can succeed or fail based on user reaction to the system. Because a properly implemented PKI can become essential to the operation of the entire network, it is imperative that users understand and accept their role within the PKI.
- **Lack of critical mass:** PKIs are large systems needing careful planning and deployment to succeed. Getting enough of the components established can be challenging, and the PKI is useless until they are. This can make it difficult to justify the creation of the PKI. The high cost of establishing the PKI prior to receiving any of its benefits has cooled many organizations' interest in establishing their own PKIs.

Even with these problems, it is likely that PKIs will eventually be ubiquitous. Their advantages are too clear for them to remain on the sidelines. Many organizations are working to develop technical and management standards for PKI. As these standards evolve and become more robust, the deployment risks will be reduced, encouraging pervasive use of PKI.

Reference

1. X.509, <https://en.wikipedia.org/wiki/X.509>

Applying Cryptography: Summary

- Cryptography can be deployed at many levels across a network
 - Application level
 - Transport level
- Network level (VPNs, IPsec, SSL)
- Many choices, but not all are compatible
- PKI is used to establish trust and is an important aspect of key distribution

As standards related to cryptography become more prevalent, there might come a day when almost every piece of information that is processed by a computer system is protected by some form of cryptography. In the meantime, this section looked at some of the current methods for cryptographic protection of our information systems. To organize our discussion, we showed how cryptography can be applied at several levels of a network, including the application level, the transport level, and the network level. Each of these levels brings with it advantages and disadvantages.

At the application level, each application must provide its own cryptographic services. This allows the application developers to closely match the services to the needs of the application. The downside is that each application might need to replicate similar cryptographic services and some applications might have implemented the services better than others. Still, replacing insecure applications, such as Telnet, with applications that support cryptography, such as SSH, can provide an immediate increase in security.

If your application neither supports application level nor transport level encryption, you can still take advantage of network level cryptography. Applying encryption at the network level, referred to as virtual private networking, addresses both consistency and availability issues. Any information that flows across the network can be protected. The downside here is that individual application needs might not be taken into account.

This module also included descriptions of current protocols and products that implement cryptographic security. This included a detailed discussion of how IPsec, the current standard for implementing VPNs, can be used to protect all information that flows across an untrusted network. At the application level, we described GPG, one of the deployed public key-based applications. Finally, we discussed what many consider to be a key component of cryptographic protection, PKI systems, and how they can be used to establish trust, even between people who have never before interacted with each other.

Although it will be a long time before we reach a point where all of our information assets are protected at all times by cryptography, many current applications and protocols are available that support cryptography. Using available applications and protocols, such as SSH and IPsec, can provide an immediate improvement in your organization's security.

SANS

Lab 4.2 – *GNU Privacy Guard (GPG)*

In this module, you learned about an open-source replacement for PGP, which was created under the name GNU Privacy Guard (GPG), written originally by Werner Koch, and is compliant with RFC 4880, from which PGP is based. This means that compatibility between the two tools exists, such as that with the importing of public and private keys, digital signatures, and encryption. The GNU Privacy Assistant (GPA) tool is a graphical front-end to GPG, which is natively a command-line tool. Many users find the command-line usage of GPG to be intimidating and, as such, we use GPA as a front-end GUI to interact with GPG.

Lab 4.2 – GNU Privacy Guard (GPG)

Purpose

- Learn how to utilize GPG
- Understand the operations of cryptography algorithms

Duration

- 20 minutes

Objectives

- Introduction to GPG and GPA
- Encrypting, decrypting, and signing files with GPG and GPA



Purpose

- Learn how to utilize GPG
- Understand the operations of cryptography algorithms

Duration

- 20 minutes
- The estimated duration of this lab is based on the average amount of time required to make it through to the end. The duration estimate of this lab can decrease or increase depending on various factors, such as the booting of virtual machines, the speed and amount of RAM on your computer, and the time you take to read through and perform each step. All labs are repeatable both inside and outside of the classroom, and it is strongly recommended that you take the time to repeat the labs both for further learning and practice toward the GIAC Security Essentials Certification (GSEC).

Objectives

- Introduction to GPG and GPA
- Encrypting, decrypting, and signing files with GPG and GPA

Lab 4.2 – Overview

You use your Kali Linux VM for this lab. You first cover the basics of GPG and GPA and introduce the GPA GUI. Two key-pairs have been generated for you: “SEC401-Student” and “SEC401-Student2.” You use these key-pairs to encrypt and decrypt files as well as generate and validate digital signatures. Finally, you generate a new key-pair.

You use your Kali Linux VM for this lab. You first cover the basics of GPG and GPA and introduce the GPA GUI. Two key-pairs have been generated for you: “SEC401-Student” and “SEC401-Student2.” You use these key pairs to encrypt and decrypt files as well as generate and validate digital signatures. Finally, you generate a new key-pair.



SANS

**NOTE: Please open the
separate Lab Workbook
and turn to Lab 4.2**

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.

Lab 4.2 – Exercise Takeaways

In this lab, you completed the following tasks:

- ✓ Introduction to GPG and GPA
- ✓ Encrypting, decrypting, and signing files with GPG and GPA

In this lab, you completed the following tasks:

- ✓ Introduction to GPG and GPA
- ✓ Encrypting, decrypting, and signing files with GPG and GPA

In this lab, you used the GNU Privacy Assistant (GPA) to interact and manage GNU Privacy Guard (GPG). Using these widely used tools to encrypt, decrypt, digitally sign, and verify files is a great way to reinforce confidentiality and integrity in relation to the CIA triad. Encryption provides the confidentiality, and digital signatures provide integrity and non-repudiation. GPG and PGP are fully compatible, offering flexibility when dealing with users of both programs.

SANS

Lab 4.2 is now complete

This page intentionally left blank.

SANS

Module 21: Incident-Handling Foundations

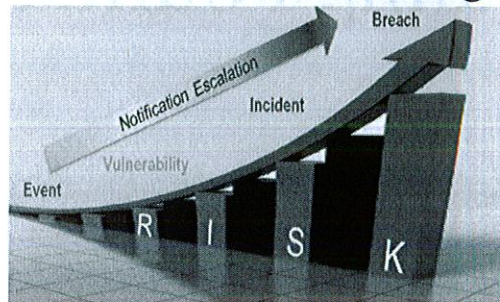


Module 21: Incident-Handling Foundations

This page intentionally left blank.

Objectives

- Incident-handling fundamentals
- Six step process for handling an incident
- Legal aspects of incident handling



A sad fact to consider is that a plethora of companies worry about their network or computer systems getting compromised, but very few have spent time preparing for that eventuality. With new security vulnerabilities released on a daily basis, we have gone well beyond the question of, "What if?" and have landed squarely on trying to answer the question of "When will we be breached?" Understanding the basis of incident-handling procedures is paramount to the success of maintaining a healthy security posture over time. In this module, we explore the fundamentals of incident handling and why it is important to your organization. We outline a six-step process to aid you in creating your own incident-handling procedures and, finally, we explain some of the laws relating to compromised systems and how you need to react to these threats in the event they become a criminal matter.

Reference

1. Incident Response, <https://www.riskbasedsecurity.com/incident-response/>

Incident-Handling Fundamentals

The student will understand the concepts of incident handling and the six-step incident-handling process

SANS

SEC401 | Security Essentials Bootcamp Style 137

Have you been hacked today? How would you know? What would you do? Imagine receiving a phone call in the middle of the night and hearing a frantic voice from your company's Help Desk shouting, "Help! We've just been hacked! A group calling themselves the 'VORTEX' just extracted 2 million financial records from our server!" As you wipe the sleep from your eyes and glance at the clock, your mind is racing for answers to questions you thought would never be asked. The voice on the other end of the line wants to know how this happened, how can he fix it, and most importantly, who is going to fix it? Obviously, this is a situation in which none of us wants to find ourselves, but chances are this will happen to you at some point in your career as an information security professional.

Incident-Handling Fundamentals

- Incident handling is an action plan for dealing with intrusions, cyber-theft, denial of service, malicious code, fire, floods, and other security-related events
- Incidents can be intentional or unintentional
- Incident response plans help to know what to do when an incident occurs

Why Is It Important?

- Sooner or later an incident is going to occur
 - Do you know what to do?
- It is not a matter of "if" but "when"
- Incident-handling plans are similar to auto insurance
 - You might not use it every day, but if a major problem occurs, you are going to be glad that you had it
- Planning is everything

Incident handling is the action or plan for dealing with intrusions, cyber-theft, denial-of-service attacks, malicious code, and other events. The scope of incident handling goes well beyond dealing with just intrusions; it covers the gamut from insider crime to anything that causes a loss of availability, whether intentional or unintentional. Natural disasters, such as fire or flood can be considered incidents because they represent a threat of harm to intellectual property or even the survivability of a company. Safeguarding intellectual property is becoming increasingly important as we move deeper into the information age. Brand names, proprietary information, trade secrets, patents, copyrights, and trademarks are considered to be extremely valuable data, and, as such, a plan is needed in the event this information ever gets compromised.

Another key point to consider is the concept of taking action during an incident. Observing an attacker in the process of defacing a web server or uploading a rootkit is not incident handling. Identifying an action is important, but you must act on that information to secure your systems in a timely manner. One of the best ways to act on an incident and minimize your chance of making a mistake is by having well-documented, proper procedures in place. Being able to rely on solid documentation on what to do when an incident occurs will help in minimizing the chance that a crucial step in the process will be overlooked or forgotten.

The size of your organization does not matter; the fact remains that sooner or later, you are going to experience an incident. In competitive markets, being prepared to handle an incident and handle it correctly could be the difference between thriving and disappearing. It might seem shocking given the widespread media attention to security vulnerabilities and exploits, but many companies have chosen to deal with an incident by simply ignoring the evidence of a security breach. The rationale seems to be: "We've never had an incident in the five years we have been in business, so why should we worry about it?" In this case, the truth of the matter is that the company probably had several incidents; however, because the company has not detected and reacted to the incidents, it has ignored the problem. It should be obvious that this mindset is very dangerous. For those companies that adopt this way of thinking, it is only a matter of time before it catches up to them.

You've probably noticed that planning is a central theme in our discussion on incident handling. It is essential to the success of a strong incident-handling foundation. On the one hand, if you are prepared and know what to do,

dealing with an incident can be fairly straightforward. On the other hand, if an incident catches you off-guard, you'll be in for a lot of sleepless nights. Although planning should be considered critical, don't get discouraged if you spend countless hours in planning and preparing for an incident and do not use those plans right away. It is easy to get discouraged and feel you have wasted a lot of your valuable time. Think of it as an insurance policy. Many of us pay our insurance premiums with the hopes of never having to file a claim; but when we do, we are happy and relieved we have that insurance!

Examples of an Incident

Which of the following is an incident?

1. An attacker exploiting Sendmail on a Unix system
2. An attacker running Active Directory scans against a Unix system
3. A missing backup tape containing sensitive information



SANS

SEC401 | Security Essentials Bootcamp Style 141

Now that you have a basic understanding of incidents and events, which of the following would you consider an incident?

- Attackers exploiting Sendmail on a Unix system
- Attackers running Active Directory scans against a Unix system
- A missing backup tape that contains sensitive information

If you answered, "Yes," to all three, then congratulations! Some might not consider the second example to be an incident because an attacker is running a Windows exploit against a Unix system, which would not be successful and, therefore, would not be a concern. In this example, however, we need to keep in mind the definition of an incident: harm or the threat of harm. Clearly, this is a potential threat of harm.

Even though this attack was not successful, a threat is still implied and there is a good chance that the next time the attacker might be successful, using a different target or set of tools.

Hopefully, it is obvious that the first and last examples would be considered incidents. One is an unauthorized exploit that allowed an attacker to gain access to a Unix system; and the other, although not as glamorous as a remote attack, is still an incident because the tape is missing.

Reference

1. The Importance of Incident Response in Higher Education, <http://ayehu.com/the-importance-of-cyber-security-incident-response-in-higher-education/>

Overview of the Incident-Handling Process: First Responder

Incident handling is similar to first aid. The caregiver tends to be **under pressure** and **mistakes can be very costly**. A simple, well-understood approach is best. Keep the six stages, (preparation, identification, containment, eradication, recovery, and lessons learned) in mind. Use **pre-designed forms and procedures**, and **call on others** for help.

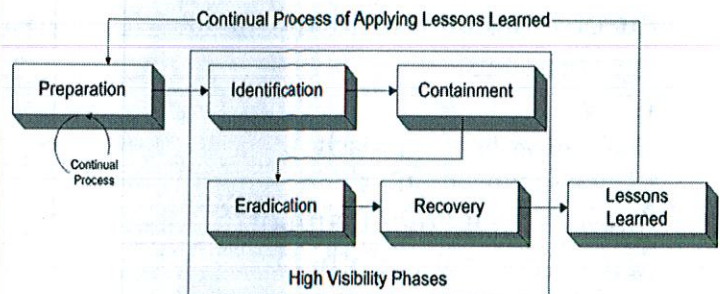
A good way to get an overview of the incident-handling process is to compare it to giving first aid. In both cases, time is not your friend. You are under immense pressure, and mistakes can be costly. That is not to say that you need to dive headfirst into a situation without thinking. Law enforcement agencies tell story after story of the well-meaning system administrator that ruined the evidence, usually within a couple of minutes after responding to the incident. You do need to act, but take the time to think things through before beginning your work. We strongly recommend that you use pre-established procedures that specify how to act during common attack situations.

As part of the incident-handling process, pre-designed forms should be used to aid in recording events. These forms provide a convenient way to document each step of the handling process and to ensure that crucial information such as dates, events, people involved, and systems affected is not missed or overlooked. Some examples of these forms include important contact information, incident survey, and incident identification forms.

As is almost always the case in legal matters, having corroborated information is better than a single source that claims the event happened. For instance, if two people witnessed a message flash on the screen, it will likely have more validity in court than if one person saw the message. In addition, attackers sometimes use tools to alter or delete their traces in log files. In this case, if you can produce two independent sources for the information, there is more validity to help discount the deleted log files.

The Six-Step Process for Incident Handling

1. Preparation
2. Identification
3. Containment
4. Eradication
5. Recovery
6. Lessons Learned



Based on the importance of incident response across the industry, it is important that a clear and standard process be followed. To create a starting point, the U.S. Department of Energy (DOE) led an initiative to build a six-step process. The six-step process used in this course and throughout the industry is based on the original process developed as part of a joint effort lead by DOE.

The six steps listed here can help serve as a roadmap or a compass, if you will, to develop a phased approach to incident handling. Keep in mind that in order for this process to be successful, each step must be followed. The six steps are:

1. Preparation
2. Identification
3. Containment
4. Eradication
5. Recovery
6. Lessons Learned

Preparation

- This is the most critical and often overlooked step
 - Out-of-band communications is very important
 - Policy:
 - Organizational approach
 - Inter-organization
 - Obtain management support
 - Identify contacts in other organizations (legal, law enforcement, partners, etc.)
 - Select team members
- Compensate team members
 - Update disaster recovery plan
 - Have emergency communications plan
 - Escrow passwords and encryption keys
 - Provide training
 - Provide checklists and procedures
 - Have a jump bag with everything you need to handle an incident

SANS

SEC401 | Security Essentials Bootcamp Style 144

The preparation step is the first and most critical step of the incident-handling process. The tasks associated with this step must be performed in advance—before the incident has occurred. This is the reason why it is often overlooked—or even skipped. It is recommended that you spend enough time preparing all the elements that are required during an incident, with the goal of increasing the efficiency and success of your incident handling efforts.

When it comes to incident handling, planning is everything, and preparation plays a vital role. It is very important to have a policy in place that covers an organization's approach to dealing with an incident. One item that a security policy needs to cover is whether a company is going to notify law enforcement officials or remain silent when an incident occurs. The answer to that might depend on the severity of the incident; if so, what guidelines should the responder use to decide whether to call? If you are going to contact law enforcement, have a list of phone numbers for each agency you might need to involve.

Another important item to consider is whether to contain the incident and move into cleanup phases or to observe the attack in an attempt to gather more evidence known as watch and learn. The policy should also contain direction for inter-organization incidents and how the company works with other companies regarding an incident.

Incident handlers can be under extreme pressure. Consider a worm that infects your entire infrastructure, effectively making your network systems unusable. This is one reason incident-handling teams must never rely on Voice Over IP. If you have a VoIP installation, consider the use of cell phones, walkie talkies, or some other backup method of communications. Incident handling can become a large-scale effort involving many people on many systems simultaneously. This should be taken into account during the planning phase.

The time to make these types of decisions is before the incident, keeping senior management and legal staff apprised of any changes to policy. Because of the sensitive nature of incident handling, any decisions made could greatly affect your career down the road if you did not get approval or reach consensus with management. The last thing you or your company wants is for senior management to question or doubt the decisions that were made during an incident.

When it comes to selecting members of the team, keep in mind that not everyone makes a good incident handler. There are some very smart people in this industry whose personalities do not lend themselves to work under immense pressure and as part of a team. People who like to work solo and need to be the hero usually do not make good team members. Ideally, a person should have a strong technical background, thrive in a team environment, be able to handle stress, and have the ability to make sound decisions grounded in reality.

As the incident response team begins to mature and has responded to several large incidents, it is possible that members of the team will get burned out and leave the team. Although this is certainly understandable, an approach you might want to take is to provide compensation and other rewards for members of the team. This might run counter to your current policies, but keep in mind that incident handlers are often called to perform their duties after normal business hours, weekends, and holidays while under a lot of pressure to get things restored as quickly as possible.

During the preparation phase, an organization should make plans to update its disaster recovery plan to include incident handling. After all, what is a disaster? It is an incident and needs to be handled as such. Although disaster recovery plans are often thought of as a checklist to get a business back up and running as quickly as possible, the skills possessed by the incident-handling team could be put to good use to reach this goal. In addition, the disaster recovery plan and the incident-handling procedures guide should contain information for emergency communications.

The issue of making privileged passwords available to others can be a delicate situation. However, in an emergency, a handler might need access to critical systems.

One idea to consider is to incorporate a procedure where system passwords are kept in sealed envelopes in a locked container. This might seem cumbersome, but it does work and keeps the passwords private until they are needed by the incident-handling team. In order for this to work, the system administrators must keep the passwords in the sealed envelopes up to date, and the incident handlers must make every effort to tread lightly on the systems, inform the system administrators of any changes made, and above all, never use a privileged password unless they are qualified on that operating system. One thing that will certainly make an incident worse is having someone who has no idea what they are doing issuing commands as administrator or root.

Our computing environments are complex and will change over time. Training is critical for each member of the incident-handling team. Memory fades over time, especially if the members are not working on honing their skills on a regular basis. Having a checklist on how to bring a system down safely or on how to restore a system can help in preventing errors and can reduce the stress on the handler. If your team is following a checklist and the resulting operation fails, it might be the fault of using an outdated checklist on a regular basis, so ensure they are updated to your organization's current environment.

Reaction time to an incident is absolutely critical. Every effort should be made to find members of the incident-handling team who will be able to respond on short notice. One way to mitigate the effects of delayed reaction is using what the military calls a jump bag. This bag should contain in a central location everything needed to respond to an incident. Items such as contact numbers, checklists, telephone, notepad, pencils, and so on, are items that you would want to include. Also, it should include, spare network cables, a hard drive, mini-hub, and tools for working on a PC should be considered essential.

Identification

- Who should identify an incident?
 - How do you identify an incident?
 - IDS alerts, failed or unexplained event, system reboots, poor performance, etc.
 - Be willing to alert early but do not jump to a conclusion
 - Look at all of the facts
 - Accurate reporting
 - Notify correct people
- Utilize help desk to track trouble tickets to track the problem
 - Assign a primary handler
 - Do not modify information
 - Identify possible witnesses and evidence
 - Determine whether an event is an incident

Some possible signs of an incident that might warrant further investigation essentially include anything suspicious, such as intrusion detection alerts, unexplained entries in a log file, failed logon events, unexplained events (such as new accounts), system reboots, poor system performance, and so on.

Being able to correctly identify an incident could be the difference between cleaning up the problem in a few minutes and causing your organization's network to be down for several hours or even days. Obviously, any system outage could potentially cost your company a lot of money, so it is important to be able to identify an incident correctly the first time and respond accordingly. For example, after a fire alarm is pulled and a building evacuated, qualified firefighters respond to the scene and investigate. Only then does the firefighter in charge at the scene authorize re-entry into the building. This should be the paradigm we work under—be willing to alert early, have trained people look at the situation, and be able to stand down quickly at a minimum of expense if nothing is wrong. No matter which course of action you decide to pursue, make certain you have mechanisms in place to correctly identify an incident.

There is nothing wrong with alerting early if you maintain situation awareness, and everyone understands this might not be an actual incident. All attempts should be made to avoid overreacting to the situation and escalating it too fast, only to realize an hour later that you made a mistake. If that happens enough times, then when a real incident occurs, no one will believe you because of the false alarms.

Chances are that your organization has a help desk operation that would be ideal for helping out with tracking the incident and maintaining a paper trail. They could also be utilized to facilitate communication and contact other personnel as the situation warrants.

It is important to keep in mind that a primary handler should be assigned as a team leader to keep the process flowing while also making sure that no steps are overlooked or missed. For smaller incidents, often of the "Would you check this out?" category, there isn't a need to send a core team of incident handlers. It is a recommended practice to have a core team of well-trained handlers and also have incident-handling skills and training as part of the job description for security officers and system administrators. An organization that adopts this approach benefits by having multiple layers of "firefighters."

However, in such a case, it is important to assign tasks in a way that encourages cooperation among the team and allows all members to succeed. When assigning tasks to part-time members of the team, do so in a way that it is clear what is expected of them: the quality of their investigation, their responsibility to preserve and collect evidence, what documentation they should produce, and when it is due. It is also important that they know who they should contact if they feel they need additional guidance or support.

After you determine that the event is actually an incident, the legal representation on the team might decide to take the steps needed to build a criminal or civil case. In this situation, witnesses should be identified, and a written statement of what they heard or saw should be taken immediately while the information is still fresh in their minds. If a decision is made to involve law enforcement, make sure senior management is notified and proper approval is received.

Containment

- The goal is to stabilize the environment
- If practical, make a disk image of the systems for analysis
 - A binary backup, NOT a full or incremental backup
- Secure the area
- Understand physical versus virtual containment
- Change passwords locally
- Work closely with cloud providers on how to handle an incident
- Utilize help desk to track trouble tickets to track the problem

Okay, we have spent countless hours preparing for the eventuality of an incident. We have a good idea of what it takes to identify an incident, but where do we go from there? Being able to identify an incident solves only part of the problem. We are still left with the task of isolating and eliminating the source of the incident. This section discusses some steps that can be taken to contain an incident and, hopefully, limit its damage to the organization.

In containing an incident, you must first secure the area. In doing so and if possible, a forensically sound backup should be made of all infected systems. If the original hard drive cannot be kept for evidence, multiple copies of the backups should be made for future analysis, if needed. One copy should be kept for evidence and the other copy used to analyze the incident. At some point in the containment process, a decision needs to be made of whether the systems should be pulled off the network or whether the entire network should be disconnected from the Internet or whether virtual containment should be utilized instead. Also, passwords should be changed as soon as possible to make sure a compromised account couldn't be used for reentry into the system by a remote attacker.

One of the key aspects of the incident-handling process is to be able to present, with a high level of detail, the different pieces of evidence found and all the actions performed during the whole process. For this purpose, you should take detailed notes of all the events associated with the incident, from the Identification (step 2) to the Recovery (step 5) phase, preferably using numbered paper notebooks.

Eradication

- Fix the problem before putting resources back online
- Determine the cause, not the symptoms
- Identify and remove backdoors
- Improve defenses
- Perform vulnerability analysis
- Make sure reinfection does not occur
- Verify that the problem has been fixed

Before the system goes back online, an incident handler must make sure that they fix the problem or the vulnerability that the attacker used to compromise the system. At first glance, the tendency might be to wipe out the entire operating system and rebuild it from scratch. Although this is certainly an effective way to remove any malevolent code, the opportunity for re-infection via the same channel still exists. There are a myriad of cases where systems were taken offline, rebuilt, and put back on the network only to be compromised again within minutes or hours. This is because a root cause analysis wasn't performed to determine why the incident happened in the first place.

It is not enough to simply recover the system and put it back online: The underlying security mechanisms of the affected systems must be altered, fixed, or upgraded to accommodate any new vulnerabilities. If it is a production system, you might hear voices of dissent from the organization about modifying a server running on a production network. This is an important, and to an extent, valid argument, but the counter is that if the system was compromised, then it must contain a vulnerability that might exist on other servers and could be exploited on a continual basis until the problem is fixed. Further, cleaning up the damage from an incident does nothing to prevent the problem from occurring again unless the problem is accurately identified and removed, patched, or otherwise mitigated.

Attackers often try to establish additional ways of ensuring remote access to the compromised system, so they have control of it even if the vulnerability exploited originally is fixed. Such backup access methods are known as "backdoors," and are implemented using several methods. Some of the most common ones include a process of listening on a specific port and offering shells access (without requiring authentication), creating a new user account with high privileges, and scheduling jobs that periodically run programs that open new paths to access the system. As an incident handler, you need to not only fix the vulnerability used during the initial system compromise, but also identify and remove every additional backdoor left by the attacker.

After the system is recovered, it is a good idea to run a vulnerability scanner against the affected system to see whether the problem is, indeed, fixed and that no new holes were opened up in the process.

To sum up, your main goal as an incident handler is to make sure that a new compromise using the same, or even a similar, vulnerability does not happen again.

Recovery

- Make sure you do not restore compromised code
 - Install from original media, add updates, and restore data
 - Restore a trusted backup patch
- Validate the system
- Decide when to restore operations (system owner or business)
- Monitor the systems closely

The key point to consider in the recovery phase is to ensure you are not restoring vulnerable code that has already proven itself to be exploitable by any number of attack methods. For example, if you restore a system from tape backup, then you could be restoring a previous state that contained the vulnerability exploited by the attacker. Vulnerable code, in this context, refers to operating system software that hasn't been patched to the latest levels, source code, and/or application software being used on the affected system. Although there is no easy solution, using a file integrity tool such as Tripwire might help in restoring the system to a known good state. Use Tripwire to take a snapshot of the compromised server, restore from tape backup, and run Tripwire again to compare the results. This method will tell you exactly what files were changed, modified, or deleted during the exploit and it gives you a better understanding of how the attack occurred and what can be done to prevent it from happening in the future.

The two main options available when restoring a compromised system are:

- Installing the operating system (OS) and applications from scratch using the official and original media, adding the latest OS and application software updates (fixing the vulnerability exploited during the incident), and finally restoring the data from a backup.
- Restoring the system from a trusted backup and patching the system, at least fix the vulnerability involved in the incident. The trusted backup already contains the latest system and application data available.

Before the system can be brought back into production, the incident handler needs to validate the system along with the system administrator. Removing the vulnerability could have affected other functions of the system that are deemed critical by the business. Anything that breaks after the recovery is likely to be blamed on the incident handler, so every effort should be made to ensure the system is working as normal before turning it over to the system administrator. In addition, the decision on when to put the system back into production has to be made by the system owner. The handler can give advice and be as helpful as possible, but, ultimately, the final decision of bringing a system back online rests in the hands of the system owner and/or administrator.

It should go without saying that if the eradication was not complete, or the infection vector was not closed off, there stands a chance of re-infection. Monitor the systems closely for the first few hours of operation to see whether anything crops up that could be attributed to the original incident.

Lessons Learned

- Identify the most relevant conclusions and areas for improvement
- Develop a report and get consensus
- Conduct lessons learned or follow-up meetings within 24 hours of the end of the incident
- Send recommendations to management, including a cost analysis

After the system has been restored and is back in operation, a report outlining the entire process should be drafted by the primary incident handler. It is very important to summarize the incident, identifying the most relevant conclusions obtained to aid in avoiding similar incidents in the future. The report should contain areas for improvement, both in the security infrastructure and in the incident-handling process itself. Additionally, the report must point out new security actions or projects identified during the incident and that must be implemented to increase the overall security of the IT environment.

The goal should be to get consensus with everyone involved. After the report has been drafted, all members of the incident-handling team should meet for a "lessons learned" overview. The goal of this meeting is to come up with a list of items that need to be included in the executive summary of the report. The executive summary should contain a brief synopsis of the entire incident, including the steps taken to recover and recommendations made.

Key Mistakes in Incident Handling

- Failure to report or ask for help
- Incomplete/non-existent notes
- Mishandling/destroying evidence
- Failure to create working backups
- Failure to contain or eradicate
- Failure to prevent re-infection
- Failure to apply lessons learned

Conducting a follow-up meeting with all involved parties is never easy, but it is vital to making sure the organization understands what happened, why it happened, and what steps were taken to make sure it doesn't happen again. During every incident, mistakes occur and there is a tendency to place blame; however, the goal of the follow-up meeting should be to improve the process and learning from the mistakes.

Some key mistakes that are common in many organizations are listed here:

- Failure to report an incident or ask for help
- Incomplete or nonexistent notes
- Mishandling or destroying evidence
- Failure to create working backups
- Failure to contain or eradicate the incident
- Failure to prevent re-infection
- Failure to apply lessons learned

Reference

1. Security Incident Planning, Stephen Northcutt

Putting the Steps Together

- Steps must be customized for your environment
- Every incident is different
- Planning is everything
- Make things simple with checklists and tested procedures
- Practice, practice, practice

It might seem obvious that the six-step incident-handling process needs to be customized by each organization to take into account the various policies, network topologies, and other aspects of operation that might affect any of the phases outlined. Every incident is different and needs to be planned for accordingly by developing checklists, getting the required training, and assembling a team that best represents the technologies used by a particular organization:

- **Preparation:** It is essential to plan for the eventuality of an incident. Remember, an incident will happen; it's simply a matter of time.
- **Identification:** This is the ability to distinguish between an event and incident. Staying current on potential vulnerabilities and exploits is a critical step in being able to identify an incident on your system.
- **Containment:** You must isolate the incident to prevent it from spreading or causing more damage to the organization. This might also involve gathering information to be used as evidence or making the decision to pull a system from the production network.
- **Eradication:** Eliminating the source or cause of the incident is an integral part of the incident-handling process.
- **Recovery:** This is the restoration of service and turning over the affected system back to the system owner or the administrator. The incident handler should take all precautions necessary to ensure the system is fully recovered before returning it back to the production network.
- **Lessons Learned:** Conducting a follow-up meeting after the incident is critical to understanding what happened and why, and to ensure that the proper steps were taken to prevent similar incidents from occurring.

To improve the incident-handling capabilities of the organization, it is strongly recommended you practice the six-step process over the production environment before the real incidents occur. Several tasks, such as penetration tests or specifically designed attack simulations, help test the incident-handling capabilities and policy. On several occasions, these types of tasks have been useful in testing the readiness of an incident-handling team to act.

Legal Aspects of Incident Handling

The student will be able to identify areas of law that are important to incident handling and understand important practices in handling evidence

SANS |

SEC401 | Security Essentials Bootcamp Style 154

Legal Aspects of Incident Handling

This page intentionally left blank.

Legal Aspects of Incident Handling

- Plans, policies, and procedures developed for incident handling must comply with applicable laws
- This is not a legal course:
 - Plans, policies, and procedures must be reviewed by legal counsel
 - You are not the expert, so work closely with the legal department or counsel

All incident-handling plans, policies, and procedures must comply with national, state and provincial laws, rules, and regulations. In Europe, for instance, although European Union (EU) Community Law has precedence in certain cases over national member state laws, member states have considerable latitude in how they adopt such laws. Adoption might vary from country to country. It is often challenging for large multi-national corporations to comply with these laws across the EU.

The two dominant legal systems in the world are Common Law and Civil Law systems. The Common Law system evolved in England over several hundred years and is often referred to as judge-made law.

Common law is supplemented by written statutes and legislation. It is the system in operation in the United States, most parts of Canada, Australia, Ireland, and the UK.

Civil law, on the other hand, comprises codified, or written laws, which are supplemented by additional written laws and legislation. The most famous "Code" is the Code Napoleon (the French Civil Code of 1804), named after its proud creator, Napoleon Bonaparte.

It has influenced the laws of Belgium, Luxembourg, Netherlands, and the old French colonies. Some countries and states (Louisiana, Scotland, and the Canadian province of Quebec) have "hybrid" systems. Some legal scholars believe that the differences between the two systems are eroding over time. There are, however, still significant differences, especially in the criminal law arena, that are best left to local lawyers to interpret.

Laws in both civil and common law systems are frequently revised and amended. Because of this, those responsible for legal action must work hard to stay current.

Incident Handling and the Legal System

Criminal Law

- Fines and/or imprisonment (global challenge)

Civil Law

- Compensation for damage (compensatory, punitive, or statutory) or loss

Others

- Regulatory laws
- Reporting security breaches, cyber-insurance, international standards, policies

As you can imagine, the security professional needs to take many factors into account when reacting to an incident; for example, whether law enforcement should be advised, whether charges should be filed, or whether a criminal offense has been committed.

Criminal law was designed to protect the public from conduct considered in conflict with certain societal norms (for example, assault, murder, rape, fraud, and more recently computer crime). Criminal law generally imposes fines and orders the confiscation of assets (for example, the proceeds of crime, or "drug money"), and/or may impose a period of imprisonment.

Certain acts may have both criminal and civil consequences. A drunk driver may be prosecuted for the crime of drunk driving and sued by the victim for damages for his/her injuries. Computer crime laws may contain both civil and criminal law penalties.

Computer crime has proven to be challenging for global law enforcement agencies because the crimes are often anonymous, hard to trace, and borderless. The criminals might reside in a jurisdiction with inadequate, if any, computer crime laws. As a result, it might be impossible to extradite them. Some computer crimes might even fall between the cracks. The law attempts, with limited success, to keep pace with evolving threats. For example, international treaties, such as the Cybercrime Convention, attempt to ensure that signatories have similar computer crime laws and that international cooperation is rendered more effective.

Civil law deals with adjudicating private disputes between parties, such as neighbors fighting over noise pollution. The Law of Torts is the area of the civil law that deals with many such disputes. A "tort" is simply "a civil wrong." The Law of Negligence forms an integral part of tort law. Generally speaking, in order to be held accountable for negligence, a party must owe "a duty of care" to the injured party; there must be a breach of that duty, and; damage must follow as a result of the breach.

In the security arena, damage resulting from a security breach can be hard to prove; so it is important to document the cost of all remedial measures, including the time/number of personnel spent on such efforts.

Sometimes, in certain egregious cases, or where the law allows it, the damages awarded may be punitive in nature—more than is necessary to restore the injured party to the position it was in before the breach.

In the event of a malware attack, a denial-of-service attack, or another attack that affects the availability of a system, or where sensitive or valuable information has been stolen, it is important to get legal advice to ascertain whether court orders can be obtained to try to trace and/or recover assets or get compensation from a defendant. Determined insiders might try to move stolen assets offshore. Involving legal counsel and law enforcement agencies in a timely manner might be of the essence in trying to recover them.

Certain sectors, such as the pharmaceutical, healthcare, and financial services sectors, have always been heavily regulated around the world because there is a greater potential for harm to the public if something goes wrong.

There are modern regulations affecting generic sectors, such as merchants dealing with credit card information. The Payment Card Industry (PCI) Data Security Standard is an industry regulation developed by VISA, MasterCard, and other bank card networks. It requires organizations that handle bank cards to conform to security standards and follow certain requirements for testing and reporting.

Traditionally, senior management has been very reluctant to report security breaches for fear of negative publicity and other adverse consequences. However, certain laws mandate that security breaches be reported to consumers in defined circumstances, usually where the exposed or lost data was unencrypted.

If competitors or foreign governments are implicated in an attack, counter-espionage laws might be relevant. Certain countries, such as Canada, Australia, and the EU member countries, have strong privacy laws that contain security-relevant provisions that must be respected. Investigations might also reveal illicit employee activity, such as the downloading and storage of illegal software, music, videos, or pornography on company property. Such activity might expose the company to liability and/or severe penalties. Hence, strong e-mail and computer usage policies are essential. All employees must be fully aware of what constitutes appropriate behavior and be aware of the consequences of non-compliance.

Criminal Law

- Victim is society
- Purpose of prosecution is punishment
- Deterrent effect of punishment
- Burden of proof is reasonable doubt
- Felonies: Jail > one year
- Misdemeanors: Jail < one year

Civil Law (Tort Law)

- Damage/loss to an individual or business
- Type of punishment is different: No incarceration
- Primary purpose is financial restitution
 - Compensatory damages, actual damages, attorney fees, lost profits, and investigation costs
 - Punitive damages: Set by jury to punish offender
 - Statutory damages: Established by law
- Easier to obtain conviction
- Burden of proof is Preponderance of evidence

There are two main categories of law: criminal and civil. With criminal law, the victim is society and to take criminal charges against someone, law enforcement must take the case. An individual or company cannot take criminal charges against someone. Criminal charges are the only laws in which someone can get jail time. With civil laws, you can get monetary restitution, but not jail time.

When dealing with law, there is a criterion that determines whether someone is guilty. With criminal law, the burden of proof says you have to prove beyond a reasonable doubt that someone committed a crime. Depending on the severity of the crime, there are different amounts of jail time one can get for a crime.

We mentioned previously that there are two types of law: criminal and civil. With civil law, you do not need law enforcement involved to take action against an individual. However, with civil law, a person cannot get jail time. A person can be ordered to pay only monetary damages. Because law enforcement is selective about which "hacker" cases it takes, it is common for a company to take civil action against an attacker if the attacker is known and there is proof the attacker caused damages to the company. In civil cases, because there is no jail time, the cases are generally easier to prove and take less time in the courtroom.

Chain of Custody

- Document (accurately) evidence items and its custody, transfer, and disposition
- Maintain a provable chain of custody
 - Attestation
 - Collect
 - Ensure evidence is auditable
 - Sign and seal

- EVIDENCE -

Investigative Agency: _____ Case No.: _____

Date of Collection: _____ Location of Collection: _____

Collector's Name: _____ Witness's Name: _____

Description of Evidence: _____

Location Where Seized: _____ Date of Seizure: _____

Seized By: _____ Seized For: _____

Seized At: _____ Seized On: _____

Seized From: _____ Seized To: _____

Seized By: _____ Seized For: _____

Seized At: _____ Seized On: _____

Seized From: _____ Seized To: _____

CHAIN OF CUSTODY

Seized By: _____ Seized For: _____

Seized At: _____ Seized On: _____

Seized From: _____ Seized To: _____

Seized By: _____ Seized For: _____

Seized At: _____ Seized On: _____

Seized From: _____ Seized To: _____

Chain of custody is a concept in jurisprudence that applies to the handling of evidence and its integrity. It also refers to the document or paper trail showing the seizure, custody, control, storage, transfer, and analysis of physical and electronic evidence.

In a criminal trial, it is usually important to prove that the chain of custody has been respected. In other words, it will often be necessary to document the chain of custody for the evidence in question from the time it was seized to the time it is sealed and subsequently presented to the court. Unexplained gaps in the chain of custody can cause serious problems because the defense might be able to argue that the integrity of the evidence cannot be assured during that time frame. It is prudent to allow law enforcement personnel do their jobs in managing the chain of custody.

When you are required, document all dates and time stamps on the items seized, and keep a record of serial numbers. Chain of custody is an important application of the rules of evidence. The methods and procedures used can affect the admissibility of the evidence collected and, although this is not generally considered a problem, maintaining good procedures ensures that any evidence gathered will be admissible in a court of law.

The first step in maintaining chain of custody is to establish the basics of the situation: who, what, where, and when. Before you touch the computer, it is a good idea to write down where you are, describe the situation, and note all serial numbers of the machine(s) in question.

After the baseline has been established, the collection phase can begin. If at all possible, a binary backup of the information should be performed to prevent any further steps from possibly weakening your case. However, as previously stated, it is highly advisable that incident plans mandate that only experienced and trained individuals conduct the forensic component of an investigation—and/or that trained law enforcement personnel do so pursuant to a valid search warrant.

The final step to ensure a proper chain of custody is to sign and seal each piece of evidence as it is collected. If the evidence is transferred to another person, it is imperative to get that person to sign off on an itemized list of all the data collected and transferred.

Evidence Integrity

```
Command Prompt

E:\>dir case_ERIC290905_disk1_image.img
Volume in drive E is DATOS
Volume Serial Number is 2533-B511

Directory of E:\

11/03/2006  01:49                665.387.008 case_ERIC290905_disk1_image.img
               1 File(s)                665.387.008 bytes
               0 Dir(s)                606.785.536 bytes free

E:\>md5deep case_ERIC290905_disk1_image.img
2c65ab703ce06daf29426dc35a4bbc64 E:\case_ERIC290905_disk1_image.img

E:\>type case_ERIC290905_disk1_image.img | md5deep
2c65ab703ce06daf29426dc35a4bbc64

E:\>sha1deep case_ERIC290905_disk1_image.img
504937b1a993f986a3023975fc9cf421f2e071f6 E:\case_ERIC290905_disk1_image.img

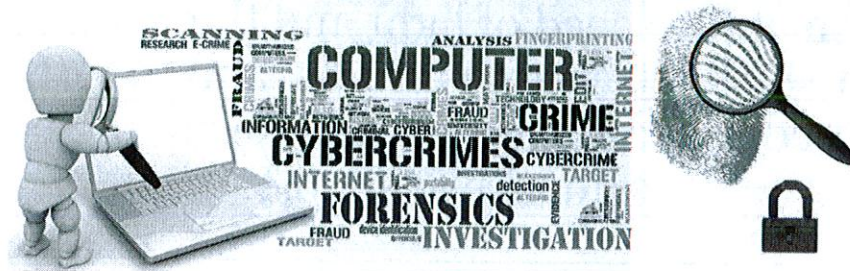
E:\>_
```

Trained forensic computer security professionals follow defined procedures that have been approved by the courts in their jurisdictions for preserving computer-based evidence. They generally take a mirror image or snapshot of the targeted media, often using commercially available forensic tools. Subjecting the image to a hashing algorithm, such as MD5, provides a value that represents the imaged data and its integrity. The idea is that if the imaged data subsequently changes in any respect, however minute (if its integrity is not fully preserved), the hash will not compute.

The process must be fully auditable, verifiable, and repeatable. Follow reasonable procedures when examining the evidence; keep good notes; and, if possible, run "script" or some other command-line history tool to demonstrate later the exact steps you took to assess the situation. The basic premise is that if you can repeat what you did during the investigation, then you are in decent shape going into court. Whenever possible, it is a good idea to run checksums to maintain the file integrity of the data being collected. MD5 is one of the most popular methods accepted by the courts and can easily be used on both Windows and Unix platforms. To validate a copy, one could obtain the hash, store it in a safe location, possibly with a digital signature, and if necessary, copy the evidence file to external media. To validate the copy was accurate and nothing has changed, run the hash on the external media and compare it to the well-known good hash.

Real and Direct

- Real evidence is the tangible item: the seized computer, the USB thumb drive, the printout
- Direct evidence comes from what the handler actually saw—not what the handler surmised



SANS

SEC401 | Security Essentials Bootcamp Style 161

When it comes to presenting evidence in court, you should present the most compelling or "best evidence" available to you. If you are fortunate enough to have real and direct evidence of the facts in dispute, your chances of success are much better than if you must rely entirely on circumstantial evidence or third-party testimony. Real evidence is often the most compelling type of evidence. It is usually a tangible object, such as the blood-stained murder weapon, evidence on a seized computer disc, or other documentary evidence that in and of itself tends to confirm the facts in issue.

Direct evidence is also a strong form of evidence. It usually refers to evidence gathered from an eyewitness or the person who watched or logged an incident as it occurred, not from someone who merely speculates as to what occurred during an attack. In cybercrimes, it can be relatively easy to demonstrate the what, where, and when of a case. It can, however, be difficult to prove the "who" and "why" behind the attack. A person might claim someone else used their password at the time of the attack and so forth. This might be a good reason to adopt a watch-and-learn approach to incident handling. Such an approach allows you to build your case rather than speculate on who the perpetrator might have been during the time of the attack.

The final type of evidence we need to discuss is hearsay or, as it is sometimes called, "third-party evidence." This is evidence that is the opposite of direct evidence. It is one party's testimony to what another party said or did. There are many exceptions to this ancient rule, which has been all but abandoned as outmoded in many jurisdictions.

Reference

1. Cyber Forensics, <http://amtagglobal.co.in/cyber-forensics/>

Best Evidence

If a tractor trailer crossing a bridge was hit by a helicopter, you wouldn't normally expect the real evidence to be brought to the courtroom. Instead, photos, models, and drawings are used. Cyber cases happen at the speed of light and there are times when screenshots, network traces, and so forth, must be used. Be ready to prove these are the best evidence available.

As stated, try to put your best foot forward in presenting evidence at trial. The "best evidence rule" in common law systems usually refers to a requirement to produce the original of a piece of evidence rather than a mere copy. In the context of computer-based evidence, the concept of originality is clearly challenging. However, most legal systems provide for special recognition of computer-based evidence. Printouts and other forms of output, like screenshots, are used to accurately reflect the data in question. Such evidence is often permissible. Therefore, in many cases, a properly constituted snapshot or mirror image of the data in question constitutes the best evidence available and should be admissible. It can be useful to have screenshots of the entire incident-handling process as a visual aid for the court, and in some cases to show context.

Summary

- **Perform all six incident-handling steps**
 - Preparation is very important
 - Continue with Identification, Containment, Eradication, Recovery, and Lessons Learned
- **You must have a basic understanding of the legal aspects of incident handling**
 - You are not law enforcement
 - You are not a lawyer
 - Do not take on more than you can handle
- **Learn from the past and keep improving your incident-handling procedures**

Summary

Incident handling can be extremely difficult, and the opportunity to make a mistake that could jeopardize an investigation is decreased if planning and preparation are taken into consideration before an actual attack occurs. Employing a six-step process that covers preparation, identification, containment, eradication, recovery, and lessons learned will aid in creating an incident-handling team that is capable of reacting quickly and accurately to any attack that might occur during its tenure. There are several laws that pertain to incident handling, and the organization must keep these laws, in mind when developing incident-handling policy and procedures.

Evidence collected must satisfy the minimum requirements of being able to prove what, where, why, and, if possible, who conducted the attack. Maintaining a chain of custody is considered crucial, and having pre-defined checklists and deploying a standard of sealing and signing evidence will help ensure evidence is not corrupted during the course of the investigation. Mistakes will happen, but when they are made, it is important to learn from them and change your plans accordingly, so the mistake does not happen again. Finally, being part of an incident-handling team is a high-pressure job where mistakes can be costly. Being able to respond quickly but accurately is considered vital, and those who are willing to make that commitment should be rewarded accordingly.



Lab 4.3 – Hashing

We previously covered the use of hashing in relation to password security. Hashing is a one-way transformation of data into a fixed-sized output, or hash, representing the integrity of that data. Each hashing algorithm has its own output length. As an example, the MD5 hashing algorithm uses an output length of 16-bytes, and SHA1 uses an output length of 20-bytes. A collision occurs in a hashing algorithm when two completely unique pieces of data are being passed as input to the algorithm result in the same hash. Increasing the output length of the hashing algorithm allows for more unique hashes and should decrease the chance of a collision if it is well-written and well-tested.

Lab 4.3 – Hashing

Purpose

- Learn how to utilize hashing programs
- Understand the operations of cryptography algorithms

Duration

- 20 minutes

Objectives

- Introduction to hashing tools and file integrity validation
- Automating file integrity checks

SANS

SEC401 | Security Essentials Bootcamp Style 165

Purpose

- Learn how to utilize hashing programs
- Understand the operations of cryptography algorithms

Duration

- 20 minutes
- The estimated duration of this lab is based on the average amount of time required to make it through to the end. The duration estimate of this lab can decrease or increase depending on various factors, such as the booting of virtual machines, the speed and amount of RAM on your computer, and the time you take to read through and perform each step. All labs are repeatable both inside and outside of the classroom, and it is strongly recommended that you take the time to repeat the labs both for further learning and practice toward the GIAC Security Essentials Certification (GSEC).

Objectives

- Introduction to hashing tools and file integrity validation
- Automating file integrity checks

Lab 4.3 – Overview

You use various hashing algorithms to calculate the hash for various files. By changing a single byte or character in the file, the calculated hash changes, demonstrating its use in file integrity checks. You then run the hashing algorithm between the trojan1 and trojan2 programs from a 401.2 lab to note the differences. Finally, you run a Python script that validates the hash for the index.html file used by the Apache Web Server on your Kali Linux VM. This script checks the integrity of the file against a stored hash every 5 seconds. When altering the index.html file, the script notifies you of the defacement.

You use various hashing algorithms to calculate the hash for various files. By changing a single byte or character in the file, the calculated hash changes, demonstrating its use in file integrity checks. You then run the hashing algorithm between the trojan1 and trojan2 programs from a 401.2 lab to note the differences. Finally, you run a Python script that validates the hash for the index.html file used by the Apache Web Server on your Kali Linux VM. This script checks the integrity of the file against a stored hash every 5 seconds. When altering the index.html file, the script notifies you of the defacement.

SANS

**NOTE: Please open the
separate Lab Workbook
and turn to Lab 4.3**

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.

Lab 4.3 – Exercise Takeaways

In this lab, you completed the following tasks:

- ✓ Introduction to hashing tools and file integrity validation
- ✓ Automating file integrity checks

In this lab, you completed the following tasks:

- ✓ Introduction to hashing tools and file integrity validation
- ✓ Automating file integrity checks

In this lab, you used the `md5sum` and `sha1sum` tools to calculate the MD5 and SHA1 hashes of a file in its original state. You then modified the file, reran the tools, and determined that the hashes changed significantly. When restoring the file back to its original state, the hash matched the original output once again. This demonstrates how hashing can be used for file integrity checking. Next, you ran a Python script used to monitor the `index.html` file used by the Apache Web Server on your Kali Linux VM. When making a change to this file, the script generates an alert due to the mismatch in the hash calculation. These tools and techniques can be used in your organization to ensure that unauthorized changes are not being made to sensitive files.

SANS

Lab 4.3 is now complete

This page intentionally left blank.

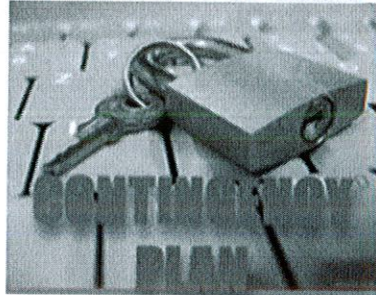
SANS

Module 22: Contingency Planning – BCP/DRP

Module 22: Contingency Planning – BCP/DRP
This page intentionally left blank.

Objectives

- Contingency Planning
- Business Continuity Planning (BCP)
- Disaster Recovery Planning (DRP)



SANS

SEC401 | Security Essentials Bootcamp Style 171

It is always important to remember that bad things happen to good organizations. Unexpected events occur all of the time and the difference between those that survive and those that do not survive is typically based on whether or not you have a contingency plan. When there is an unforeseen event, it is too late to do planning. Planning needs to be done before there is a problem, so the proper focus can be put on execution. In contingency planning, there are two key pieces: BCP and DRP. BCP or business continuity planning focuses on proactively fixing problems before they occur. DRP or disaster recovery planning focuses in on reacting to a disaster and getting the organization back to a normal operating state.

References

1. Contingency Plan Graphic, <http://securityplaybooks.com/product/contingency-plan/>
2. NIST Contingency Planning Guide for Information Technology Systems, http://ithandbook.ffiec.gov/media/22151/ex_nist_sp_800_34.pdf

Contingency Planning

The student will understand the critical aspect of contingency planning with a business continuity plan (BCP) and disaster recovery plan (DRP)

A critical aspect of your organization is planning for contingencies. In this section, we give you an overview of contingency planning—what it is and why you need it—and then we walk you through the contingency planning lifecycle. You will be equipped to create a contingency plan for your organization and to provide references for additional reading.

First, we define what a business continuity plan (BCP) and disaster recovery plan (DRP) are, and we explain why an organization needs them. Subsequently, we dive into the process for developing a BCP and DRP.

BCP Key Components

Planning

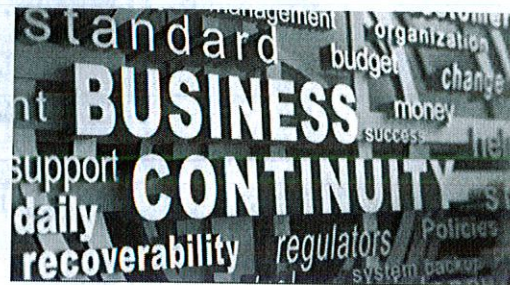
- **Assess:** Identify threats (BIA)
- **Evaluate:** Likelihood and impact of each threat

Business continuity planning

- **Prepare:** For contingent operations
- **Mitigate:** Reduce or eliminate risks

Disaster recovery planning

- **Respond:** To minimize the impact
- **Recover:** Return to normal



SANS

SEC401 | Security Essentials Bootcamp Style 173

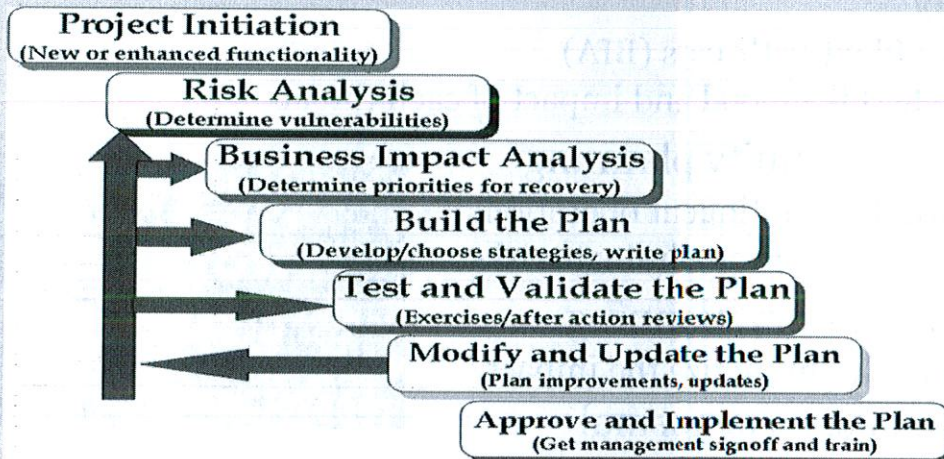
The key components of a business continuity plan are:

- **Assess:** Identify and triage all threats. This assessment is the beginning of the business impact analysis (BIA). We must understand what the threats are and assess the impact the threat would have on the business if the threat were to become reality.
- **Evaluate:** Assess the likelihood and impact of each threat. Realistically, what is the chance that the threat will happen? Perform the cost-benefit analysis to ensure any investments are justified.
- **Prepare:** Plan for contingent operations to occur within the necessary time frame. This step includes not only the preparation of the BCP, but also the ongoing management of the plan. Ensure employees are properly trained and that all documentation is in order. Perform periodic testing of the plan in accordance with your policy.
- **Mitigate:** Identify actions that might eliminate risks in advance. Are there things we can do that will decrease the likelihood of the threat becoming a reality? Are they cost-justified?
- **Respond:** Take actions necessary to minimize the impact of risks that materialize. When disaster strikes, a quick response can minimize the impact to the business. Organizations that are well-prepared are in a better position to respond quickly than those that have not thoroughly planned for disasters.
- **Recover:** Return to normal as quickly as possible.

Reference

1. The Evolution of Business Continuity Management, <https://www.johnseastern.com/the-evolution-of-business-continuity-management/>

BCP-DRP Planning Process Lifecycle



SANS

SEC401 | Security Essentials Bootcamp Style 174

This slide shows the basic steps that are necessary when developing a BC/DR plan. We start with Project Initiation, for which new or enhanced functionality is required. At this point, you must get management approval to start the project. Management is instrumental in making sure that you have access to the resources that are required to get the job done. The next sequence of steps in the process concerns the company's vulnerabilities, their significance to the company, and what the company is going to do about them.

First, the company determines its vulnerabilities through a Risk Analysis. The company then assesses the impact that each of these vulnerabilities represents for the company by completing a Business Impact Analysis. Realistically, no organization has the resources to deal with every vulnerability. Instead, in this step, the company prioritizes the vulnerabilities based on its likelihood and impact. Those vulnerabilities that represent a greater risk to the company are identified so that steps to avoid its occurrence can be planned. In the event that those plans fail, the prioritized vulnerabilities can also be given priority in terms of recovery of affected operations.

Remember, not all losses are directly associated with loss of money (although it will most likely affect the company financially in the long run). Do not forget to include the "intangible" losses, such as customer satisfaction or loss of consumer confidence. For instance, if a major e-commerce shopping site is down for a long time, consumers will become frustrated and will perhaps begin shopping somewhere else. At that point, it does not matter what caused the problem: earthquake, flood in the datacenter, or denial-of-service attack. The fact is that the site was down. The faster the company is able to recover, the better. Conversely, professional handling of a disaster can actually improve an organization's reputation with its customers and other stakeholders.

What Is a Business Continuity Plan?

- Business continuity planning (BCP) focuses on the availability of critical business processes
- Performs a strategic look across the entire business and asking what could happen
- It includes disaster recovery and business resumption planning
- It considers long-term impact to the business
- It focuses on identifying problems and proactively fixing them before they occur.



A business continuity plan (BCP) is defined as a plan for emergency response, backup operations, and post-disaster recovery maintained by an organization as a part of its security program that will ensure the availability of critical resources and facilitate the continuity of operations in an emergency situation. Business continuity planning (BCP) enables the quick and smooth restoration of business operations after a disaster or disruptive event.

The BCP is an overarching plan that details recovery from a disaster and business resumption planning, as well as a compilation or collection of other plans including:

- Disaster recovery plan
- End-user recovery plan
- Contingency plan
- Emergency response plan
- Crisis management plan
- Other plans as required (for example, a server recovery plan or a phone system recovery plan)

A BCP is a business' last line of defense against risks that cannot be controlled or avoided by other risk management practices. In addition to an immediate action plan for recovering the business, the BCP should also consider a long-term plan that keeps the business running. For example, after the company has relocated resources and established operations in a new location, how should the business work to re-establish the production site? Long-term planning should also include public relations and possibly marketing, with a plan to maintain the positive, reliable image for the company following a disaster. The BCP doesn't just define how a company should react to a disaster to keep the business operational—it must also define how the business will restore 100% of the operation including the ability to continue to meet defined business goals.

Business resumption planning (BRP) is the generic term used to refer to the actionable plan that coordinates efforts to restore an organization to normal working order. This concept encompasses a wide-scale of topics, from the immediate plans to restore business operations to long-term business resiliency planning that will help an organization maintain a polished and undeterred image for consumers, even when faced with a disaster.

Like a business continuity plan, the BRP doesn't just involve IT, it involves all levels of the organization. The best BRPs include how the organization will continue to meet and exceed the defined goals for a business following a disaster.

Reference

1. How to Build a Disaster Recovery Plan for Your Business, <https://blog.colocrossing.com/how-to-build-a-disaster-recovery-plan-for-your-business/>

What Is a Disaster Recovery Plan?

A disaster recovery plan (DRP) covers the recovery of IT systems in the event of a disruption or disaster

It consists of a tactical plan that starts immediately following a disaster

- Recovery of datacenter
- Recovery of business operations
- Recovery of business location

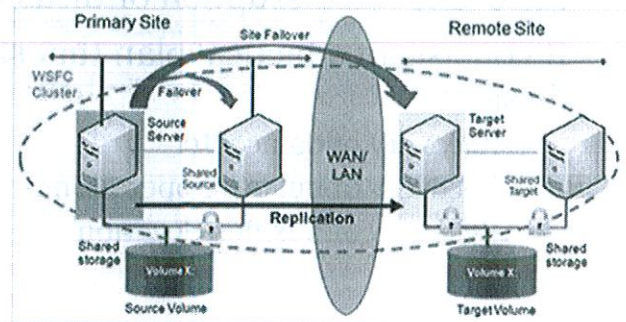
A disaster recovery plan (DRP) covers the tactical recovery of IT systems in the event of a disruption or disaster. It provides the capability to process essential organizational applications, even if they are not operating at 100% efficiency, in addition to the ability to return to normal operations within a reasonable amount of time.

The terms *BCP* and *DRP* are often used interchangeably but are actually two distinct measures that tackle different areas of the recovery process. Business continuity planning deals with the restoration of the business processes—or the continued operation of a business process: Organizational processes could operate without computers. For example, checks can be written by hand. With the continuity plan, the company can reduce the impact a disaster could have on the normal business operation. The disaster recovery plan covers the restoration of the critical information systems that support the business processes.

Disaster Recovery Planning

Keys steps of disaster recovery planning:

1. The recovery of the datacenter
2. The recovery of business operations
3. The recovery of the business location
4. The recovery of business processes



SANS

SEC401 | Security Essentials Bootcamp Style 178

Disaster recovery planning involves the following steps:

1. **The recovery of the datacenter:** Because the DRP relates to the restoration of the information systems, the datacenter is one of the critical areas the DRP should address in terms of how to bring it back on line.
2. **The recovery of business operations:** This is sometimes referred to as "user contingency planning." If a critical computer system is down, this part of the DRP deals with the alternative methods of continuing with the business operations. For instance, if your main payroll system were inoperable, a contingency plan could be to issue the payroll checks manually.
3. **The recovery of the business location:** As part of the business resumption plan, this section deals with the steps required to recover the actual physical business location. Often a disaster is partial, and recovery of the premises might consist first of patching together what is left, followed by backfilling what has been lost.
4. **The recovery of business processes:** Also part of the business resumption plan, this section handles the recovery of all of the various business processes, so that the company can resume normal business operations. This is the paramount step. The whole purpose of the plan is not about computers, networks, and data, but about the timely continuity and restoration of business processes.

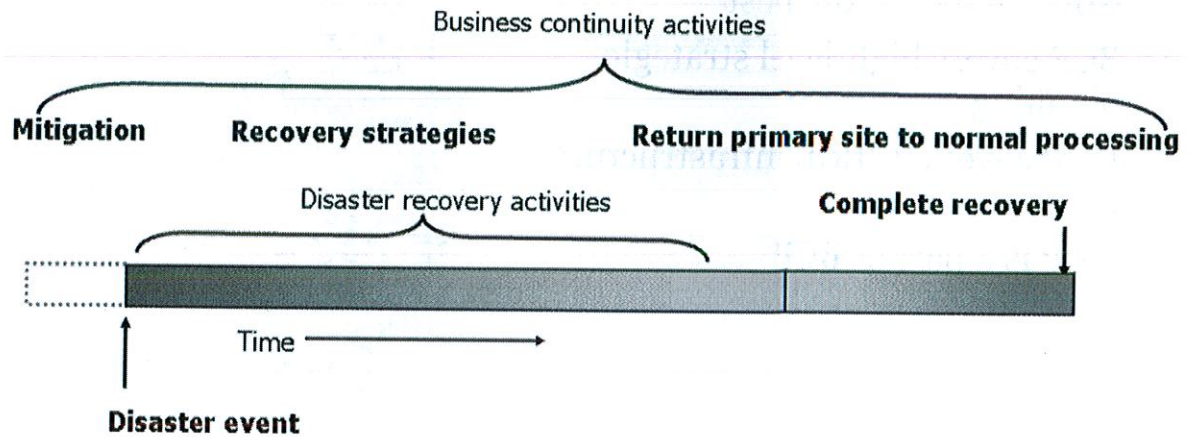
Unlike the BCP, the DRP consists of tactical action items that take place following a disaster. Where a BCP will contain high-level language that is appropriate for assessing the stability and continued operation of the business, the DRP provides clear and concise instructions that will be followed in the event of a disaster. The DRP documentation might even contain checklists for accomplished tasks as a reporting mechanism for the disaster recovery team. This type of documentation is well-suited for the basis of training materials and should be clearly written and easy to follow and understand such that there is no room for misinterpretation when following the plan. When responding to a disaster, the last thing you want is taking incorrect actions to "fix" the problem, or completing actionable items out of order.

Reference

1. Disaster Recovery Plan Template, <http://laurax2.weebly.com/>

BCP Versus DRP

Response versus recovery



SANS

SEC401 | Security Essentials Bootcamp Style 179

Disaster recovery provides a response to disruption, whereas business continuity planning implements the recovery. The preceding figure shows that the disaster recovery activities have a short time span, but business continuity activities are much more pervasive and long-lasting.

The goal of BCP/DRP is to make the response time to a disruption and the time required for complete recovery as short as possible.

During disaster recovery activities—that is, when a disaster strikes an organization—almost all normal business activities are heavily modified, reduced, or completely suspended. Only critical business processes resume, and usually at an alternate site.

As repairs are completed, normal business activities resume as the business continuity plan dictates. Recovery is complete after all normal business processes return to "business as usual."

Business continuity activities form an umbrella over a crisis situation, whereas disaster recovery activities are a *subset* of business continuity activities.

BCP/DRP

- Insurance model: Plan for the worst; hope for the best
- BCP covers high-level strategic planning
- DRP covers tactical infrastructure items
- DRP is a part of BCP



Continuity planning might be likened to insurance; it's an expense you consciously make to significantly reduce the impact of something bad that occurs. Although you pay the premium, you hope that nothing bad occurs. Even if it does not, the insurance premium and the expense of continuity planning are not wasted. They purchase certain assurances as a key component of the organization's risk management. As a wise man would say, "Plan for the worst; hope for the best."

The key component of a continuity planning is to enable your business to continue to operate. Having a split operations model allows two or more sites to actively cover one another for extended periods of time if needed. This model addresses a lot of the vulnerabilities of the Classic model where all backup materials were on-site, or nearby which doesn't help if there is a large-scale disaster.

International organizations and nationwide firms can utilize this model on different coasts and even different continents. In this way, routine workloads can be distributed among their locations. Consider two sites, one called “Germany” and one called “Australia.” Some organizations even do testing for patches and software updates on the backup site (Australia).

If it works for a week or ten days, they make the former backup site (Australia) active; Germany, then, becomes the backup site. If the new active site (Australia) stays stable while in production for a week or ten days, they then move the patches onto the current backup site (Germany). At this point, both sites are completely identical and fully capable of being a backup for one another.

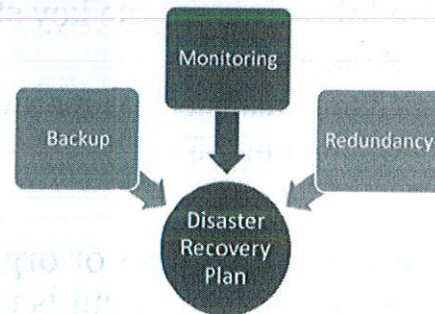
It is always important to remember that information is the lifeline of any business. If the information or access to that information goes away, the company would have monetary impact and go out of business. Although redundant sites can be expensive, it is less expensive than going out of business.

Reference

1. Disaster Recovery Plan Template, <http://www.disasterrecoveryplantemplate.org/difference-between-drp-and-bcp/>

DR Planning Process

- The following are the key steps in the DR planning process:
 - Management Awareness
 - Planning committee
 - Risk Assessment
 - Process Priority establishment
 - Recovery Strategies
 - Testing Criteria



In the following slides, we will dive more into each of these highlights of the DR planning process.

Management Awareness is critical as without management being in the know on what is going on with current risks, and resource needs, you will likely not get your project completed. This is due to lack of funds or the misappropriation of funds by you, which was not pre-approved or committed to by management. Additionally, approval by management ensures that what is in progress or proposed with the DR plan is in line with the business needs and goals.

Assembling a planning committee is not critical but is helpful, as it ensures all areas of business get a say in the importance of business processes and how the DR plan will be maintained and executed during table top exercises, dry runs and in real life. The planning committee should consist of one key person who is a stakeholder for most, if not all, of the business processes from their business department. In smaller organizations, two people working together to ensure all bases are covered may be plenty. How many members will completely depend on the number of business departments.

Completing a risk assessment and a business impact analysis is an essential part of building a successful and complete DR plan as it identifies all the assets, business processes and data and the criticality of each of these.

When the assessments are complete, the planning committee can then come together to establish the priority at which the processes will come back online in order of importance and dependence on each other.

Recovery strategies will be based largely on the scenarios which the DR plan prepares the organization for. Decisions must be made to determine which scenarios there needs to be.

Building testing criteria for the testing of the plan, which should occur as often as quarterly but no less than yearly. The testing criteria are the things which should be tested for and are metrics for measuring how successful a DR plan is being executed and assists in where the plan might need improvement.

Reference

1. Disaster Recovery Planning, <http://www.vasquezit.com/growth#disaster-recovery-planning>

Management Awareness

- It is important for management to be aware during the DR planning process
- The following are the key steps:
 - Build Awareness
 - Obtain management approval
 - Obtain funding
- Employee resources or organization capital, management and executive commitment is pivotal to accomplishing the objective

It is important for management to be aware during the DR planning process and Business Continuity Planning process. Creating a DRP and BCP does not require expending resources outside of employee time to create it. It is very likely there are areas of the organization where there may need to be resources purchased to complete a well-rounded DRP and BCP. Because of this, funding will likely be needed to complete a DRP and BCP and have it be effective for all scenarios which are most likely to occur and cause the most damage.

As with many projects, which have to do with the business and may require culture changes, employee resources or organization capital, management and executive commitment is pivotal to accomplishing the objective.

Risk Assessment

- Security is about mitigating risks the best way possible
- The following are the key steps:
 - Identifying all threats
 - Identifying all scenarios of risk to the organization
 - Mapping the risks on likelihood and criticality
 - Recommendations on how to fix them
- A risk assessment is how an organization identifies all the risks to it and documents ways in which those risks could be mitigated.

Security is about mitigating risks the best way possible. A risk assessment is how an organization identifies all the risks to it and documents ways in which those risks could be mitigated. Completing a risk assessment allows the managers and department heads of an organization to identify the information assets of the organization and what the values of those assets are to the organization.

Identifying the current assets assists in creating an overview of the quantitative value of the exposure of those assets to existing known risks, based on the existing controls in place. The risk assessment will identify whether the controls in place are sufficient for protecting the assets. This will assist in the generation of funds from management as the risk assessment will show the risk levels discovered and show the areas of concern and where resources should begin to be applied to address those issues.

In regards to the DR, a risk assessment will help identify areas of concern which will apply directly to a DR scenario and how effective a DR plan would be if executed.

Scenario Identification

- Scenario identification is something completed during a risk assessment
- Focused on “What If?” scenarios
- Can be performed with “Quick Table Tops”
- Most likely and damaging scenarios should be considered in DR plan
- Verify that the risk is covered to the best of the organization's ability

Scenario identification is something completed during a risk assessment as well as the DR planning phase. Once you have an understanding of all of the business processes, the assets and the storage and flow of critical data through the organization, scenario identification is literally like playing the what if game. Saying what if this given situation, and then during a very quick run through of how the organization would respond today given the current resources and setup and what would be the estimated impact and what is the likelihood of the scenario?

The scenarios which are most likely to occur are the ones which should be definitely considered in creating DR plans for to ensure the risk is covered to the best of the organization's ability.

Business Impact Analysis

- Determine what the critical business process are
- Identifying all threats to business functions
- What is the level of impact
- How much impact would there be
- How long can the organization survive
- What are methods of mitigation that can be put in place

A business impact analysis is what the name implies, it is an analysis of the impacts to business. It is often a report which lists out all the identified areas

In the process of developing and or maintaining a BCP/DR plan, it is vital that a business impact analysis is completed regularly, at least yearly, if not more frequently. The business impact analysis gives management a resource which shows them how they can prioritize resources based on the value of the assets and processes within the organization. It also identifies what the impact would be to the organization if those assets or processes were not available.

Performing a Business Impact Analysis (BIA)

- Determine the maximum tolerable downtime (MTD) for any given system
 - How long can your systems be compromised?
- BIA is useful when developing DRP
- BIA evaluates the effect of a disaster over a period of time
- It builds on the risk assessment results: What bad things could happen and what is the impact?

The business impact analysis (BIA) documents what impact a disruptive event might have on an organization. The BIA prioritizes business functions versus risks to identify the criticality of functions and the timeframe for which they must recover. Some business functions, if down for only a few seconds, might dramatically and detrimentally impact the business. Other business functions might be interrupted for days or weeks and have no negative effect on the corporation. From a big picture BCP perspective, the BIA helps us focus on those areas of our business that must have priority when recovering from a disaster.

The primary goal of the BIA is to determine the maximum allowable downtime for any given system, or maximum tolerable downtime (MTD). Understanding the MTD for business processes is mandatory before designing your disaster recovery plan. Without the MTD calculation for systems, you won't know whether the plan meets or exceeds the requirements of the business. Although exceeding business requirements is usually a good thing, the cost of doing so might not be.

Key Business Impact Analysis (BIA) Questions

Some of the key questions might include:

- What would be the impact of an information technology failure on cash flow and revenue generation?
- Would the disaster impact the level of service?
- How long could the outage last before it begins to affect your productivity?
- Would there be irretrievable loss of data?
- What are the key resources that are required to continue to operate?
- At what point would those resources need to be in place?
- How does this process/system interact with other processes and systems?
- What are the dependencies on this process?

The process of developing the BIA typically involves interviewing the various key users of the various computer systems (for example, payroll, accounts payable, and accounting) to get a better understanding of how a disaster could impact the ability to continue operations. Some of the key interview questions might include:

- What would be the impact of an information technology failure on cash flow and revenue generation?
- Would the disaster impact the level of service?
- How long could the outage last before it begins to affect your productivity?
- Would there be irretrievable loss of data?
- What are the key resources that are required to continue to operate?
- At what point would those resources need to be in place?
- How does this process/system interact with other processes and systems? What are the dependencies on this process?

When putting together the BIA, the answers should come from, or be concurred by, executive management. At that level, management understands cost tradeoffs such as between mitigation and loss and has individual accountability either way. Lower management might err toward too much (for example, too expensive) risk avoidance, whereas upper management might prefer to accept certain risks and redirect mitigation resources elsewhere in the business. This is a common mistake in BCP/DRP planning.

Critical Application Analysis

- Identify critical applications
- Identify critical business processes
- All applications which are important to the business operation should be documented
- Interview all key people for each known business process
- Find new processes and applications running and all associated documentation

When completing the risk assessment and the business impact analysis, there is a very good chance the identification of the critical applications have already occurred. However, it is a good step to deliberately go out and identify any and all applications in use and how they are in use within each business department. This is important because depending on the size of the organization, there is a decent chance applications were missed during those previous assessments.

All applications which are important to the business operation, no matter how small, should be documented. Often times it is the odd application and process which only happens once a day or has a very narrow purpose for the organization which is missed or is not really documented or thought of often by anyone, especially if it is running as an automated scheduled task. It is important to ensure these applications are identified, so their criticality can be rated as it is quite possible these applications may be a show stopper if they quit working in the event of a DR. Even if there is no documentation of their existence, but someone knows of the application presence and purpose, but no one knows exactly how the application and process was running or setup. This happens all too often in an organization and is commonly overlooked.

Literally interviewing all key people for each known business process to find new processes and applications running and all associated documentation is a critical part of the process.

Recovery Window

- Mean downtime is the average time that a business process or information system is no longer functioning
- Determine acceptable mean downtimes
- Each business function has different windows
- A recovery point objective is the maximum period of time where data could be lost

Mean downtime is the average time that a business process or information system is no longer functioning. The mean downtime takes into account all the time it takes to complete repairs or any maintenance needed to bring the system or process back online. Essentially, mean downtime is the average time which management has determined to be acceptable before the risk of too much damage will have occurred to the business. This is the point of no return and the organization will not be able to return to normal operations and possibly shut down.

The mean downtime, when defined at an acceptable duration, is very similar to the recovery time objective also known as the RTO. This is the targeted duration of time within which a business process or information system must be restored after a disaster scenario to prevent unacceptable consequences, such as loss of data or income.

A recovery point objective, also known as the RPO, is determined generally during the business continuity planning process. A recovery point objective is the maximum period of time when data could be lost from an information system during a disaster scenario and the information system is not in production.

Reference

1. Recovery Point Objective, http://wikibon.org/wiki/v/Recovery_point_objective_-_recovery_time_objective_strategy

Information System Contingency Plans

- Defining recovery plans for each information system
- Focuses on information systems or business functions or processes
- Information system contingency plans are the instructions for an information system and how it runs
- It also addresses how the information system starts up and interacts with other systems

It is important to clearly define information systems during contingency planning. Though it is not always required, even for small organizations, it makes the BCP and DR plan document more scalable and usable. Instead of having instructions on the contingency plan for an information system within the DR plan itself, you simply reference those instructions for the information system recovery. In the event of updating the instructions, will be the information system contingency plan which allows for the dividing of responsibilities and tasks in the event of a DR scenario.

Information system contingency plans are the instructions for an information system in how it runs and how the information system starts up and interacts with other systems. It details what is required for the information system to successfully run. It does not need to be detailed as to what needs to be done in the event of specific scenarios unless there are special tasks which need to be done.

If the hardware is special, it is not a bad idea to include in the information system contingency plan the contact information for the vendor and any sales channel contact information. Vendor contact information will be a part of the DR plan as well, allowing for hardware to be ordered if needed.

References

1. Information System Contingency Plan Template, www.va.gov/PROPATH/.../information_system_contingency_plan_template.docx
2. Contingency Plan Template – HHS.gov, www.hhs.gov/...Contingency-Disaster%20Recovery%20Plan/eplc_contingency

Define Reconstitution of Business

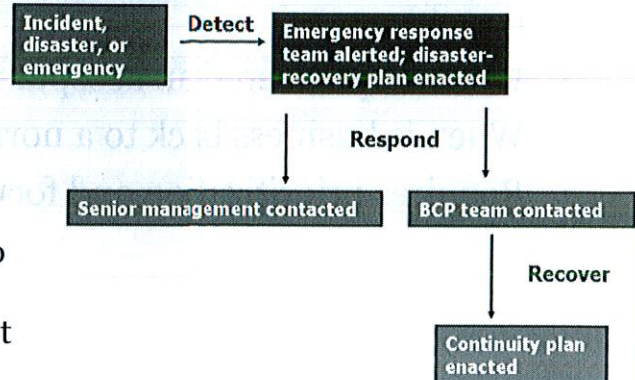
- Full resumption of business processes may not always be possible
- What is considered an acceptable level of business resumption
- When is business back to a normal function
- Requires prioritization and focus on critical business functions

Full resumption of business processes may not always be possible or it might not be possible within the recovery window executives and management would like. As a result, a definition of reconstitution of business should be defined, so there is a known goal which must be hit to know when the organization is operating at a level where the critical business functions are working. This also includes that other processes are either not fully needed to operate or there is stop gap solutions for the organization to get by until the organization is running back to full business as usual.

An example of this could mean that the main business location was destroyed along with the main data center, but a backup data center and a backup workspace are available. However, there is only enough room in the backup facility for the minimum number of staff to work and complete the basic functions of the business. All other functions are on hold or those individuals, if possible, can work from home. It is only until all business functions are fully running and there is little to no hindrance can it be considered back to business as usual. Until then, a definition of what it means to have the business reconstituted for your organization must be documented, so it is known when that has been achieved.

Communication Plan

- Having a plan in place for communication is very critical
- The plan should cover the following:
 - Call Tree
 - Out of Band communications
 - What should be communicated to whom during the scenario
 - Who are the key players that must be informed



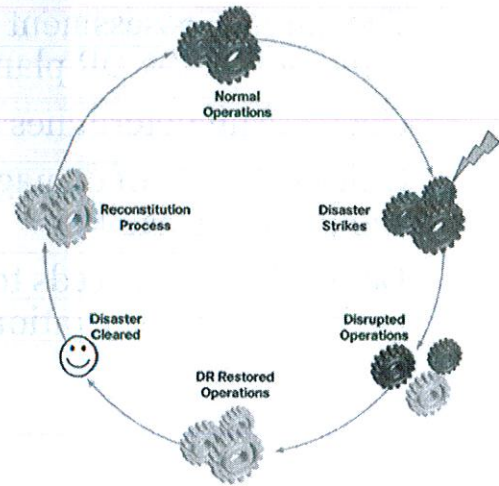
Having a plan in place for communication is very critical to the success of any BCP and DR plan. At the very minimum, there should be a list of all the people who need to be involved during the DR scenario and what their contact information is, such as phone number and alternate phone number. Follow this with a call tree so there is a hierarchy of who should be contacted first, second, or third so those who need to know have priority of knowledge. It is also important that there are backup people to reach in the event the primary individuals are unavailable.

In the case of a damaging breach or a DR scenario, where part of the traditional communication methods is unavailable, all the out of band communication methods which are available to use should be documented. This will include the ability to transmit documents themselves and such as in the case of a breach, using e-mail or file transfer programs is not ideal and may simply be not possible. Therefore it is important to have an alternative, such as faxing needs to be documented in the communication plan.

Not everyone must know everything about what is going on, so it is important to include in the communication plan the hierarchy of communication and the frequency at which updates of the situation are given to the people within the hierarchy.

Notification and Activation

- When should the plan be activated?
- Clear control gates and metrics must be identified
- Which part of the plan should be activated?
- Who should be notified?
- When should they be notified?



The BCP and DR plan should have within it identifiers of known scenarios, where it is defined when the BCP-DR plan should be activated. These qualifiers should be defined and even then, the activation of the plan generally cannot be activated without the approval of the management or executive staff. An exception would be if they cannot be reached and activation needs to happen to reduce damages. In this case, the BCP coordinator or a member of the emergency management team will activate the plans.

The part of the plan, which will be activated, will be whichever part is needed given the scenario at hand. The DR plan as a whole will always be initiated to determine which areas are affected and which information systems or process need recovery.

Notification process and procedures should always be aligned and defined in the communication plan. Who should be notified and when, including executive or management, should all be defined. This includes the predetermined frequency in which they are notified. Notification should be performed by the team lead of the emergency management team. The notification plan spells out the chain of communication on who reports to whom, the frequency of reporting and what needs to be reported. Frequency should be often enough that the individuals who need to know what is going on to assess the situation and potentially shift efforts if necessary, but not so frequently that it is hindering the recovery teams efforts in progress.

Reference

1. Disaster Recovery Best Practices,
http://www.cisco.com/en/US/technologies/collateral/tk869/tk769/white_paper_c11-453495.html

Damage Assessment Procedures

- The damage assessment should be done within the initial activation of the DR plan by the technical recovery team
- Onsite team determines overall damage
- Quick estimate of damage to determine to what level the BCP DR should be engaged
- Determine what needs to be ordered and done to bring business back to normal operations

The damage assessment should be done by the technical recovery team within the initial activation of the DR plan. They assess all of the damage which has occurred, what business processes and information systems are affected, and what it will take to bring them online again.

Initially, the damage assessment is done rapidly, within the first 30 to 60 minutes, to give rough estimates for the management team to begin working with; then, they also recommend a course of action. While the initial recovery begins, a more in-depth damage assessment is done, which could take several hours (or sometimes days), depending on the size of the organization and the number of systems affected.

The in-depth analysis of the damages incurred will tell management the cost to bring the operations back online by the recovery time objective. The procedures to complete the damage assessment involve taking the current list of assets and seeing which ones are offline. This includes presenting a list of the offline systems and business processes, along with the list of hardware that needs to be ordered.

DR Site Planning

- There are three different kinds of DR sites, hot, cold and warm sites
 - Hot site – fully redundant operations
 - Warm site – infrastructure and equipment, but not live copies of the data
 - Cold site – building, but minimal to no equipment
- Balance distance vs. recovery time

Mirrored Site	Hot Site	Warm Site	Cold Site
Instantaneous to 30 seconds	30 seconds to 30 minutes	30 minutes to 72 hours	Greater than 72 hours
Zero No data loss	Zero No data loss	> Zero Some data loss	> Zero Significant data loss

There are three different kinds of DR sites, hot, cold and warm sites. A cold site is a site which is simply a space available for use with no hardware and limited to no hookups ready. It is essentially an empty office space or any other kind of commercial space where information systems can go and people can work, but nothing is set up.

A hot site is a site which has everything prepared, all hardware is hookup and actively synchronized with production so if a failover is needed, all operations can failover and function right away and the site is ready for employees to start working.

A warm site is essentially the middle of hot and cold, hence, its name. It's a site having some infrastructure, such as racks, servers, power hookups, desks, and computers but infrastructure is not necessarily all there. A failover is not possible since the data and processes are not synchronized between the production and the DR location.

Each one of these sites has an associated cost and you can have more than one kind. These days, with cloud services and hosting providers, it is not unusual for information systems to have a warm and a hot site. A warm site would be for the employees to work, if not working from home; and, the hot site is covered by the hosting or cloud providers since these providers generally have DR hot sites built-in or are easily setup. A hosting provider and a cloud provider likely have multiple locations.

When dealing with a cloud or hosting provider, it is likely your organization does not need to worry about DR site locations, but if you are not using either for all your information systems, you will need to consider site locations and take into account the distance from the current location. Having a DR site is one thing, but if they are too close, it doesn't matter.

It is recommended to keep the DR location as far as possible away from the production location. For staff to be able to work, having a DR location might need to be inside city limits for them to be able to go there. However, for the information systems, since they could be anywhere, it is a good idea to place your DR site in another geographic location.

Reference

1. Which type of backup site is right for your organization's disaster recovery needs,
<http://gigaroom.blogspot.com/2013/05/which-type-of-failover-site-is-right.html>

Exercise the Plan

- Your organization should be exercising the DR plan
- Run table top exercises
- Create runbooks
- Update the plan
- It is always ideal to complete DR plan exercises more frequently than once per year, especially if the environment is changing often due to upgrades, new deployments and business process changes

Your organization should be exercising the DR plan. It is not always the most exciting activity and is often not given the priority it needs, but it must be done. There are many ways to exercise the plan to gain value and insight into how well the plan is built and to identify if there are any gaps which need to be addressed and updated within the plan.

The most obvious way to exercise the plan is to complete a dry run of the plan. Often times, this is not possible as it interrupts business too much to be worth doing and as a result, cost prohibitive to fully do. Partial dry runs, where you test parts of the DR plan, such as the individual information system, DR practices, or business processes, are always an option. It is recommended that completing a dry run of the DR plan, whether full or partial, be done at least once per year.

It is always ideal to complete DR plan exercises more frequently than once per year, especially if the environment is changing often due to upgrades, new deployments, and business process changes.

Run books should be created before the DR plan exercise as this is what the DR team will be using and updating during the exercises. Run books are often a part of the information system contingency plan document. If runbooks don't exist or are incomplete, completing table top exercises is a great way to get a relatively complete runbook created.

Upon completion of any exercises, the DR plan document should be updated in any way needed to ensure timely resumption of services in the event of a DR scenario.

Testing the Plan

- It is critical to test the plan
- The following are common types of tests:
 - Checklist -Consistency testing
 - Structured walk-through - Validity testing
 - Simulation
 - Active simulation
 - Full interruption



SANS

SEC401 | Security Essentials Bootcamp Style 197

Checklist testing, also known as consistency testing, simply involves reviewing the business continuity plan to ensure that it addresses all critical areas of the enterprise and that the procedures to recover those areas are accurate and consistent. Checklist testing is the least expensive of all the testing methods; however, it is also the least valuable because it does not depict the company's responsiveness to disruption. Checklist testing is for sanity checking and should not be considered a viable testing method in and of itself.

Structured walk-through testing, also known as validity testing, ensures that the plan contains no errors, erroneous assumptions, or blind spots, and that it accurately reflects the company's ability to recover from disruption. Team members and other individuals who are responsible for recovery meet and walk through the plan step-by-step.

Simulation testing involves a mock-up of an actual emergency where team members respond as if an emergency is occurring. You may recover locations (including the emergency operations center and the alternate sites) and enable communications links, while team members execute the recovery steps in a walk-through manner. You do not actually perform recovery actions (restore backups). This testing method can be expensive for a company, and it can prove invaluable for the dollars spent. A simulation test is a satisfactory testing method because it gives the enterprise fairly good insight into its recovery responsiveness.

Reference

1. BCP/DR, <http://www.paradigmsi.com/solutions/bcpdrcoop-consulting/>

Plan Deactivation

- Who should deactivate?
- When should deactivation take place?
- What is involved?
- The deactivation of the plan is to go over all lessons learned and observations from the DR scenario and its execution

Just as important as knowing when the organization business processes have been reconstituted is the importance of defining when it is appropriate to deactivate the plan. The plan should be deactivated by the executive staff or per the recommendation of the recovery teams. The plan should only be deactivated once the organization is back to business as usual or very close to it and there are no additional processes which must be fully restored as a result of the initial DR scenario.

The deactivation of the plan is to go over all lessons learned and observations from the DR scenario and its execution. This is to treat the completion of the DR execution as though it was a dry run and use it as an opportunity to improve the plan and its execution.

Top BCP/DRP Planning Mistakes

- Lack of BCP testing
 - Limited scope
 - Lack of prioritization
 - Lack of plan updates
 - Lack of plan ownership
- Lack of communication
 - Lack of public relations planning
 - Lack of security controls
 - Inadequate evaluation of vendor suppliers
 - Inadequate insurance (loss of life)

SANS

SEC401 | Security Essentials Bootcamp Style 199

A number of common—almost predictable—mistakes made in contingency planning are listed on this slide.

Lack of BCP Testing/Over-Reliance on BCP

Many companies believe that just having the BCP is enough. The document is just a lifeless draft without adequate updating and testing. Organizations that test their BCP consistently often find critical flaws, as well as areas needing improvement. The time to discover these is in advance of a real disruption. For less expensive testing more frequently and more affordable than full-fledged, off-site tests, try simulating a disaster, as in a business simulation game. Pretend something has happened, with certain resources no longer available, and have your personnel (who are assumed available) walk through the plan.

Too Limited in Scope

An incomplete BCP will not address all of the organization's needs for recovery. The BCP needs to cover organizational processes and process dependencies, systems recovery, as well as the replacement of key personnel, if needed. The organization needs to continue to function throughout a disruption and beyond.

Lack of Prioritization

There is a need to prioritize the key business processes. The risk is to prioritize less-than-critical processes instead of the ones crucial for business survival. This is a time for thoughtful evaluation and decisions.

Lack of Plan Updates

The BCP should be updated periodically, especially when there are significant system or business process or personnel changes.

Lack of Plan Ownership

Someone with the power to lead, influence, prioritize, and organize the BCP is instrumental to the success of the program. This is true during planning, as well as during the execution of the plan.

Lack of Communication

There is a need for clear and precise communication with all affected stakeholders of the organization, potentially: employees, contract employees, vendors, business partners, customers, and shareholders. (This relates to Public Relations planning next.)

Lack of Public Relations Planning

Organizations often fail to consider public and investor relations, to limit the perceived disaster impact. This can literally make or break the organization.

Lack of Security Controls

During the recovery process, sometimes security controls are disregarded, resulting in a greater risk of exposure. Security controls likely might need to be altered and loosened during recovery. But, this should be a matter of a conscious decision and empowerment that are built into the plan. During execution of the plan, there should be strict adherence to the security controls incorporated into the plan.

Inadequate Evaluation of Vendor Suppliers

Many companies poorly evaluate recovery products (hot site, cold site, and planning software), relying on vendor-supplied information. This often leads to a solution that might not adequately address a company's needs.

Inadequate Insurance

Some organizations lack adequate insurance coverage and fail to support the filing of insurance claims, and these inadequacies result in delayed or reduced settlements. The plan might lack appropriate processes for capturing losses and recovery costs, without which the organization might realize a loss greater than otherwise necessary.

Summary

- BCP/DRP allows you to have a plan in place to protect your organization
- Proper planning and testing is required to have an effective plan
- Buy-in is needed across the organization
- Key personnel across the organization must all be involved in building the plan

A key motto in cyber security is to hope for the best and plan for the worst. It is better to have a plan and not need it than need a plan and not have it. BCP tends to focus more on a strategic look at business process, focusing in on proactively addressing issues before they occur. In contrast, a DRP focuses in on a tactical look on how to recover critical IT resources, after a disaster occurs. Both are critical in order to have an effective contingency planning process.

In order for the BCP/DRP to be successful key personnel must be involved. In addition, key management must have buy-in and fully support the plan. Finally, the plan must be fully tested and kept up to date. Nothing is worse than having a disaster with an outdated plan that does not allow full recovery.

SANS

Module 23: Risk Management

Module 23: IT Risk Management

This page intentionally left blank.

Objectives

- Risk management overview
- Best-practice approach to risk management
- Threat assessment, analysis, and report to management



Risk management involves an understanding of how security is implemented in your organization and how security threats affect your business operations. As a general rule, before you can begin managing risks, you need to understand your business operations and the types of risks that they might be exposed to.

Why is risk management so important to an organization? The fact is there are risks all around us. Some risks are not that damaging, although some can cause catastrophic results. The question is whether you know what those risks are. More importantly, what will you do if they become real?

As you might imagine, every industry has its share of operational risks. The information technology field is the same. Any computer or system on the Internet or another network is vulnerable to an attack. Having a system on the Internet is like taking a martial arts class—you are going to get hit. The questions you need to ask yourself are: How hard are you going to get hit? What is the damage if I do get hit? What can I do to minimize the damage? Remember, in risk management, we are concerned with the cause, the effect, and our response to the risk incident.

In this module, we structure our definitions and assumptions about risks around the concepts of the information security triad: confidentiality, integrity, and availability. We should keep these concepts in mind when performing risk assessments and subsequent risk-management decisions. In risk management, we are looking at ways to minimize the impact that could affect the confidentiality of our information, the integrity of our systems and data, and the availability of our infrastructure.

Risk management helps information systems (IS) management strike a balance between the impact of risks and the cost of protective measures. The goal of risk management is to identify, measure, control, and minimize or eliminate the likelihood of an attack.

References

1. Risk and Fraud Management, <http://www.account.com/sectors/risk-fraud-cyber-security/>
2. "Network Intrusion Detection, Third Edition" - Northcutt and Novak's 2002

The student will understand the terminology and basic approaches to cyber security risk management

This module focuses on cyber security risk management: the art of analyzing threats and vulnerabilities and determining the impact these risks can have on your enterprise. Risk management is much more than just determining the various risks to which you are exposed. It is an exploration of the various approaches and techniques for managing these risks.

You might ask yourself: Why is risk management so important? It is because every computer hardware or software implementation has some security risk associated with its use. For example, take the situation where your company wants to implement a wireless LAN architecture to co-exist with the wired network. There are documented (and some undocumented) risks associated with wireless LAN (WLAN) technology. Do you just ignore these risks and implement WLAN without any worries? This is where risk management techniques are used to determine the level of risk, and if we can live with that risk level.

Risk management's main focus is to reduce the risk until it is at an acceptable level. The actual acceptable level varies from company to company. However, risk management means that we need to identify, control, and minimize the loss associated with each risk. We begin by understanding the risk-management process, the concepts of threats and vulnerabilities, and their relationship to risk assessments.

Cyber Security Risk Management Process

Steps for an effective risk management process

1. Conduct a rapid assessment of risks
2. Fully analyze risks or identify industry practice for due care
3. Set up a security infrastructure
4. Design controls
5. Decide which resources are available and implement countermeasures
6. Conduct periodic reviews
7. Implement intrusion prevention and incident response

SANS

SEC401 | Security Essentials Bootcamp Style 205

The objective of risk management is to identify specific areas where safeguards (or countermeasures) are needed to prevent deliberate or inadvertent unauthorized disclosure or modification of information.

The steps for an effective risk management process are

1. Conduct a rapid assessment of risks so you know what your security policy needs to cover. This forms the basis for your security policy, with input from various business departments.
2. Fully analyze risks or identify industry practice for due care; analyze vulnerabilities.
3. Set up a security infrastructure.
4. Design controls; write standards for each technology.
5. Decide which resources are available, prioritize countermeasures, and implement the top priority countermeasures you can afford.
6. Conduct periodic reviews and possibly tests.
7. Implement intrusion prevention and incident response.

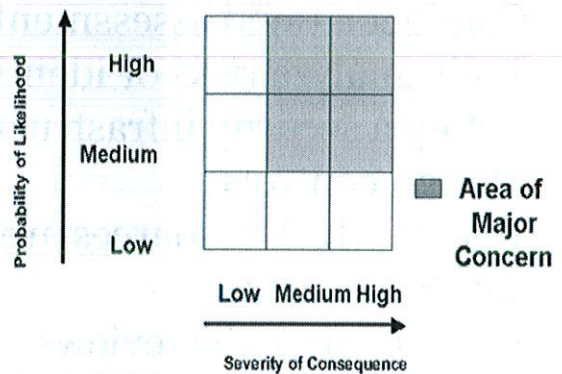
We need to start with policy because it dictates the security posture the company wants to take with respect to protecting its resources.

If you have an open security policy (for example, you allow anything in and anything out of your corporate network), and you are concerned about the risks to your network, then your desired policy does not match your implementation. The security policy points you to areas of your business operation that require protection. It is not possible to implement 100% protection for your enterprise. The best approach is to concentrate first on protecting those areas of your organization that, if compromised, could incur the most damage.

Cyber Security Risk Management: Risk Analysis Matrix

- Determine overall threats and vulnerabilities
- Create risk matrix for each business unit focusing in on likelihood and consequence (impact)

Risk Analysis Matrix



A key step in risk management is to analyze risks and determine their impact to your organization. This also involves looking at the industry's best practice for maintaining security.

As we stated previously, risk analysis involves determining the risks and determining their impact on the infrastructure. The figure in the slide is a risk analysis matrix. The X-axis is the severity of consequence, rated from low to high. That is, as the risks or the degree of severity increases, so does the damage it does. The Y-axis is the probability of likelihood that the risk could really happen, also rated from low to high. The goal is to concentrate on those areas that result in a medium-to-high severity of consequence and a medium-to-high likelihood that it would actually occur. For example, the severity of consequence of a huge meteor hitting Earth is high, but the probability of likelihood is low. This scenario would not be an area of concern. However, putting our e-commerce on the Internet and not protecting it with a firewall could result in a high probability that the system would be compromised and a high severity of consequence.

Risk

$$\text{Risk} = \text{Threat} \times \text{Vulnerability}$$

Threat: Any event that can cause an undesirable outcome

Vulnerability: A weakness in a system that can be exploited

Risk identification involves understanding the associated threats and vulnerabilities you might be exposed to.

What is the definition of risk?

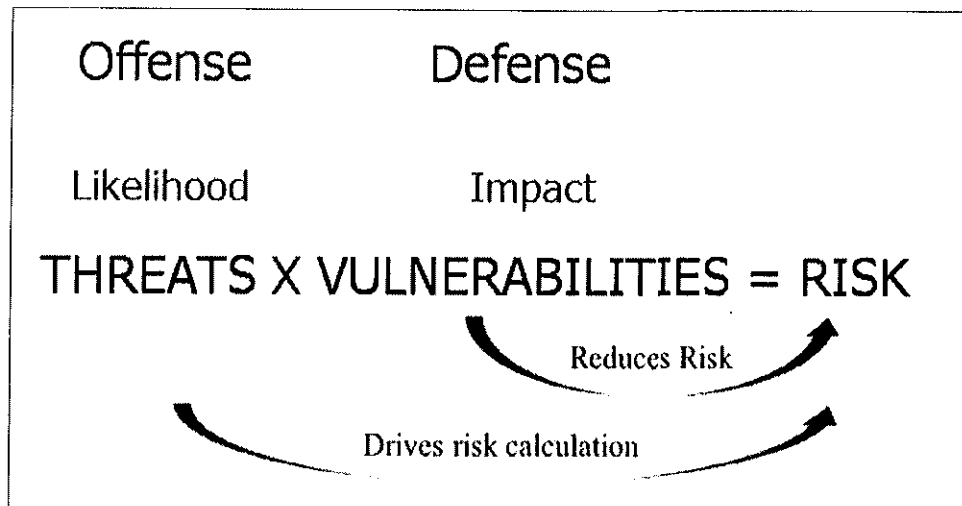
The classical definition of risk is

$\text{Risk} = \text{Threat} \times \text{Vulnerability}$

A threat is any event that can cause an undesirable outcome. A threat could be the exploitation of a vulnerability. The threat is that someone could actually exploit this weakness and compromise your system.

Vulnerability is defined as a weakness in a system that could be exploited. You have heard this one before, "A vulnerability has been found in the SSH service that, if exploited, could result in a buffer overflow," or something like that. The vulnerability is the fact that the SSH service has a flaw, a weakness that could lead to a system compromise. The danger lies in the fact that these hidden vulnerabilities are discovered and subsequently exploited.

Key Areas of Risk



It is important to understand that the focus of security is to understand, manage and mitigate the key risks to an organization's critical data. The trick is to focus on the highest priority risks. In order to do this, we must always remember that threats drive the risk calculation. We must identify and focus on the threats that have the highest likelihood of occurring or the areas in which offense is going to focus in order to cause harm to our organization. Once we identify the key threats, we then identify which vulnerabilities would allow those threats to have the greatest impact and focus on those vulnerabilities to reduce the overall risk. This will enable us to improve our defensive posture.

It is important to remember that both threats and vulnerabilities need to be utilized to verify that the correct risks are being remediated. It is also important to note that the two keys to focusing in on the proper risks are to look at likelihood and impact (i.e. consequence).

Risk-Management Questions: Risk Requires Uncertainty

- What could happen? (What is the threat?)
- If it happened, how bad could it be? (Impact of threat)
- How often could it happen? (Frequency of threat—annualized)
- How reliable are the answers to these three questions? (Recognition of uncertainty)

To decide among accepting, mitigating, or transferring the risk, we need to better understand the risk and how it affects us.

When evaluating risk, it is helpful to ask yourself some key questions:

1. What could happen?
2. If it happened, how bad could it be?
3. How often could it happen?
4. How reliable are the answers to the previous questions?

The answers to these questions help us focus on the actual threats and gain a better understanding of their impact if they were to actually happen. The first question is to ask ourselves: What exactly are we afraid of? What is the actual threat? Is the threat something tangible? Can we accurately define the threat?

If we can define the threat, what damage could it cause? What is the probable extent of the damage? For example, the damage could be anything from a few corrupted files to complete deletion of all critical files. In other words, what is the impact of the threat? Another variable to consider is the frequency of the threat. How often could this threat happen? Is it just once, or can it occur more often?

The last question relates to the recognition of uncertainty. That is, how sure are you of the answers to the three questions? Can you validate and prove your answers? This might be a difficult question to answer because it might be difficult to accurately perform our risk calculations on operating systems or new programs when new vulnerabilities are constantly being discovered.

SLE and ALE

Single Loss Expectancy (SLE): The loss from a single event

Annualized Loss Expectancy (ALE): Annual expected loss based on a threat

When all is said and done, in the end, it all comes down to money. What management will be considering is, "How much financial loss are we willing to accept in a single (threat) event?" If a company's database is compromised and that database contains your proprietary (and valuable) secret formula for your next revolutionary drug, then you could not afford even one risk to your system that might lead to the theft of this formula. Remember that risk involves uncertainty. The uncertainty here is that we cannot accurately determine the exact value of the formula. (It might make millions of dollars, or it might not make any money at all because the formula might not work.)

In order to start putting a dollar value on the risk, we perform 2 calculations. SLE and ALE. The SLE or single loss expectancy is the starting point. We are determining what the cost will be if this occurs once. While this is a good starting point, very few risks only occur once. Therefore we need to tie in the annualized rate of occurrence and calculate the ALE or annualized loss expectancy.

Single Loss Expectancy (SLE: One Shot)

- **Asset value x exposure factor = SLE**
- Exposure factor: 0 – 100% of loss to asset
- Small conference, one event/yr.
- Weather causes 50% drop in turn out
- Revenue \$100 k
- $\$100,000 \times .5 = \$50,000$ loss expectancy

The SLE is the dollar value that is assigned to a single event. That is, it is the organization's loss from a single event. The formula is

Single Loss Expectancy = Asset Value (\$) X Exposure Factor (EF)

The Exposure Factor (EF) is the percentage of loss a threat event would have on the asset. The EF is expressed in terms of 0 to 100% loss to an asset. In the conference business, you have to guarantee a certain number of hotel rooms, whether or not people show up to stay in them. Event planners use classic risk-management techniques to help them understand their overall risk. If there is an adverse condition, such as weather, this could impact attendance because people just do not show up. For example, if a bad storm causes flights to be canceled and causes a 50% drop in turn out to a small, focused conference with a value of \$100,000, the single loss expectancy would be \$50,000, because the assumption is that the event would result in a 50% loss.

Annualized Loss Expectancy (ALE: Multi-Hits)

- **SLE x Annualized rate occurrence (ARO) = Annual Loss Expectancy (ALE)**
- Annualized rate of occurrence: The frequency the threat is expected to occur
- Example, web surfing on the job:
 - SLE: 1000 employees, 25% waste an hour per week surfing, \$50/hr x 250 = \$12,500
 - ALE: They do it every week except when on vacation: \$12,500 x 50 = \$625,000

Most risks happen more than once. In those cases, we calculate the Annualized Loss Expectancy (ALE). The ALE is the annual expected financial loss from a threat. The formula is:

Annual Loss Expectancy = Single Loss Expectancy x Annualized Rate of Occurrence (ARO)

The Annualized Rate of Occurrence (ARO) is the estimated frequency at which a threat is expected to occur. Its value can range from zero to a large number. Sometimes, the ARO is easy to calculate. Other times, it is difficult to compute; in fact, many times, this number represents the uncertainty factor in our risk management calculation.

As a real-case scenario, imagine you need to calculate the amount of revenue loss because of your employees' web surfing during work hours (not work-related, of course). We start by calculating the SLE. For this, we need the asset value and the exposure factor.

If 25% of your 1,000 employees waste one hour of their time each week surfing the web and the cost per hour is \$50, then the formula becomes

$SLE = \$50/\text{hr.} \times 250$ or \$12,500 per week

That cost is not the significant BUT is only a single occurrence.

If we want to calculate the annualized cost, the formula becomes:

$ALE = \$12,500 \times 50$ weeks (assuming a two-week vacation) or \$625,000 per year

Look at the difference between the SLE and the ALE. Just by adding in the ARO, look how significantly the value increased.

Quantitative Versus Qualitative

Quantitative Risk Assessment

- Assigns an exact numeric value
- Far more valuable as a business-decision tool because it works in metrics, usually dollars

Qualitative Risk Assessment

- Easier to calculate but results are more subjective
- Results typically categorized as low-, medium-, or high-risk events
- Succeeds at identifying high-risk areas

There are two risk assessment approaches: qualitative and quantitative. In quantitative risk assessment, we try to assign an objective numeric value; typically, this value represents a monetary loss value. Qualitative risk assessment, on the other hand, deals with more intangible values and focuses on variables and not just the monetary losses.

Quantitative risk assessment is a far more valuable business tool because it works on metrics—usually in dollars. The bottom-line cost in dollars is what management is looking for when trying to understand the implications of how a risk can affect the organization.

Qualitative risk assessment is much easier to perform and can identify high-risk areas. For example, you need to perform a risk assessment to determine the impact of installing a wireless LAN access point in your organization. The first order of business is to determine the vulnerabilities, threats, and therefore the risks of using a wireless LAN. Then, you determine whether those risks apply to your organization and determine the likelihood that you are at risk. One of the risks of using a wireless LAN is the possibility of someone sniffing the wireless network traffic, and that a misconfigured access point can allow rogue client connections. These are real risks that need to be addressed. Can you put a monetary value to these risks? If someone does connect to your network via the open access point, how much is that going to cost your company in lost revenue?

As you can see from this example, quantitative risk analysis in this situation does not quite work. A qualitative approach is much better because we can arrive at a more subjective result. In qualitative risk assessment, the results are typically categorized as low-, medium-, or high-risk events. A person operating a wireless LAN access point in their house, where the nearest neighbor is 5 miles away, is at a low risk of having someone trying to connect to the network. A company in the middle of a high-tech park, with an access point that allows rogue connections, has a high risk.

Threat Assessment, Analysis, and Report to Management

The student will be able to identify each step in the Threat Assessment and Analysis process and how to report findings to management

Threat Assessment, Analysis, and Report to Management

Now that we understand risk, it is important to develop a process that we can use to present the findings to management. The key focus of the findings is to generate a report in which we identify the high risks items, the likelihood of it occurring, the cost if it occurs and the cost to fix it.

Business Case for Risk Management

- Use qualitative, quantitative, or best practice/checklist risk measurement to define the gap between our current risk status and where we want to be
- After the gap analysis, we select safeguards, such as
 - Host based solutions
 - Network based solutions
 - Preventive measures
 - Detective measures
 - Logging
 - Data focused controls

It all eventually comes down to making the presentation to management and the need to convey the big picture. It is not enough to understand the core technologies we use for our countermeasure controls: host- and network-based intrusion prevention or detection, logging, and data focused controls. The question is this: Can you show them how these technologies work together to produce the results needed?

Every enterprise has different needs and diverse expectations. A financial institution has different priorities than a military organization. A pharmaceutical company's valuable assets could be the formula for a new drug. A financial institution's assets could be client lists and account numbers. Everyone has something different to protect and a different tolerance for risk.

Business Case: Applications

Business case should always map back to risk

- Organization has rudimentary capability, and you want to upgrade
- Organization has central monitoring, and you are presenting the case for a departmental capability

If you cannot provide proof that systems are at risk, it becomes more difficult to get additional funds for the countermeasures that you recommend

Now that we have introduced the basic risk-assessment process, let's apply this process to the business case for an intrusion prevention system.

First, let's consider the different scenarios we might be working with:

- The organization has a rudimentary intrusion prevention system, and you might recommend upgrading the system
- The organization has a central monitoring system, and you are presenting the case for more proactive detection

One of the problems you might face is that many managers are uncomfortable when confronted with actual data about attacks and vulnerabilities. They might clearly see this as a weakness on their part to do their job. Even as an outside consultant, you might face the same roadblock. In fact, as a consultant, you might feel a lot of resistance, even from the system administrators. This is because they might feel that you will show management that they have not been adequately performing their jobs.

It could also be the case that managers just don't understand the severity of the situation. They might not really believe that there is a problem. If you cannot provide proof that their systems are at risk, it becomes more difficult to convince them to spend additional funds for the countermeasures you recommend.

Step 1: Threat Assessment and Analysis

Identify the types of threats

Look for evidence that these threats are actually in use and remember the threat vectors:

- External attack from network
- External attack from a business partner
- Insider attack from local network
- Insider attack from local system
- Attack from malicious code

Step 1: Threat Assessment and Analysis

Any system connected to the Internet is vulnerable to possible hacker and worm compromise. These attacks can come from many sources. If we do not install any security protection on our systems, how much system compromise (remember to think in terms of confidentiality, integrity, and availability) can we withstand? If we have valuable assets on our information systems to protect (and we all do), then what countermeasures should we install that will protect us from outside attacks?

The threat of a destructive worm is currently one of the most likely negative events and potentially most catastrophic. Although there are thousands of threats, we use that specific one to illustrate the risk-management process.

When determining what types of threats your enterprise could be exposed to, it is vital that information-security professionals spend time thinking about how they might be attacked. That is, you need to enumerate all possible threats you might have to deal with. After the list is compiled, you can then look for evidence that these threats are actually viable threats to your enterprise. For example, your initial list of threats includes destruction of the data center by a natural disaster (i.e. earthquake). After investigating further, you determine that earthquakes are not prevalent in that area and the likelihood of an earthquake is nonexistent.

The process of determining what is at risk and what is the impact if the identified threats materialize is known as “risk analysis.”

The purpose of risk analysis is to

- Identify existing countermeasures, threats, and vulnerabilities
- Support the expenditure of resources and to determine the most cost-effective safeguards to offset the risks
- Aid in the selection of cost-effective countermeasures that reduce existing risks to an acceptable level

The best way to focus on the real threats is to focus on the threat vectors as highlighted previously in the risk-analysis matrix figure.

External Attacks

- Newspaper, web articles on attacks at other places, Internet resources
- Hacking websites
- Firewall/intrusion prevention logs are an excellent source for specific threats
- Internally, try traceroutes to private addresses
- Try to connect to your wireless networks from the parking lot

Attacks coming from the outside to your internal network from the Internet have been well documented.

We hope you have implemented a firewall and an IPS. If you have, the best sources of information on what is coming into your system are the firewall logs and other security devices.

Another effective method of determining what threats or attacks you are vulnerable to is to scan your own network. One example is a scan that looks for significant problems, such as backdoors. Look for broadcast packets from your internal/private address space coming in from the Internet. If you find evidence of this, it is most likely someone trying to get into your internal network from the outside.

Also look for ways into your network that might bypass the firewall, such as wireless LAN access points. In fact, check out any network device installed with its default settings. More times than not, the default settings are weak and are easily exploitable.

I remember the first situation where I installed a personal firewall on my laptop. One day I was staying at a hotel in Florida that had Internet access. I turned on the log feature on my firewall because I was curious to see if anyone would be trying to get into my system once it connected to the hotel network. I was not online for more than 5 minutes when my firewall started beeping at me. Looking at the logs I found out that another machine was trying to do a network scan! Unfortunately, I had not put the firewall at its highest setting — the stealth mode. So they were able to see me on the network. I immediately configured it so I was completely hidden from network view, and added the machine trying to scan me to the list of blocked IP addresses.

Insider Attacks

- Insider attacks are fairly advanced, subtle, and often the cause of damage
- Although insider attacks are tough to detect, it is pretty easy to instrument attractive systems or programs and watch for access - honeypots
- Utilize existing security technologies
 - Virus-scanning software
 - Tripwire UNIX and Windows (Host IDS/IPS)

Using firewalls and IPSs to monitor activity coming from outside your network is vital. But, that does not mean you need to neglect looking for and monitoring your system from attacks from the inside. Insider attacks are often fairly advanced or subtle, and there is the possibility of extensive damage. This is because, in many cases, security inside the network is more relaxed. Remember, if an attacker can penetrate your perimeter—via a backdoor, through a hole in the firewall, or using malicious code delivered by an e-mail—you won't see their activity if you are not monitoring your internal network.

Many commercial and freeware products can help you monitor your internal network. In a UNIX environment, use a host-based IPS to report system events to a central location. Just make sure the central log system is well secured. If an attacker can modify your primary source of information, you might never figure out how he is getting in.

Another popular host IPS is a product called Tripwire (<http://www.tripwire.com>). It is available for UNIX and Windows. Tripwire can be configured to monitor critical system files.

It creates a cryptographic hash value of the file and if the file is modified, it can detect a change because the new hash of the modified file is different than the original value. Although personal firewalls are ideal for laptop computers that are used on the road, they are ideal for use on internal host systems as well.

The main issue that you have to deal with is that you have to review all these audit logs to find out what is going on with your systems. The native Windows and UNIX logs are not easy to work with; they are hard to parse and they don't always give you exactly the information you need. But, many third-party tools can make your life a lot easier.

Another way to monitor insider attacks is by using a honeypot system. The point of a honeypot is to confuse an attacker, causing them to waste time trying to break into a system of no value. Although they are attempting to break in, you can collect information about their tools and techniques. A honeypot system can be implemented to lure and monitor insider attacks. You can configure a honeypot that looks like a payroll system or perhaps an advanced research system. Then, you sit back and wait to see who, without the proper authorization, tries to gain access.

Malicious code is one of the more significant problems organizations face today. Virus-scanning software can be configured to generate reports as to how many viruses are being detected. If you see an increasing amount of viruses reaching some of your internal systems, perhaps you should look at how they are getting in. Many e-mail attachments are Trojan horses that might install software on your systems to facilitate unauthenticated remote access. Installing a host IDS product, such as Tripwire, helps monitor the malicious modification of critical system files.

Step 2: Asset Identification and Valuation

If you work in cyber security you must understand and know how to read financial statements.

If you do not know how much an asset is worth, how do you know how much to spend protecting it?

If you know what your assets are worth, it is easier to justify the increased cost of the security controls

Step 2: Asset Identification and Valuation

Although management might have a good understanding of the cost of hardware, software, maintenance, and licensing fees, it might not be aware of the value of information assets. It might be a good idea to document the valuation of the assets you want to protect. If you know what your assets are worth, it is easier to justify the increased cost of the security controls. For example, you might find it easier to justify spending \$25,000 on a firewall when you are protecting a system that generates more than \$2.5 million in revenue per year.

Assets come in many shapes and forms. If your e-commerce website processes credit card orders and maintains customer records, your most valuable asset might be the client database system. Or perhaps your organization is involved in developing a new drug or vaccine. Your asset would be the systems that contain information about this new product or discovery. If this information fell into your competitor's hands, it could be catastrophic to your bottom line.

The asset valuation might take the form of hard monetary values. Management understands the quantitative analysis of asset valuations. For example, your company's web-based sales generate over \$2.5 million in sales. This high-dollar loss in revenue could be the catalyst needed to convince management to spend more money on security controls. The difficulty comes in the valuation of intangible assets, such as projected income or reputational damage. Nevertheless, anything that is central to a revenue center has a higher bottom-line importance in management's eyes.

Step 3: Vulnerability Analysis

Vulnerabilities are the gateways by which threats are manifested

- Doorways for the use of exploit code or techniques
- Increase frequency of threat event
- Increase impact of threat event
- Vulnerabilities are the primary focus for reducing an overall risk

Step 3: Vulnerability Analysis

As stated earlier, vulnerabilities are weaknesses that could be used by an attacker to compromise a system. Every day, new vulnerabilities are uncovered. There are several websites that can keep you informed of the latest vulnerabilities.

Although it is critical to learn about newly discovered vulnerabilities, it is just as critical to know about older vulnerabilities. It is important because many systems do not have the latest hotfixes or patches installed. Many attacks exploit older vulnerabilities because they know that some systems are not patched correctly.

It is ironic that in many cases, the most critical systems in our enterprise are the ones that do not have the latest patches. This is because the application of hotfixes and patches typically requires the system to be rebooted. These critical servers (for example, web servers, domain controllers, e-mail servers, and so on) need to be operational 24/7, and bringing these systems down for any amount of time is not always an option. So, these critical servers remain unpatched and, therefore, vulnerable to well-known attacks.

Step 4: Risk Evaluation

- Match threats and known vulnerabilities, calculate ALE
- Estimate risk from unknown (not yet discovered) vulnerabilities
- Risk might be expressed monetarily (preferred) or qualitatively

Step 4: Risk Evaluation

Understanding the risks to the enterprise is only the beginning part of the process. The next step is to perform a risk evaluation. Risk evaluation is the process of taking the vulnerabilities identified in the previous step, and determining the impact levels (if any) they will have on your enterprise if exploited.

Not all vulnerabilities impact your organization. Remember our discussion on vulnerabilities, where we might see a notice that says, "A vulnerability has been found in the SSH service that, if exploited, could result in a buffer overflow."? It could very well be that, although this vulnerability could cause major damage to our system if exploited, this vulnerability might not even affect us whatsoever (that is, we do not have the SSH service installed so that vulnerability cannot be exploited). Risk evaluation involves looking at the vectors that you know apply to your organization and then listing some known vulnerabilities that could be exploited via these vectors, regardless of any countermeasures installed.

After the threats are matched to known vulnerabilities, you can then calculate the Annualized Loss Expectancy (ALE). Remember that the ALE is calculated by multiplying the Single Loss Expectancy (SLE) by the Annualized Rate of Occurrence (ARO). The SLE is calculated by multiplying the asset value (AV) times the exposure factor (EF). When calculating the value of the assets, consider the investment value, cost of the hardware and software, time your organization has invested, and potential loss of revenue.

Step 5: Interim Report

Project summary: Clearly list the top risks, the likelihood if the risk occurs, the cost if it occurs and the cost to fix it.

Asset identification and valuation report: Present the critical assets that were found and its value. It is critical to tie the threats with the vulnerabilities that will have the biggest impact

Plan to make things better: NEVER brief senior management without a plan

Remember that it is critical to understand the critical data and the servers that it resides on

Step 5: Interim Report

The interim report is essentially your pitch to management. This report should contain a project summary, which addresses all the steps you took to arrive at the decision in the report.

The report should also contain an asset identification and valuation report. This information is critical because management then has a better understanding of what are the valuable assets the company has, and help better justify the cost of the countermeasures to protect them. It is important to include information of what assets were identified and their value. When relating to the threats to these assets, give some worst cases and long-term sustainable losses.

The interim report should always have a plan to improve the situation. Never brief senior management without a plan on how to solve the problem. Be prepared to recommend solutions and more importantly, justify your recommendations.

Cost Benefit Analysis

Comparison of the cost of implementing countermeasures with the value of the reduced risk

- Make sure to show cost benefit analysis
- Allows look at multiple options to reduce a risk, including compensating controls

Important to show that this is high priority risk and the solution is the most cost effective of reducing it

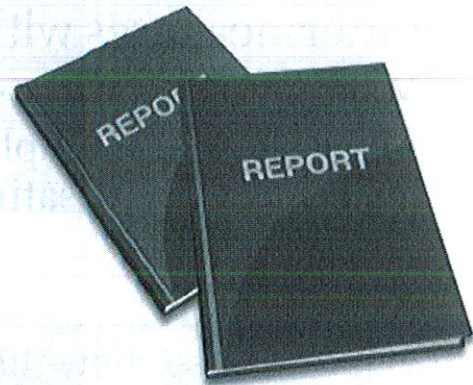
Another consideration factor is the cost of the safeguard versus the actual value of the asset. It makes sense that the cost of the protection should not be more than what the asset is worth. Would you buy a \$5,000 safe to protect an item that is worth only \$100?

Cost-benefit analysis is the comparison of the cost of implementing countermeasures with the value of the reduced risk. Can you accurately determine whether the countermeasure is 100% effective? Antivirus software is known to fail against unknown viruses, especially if the virus signatures are not up to date. Companies with firewalls are not 100% protected because firewalls can be compromised (or bypassed) and traffic containing attacks might legitimately pass through the firewall.

Benefits are the reduction in the risks your company is exposed to which means it is critical to know the acceptable level of risk. Keep in mind that the biggest benefits to the organization might be the countermeasures that protect the revenue flow. This is especially true if your organization is involved in e-commerce. The cost of a countermeasure is more than just the initial cost. There is the labor cost of monitoring the devices and the lifecycle cost.

"Final" Report

- Includes the interim report results
- Safeguard selection
 - Including easy-to-do tasks that have already been implemented
- Risk mitigation analysis
- Cost benefit analysis
- Recommendations



The final report given to management includes the interim report results, as well as the safeguard selection. These countermeasures include the actual technology control selections and might also include specific tasks. If some of the tasks have already been implemented, state so in the final report. Other components of the report include the risk-mitigation analysis, the cost-benefit analysis, and your recommendations.

When discussing the recommendations, it is best to include more than one option or solution to the problem. Make sure to map out the entire recommended architecture. Sometimes, the report can outline short-, medium-, and long-term solutions. Short-term solutions are those that can be implemented in a short amount of time and typically for little or no cost. Long-term solutions could involve a complete redesign of the infrastructure and an increased safeguard solution cost.

Reference

1. Reporting Procedures in Risk Management, <http://www.daftblogger.com/reporting-procedures-risk-management/>

Business Case: Summary

- Threat assessment and analysis
- Asset identification and valuation
- Vulnerability analysis
- Risk evaluation
- Interim report
- Establish risk acceptance criteria
- Safeguard (countermeasure) selection with risk mitigation analysis
- Cost benefit analysis
- Final report



A security professional must know the threat, understand the fundamental information security tools, and be able to apply this knowledge in risk management

This module discusses the process of risk management. We discussed how there are vulnerabilities in the software and hardware systems we use. Some of these vulnerabilities are documented and some have yet to be found. Vulnerabilities are weaknesses—security holes—that can be exploited and used to compromise the system. These vulnerabilities, coupled with threats, add up to the risks we are exposed to. Risk management is needed to make sure we are prepared just in case those threats become a reality.

We learned how we do not need to be concerned with all the vulnerabilities in our systems. We need to look at it in terms of what vulnerabilities are those we should be concerned with (the ones that directly affect you), what is the likelihood someone could exploit the vulnerability, and what would be the impact if the threat did become a reality. Calculating the SLE uses the value of the asset you are protecting and the exposure (to the risk) factor. The ALE uses the SLE value and the annualized rate of occurrence for that particular risk. These results are used to enable either a quantitative or qualitative risk assessment approach. Quantitative assigns a numeric value to the risk whereas qualitative produces a more subjective risk assessment (for example, low-, medium-, or high-risk factor).

An effective risk-management process involves working within the security framework of the security policy, identifying the various risks the organization might be exposed to, implementing a security infrastructure to handle those risks, deciding what controls should be used to build the infrastructure, and deciding what countermeasures to implement, based on the impact and severity of the risk.

Reference

1. Kaizen Risk Assessment Tool, <http://kaizencs.co.uk/risk-assesmen-tool/>

"As usual, SANS courses pay for themselves by Day 2. By Day 3, you are itching to get back to the office to use what you've learned."

Ken Evans, Hewlett Packard Enterprise - Digital Investigation Services

SANS Programs sans.org/programs

GIAC Certifications
Graduate Degree Programs
NetWars & CyberCity Ranges
Cyber Guardian
Security Awareness Training
CyberTalent Management
Group/Enterprise Purchase Arrangements
DoDD 8140
Community of Interest for NetSec
Cybersecurity Innovation Awards



Search SANSInstitute

SANS Free Resources sans.org/security-resources

- E-Newsletters
 - NewsBites: Bi-weekly digest of top news
 - OUCH!: Monthly security awareness newsletter
 - @RISK: Weekly summary of threats & mitigations
- Internet Storm Center
- CIS Critical Security Controls
- Blogs
- Security Posters
- Webcasts
- InfoSec Reading Room
- Top 25 Software Errors
- Security Policies
- Intrusion Detection FAQ
- Tip of the Day
- 20 Coolest Careers
- Security Glossary

SANS Institute

8120 Woodmont Avenue | Suite 310
Bethesda, MD 20814
301.654.SANS(7267)
info@sans.org