COSC362 review - Summary Data and Network Security

Data and Network Security (University of Canterbury)

Chapter 1 - Overview

Perfect Security Possible or Necessary?

- Takes just one “security incident” to be blamed
- Risk Assessment is essential
    - Quantification and prioritization based on likelihood and consequences
    - Must be able to assess the impact of every security threat

What is Computer Security?
Definition:
“The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability and confidentiality of information system resources” (includes hardware, software, firmware, information/data, and telecommunications).” - NIST Computer Security Handbook

Three Key Objectives

- **Confidentiality**
    - Data Confidentiality: Assures that confidential information is not disclosed to unauthorized individuals
    - Privacy: Assures that individual control or influence on what information may be collected and stored
    - E.g. Student Login password should not be improperly disclosed by others, data has not been viewed by a third party
- **Integrity**
    - Data Integrity: Assures that information and programs are changed only in a specified and authorized manner
    - System Integrity: Ensures that a system performs its operations in an unimpaired manner
    - E.g. Student name should not be modified improperly, Data has not been modified in transit
- **Availability**
    - Assure that systems works promptly and service is not denied to authorized users
    - E.g. Learn system should be available when a student wants to download lecture slides, data must be available when it is needed

Additional Security Goals

- **Authenticity**
    - Property of being genuine and being able to be verified and trusted.
    - Means verifying that users are who they say they are and that each input arriving at the system came from a trusted source.
- **Accountability**
    - Security goal that generates the requirement for actions of an entity to be traced uniquely to that entity.
    - This supports non-repudiation, deterrence, fault isolation, intrusion detection and prevention, and after-action recovery and legal action.
    - Truly secure systems are not yet an achievable goal, we must be able to trace a security breach to a responsible party.

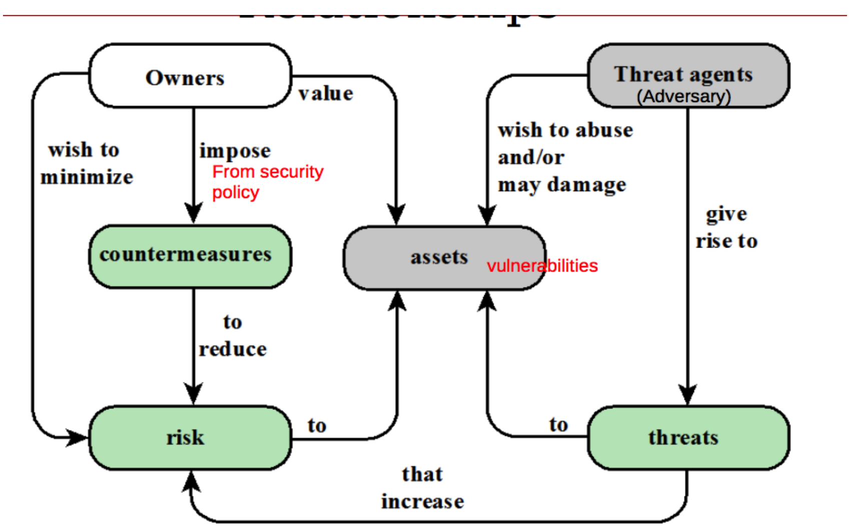
Computer Security Challenges

- Security concepts are simple and straightforward, but mechanisms are NOT
- Developers are often naive about potential vulnerabilities and security threats

    - Features themselves may contain vulnerabilities
- Real attacks are rarely as simple as they might first appear
    - Attackers are often smarter, more dedicated, better equipped, and highly motivated
- Easy to defend "known attacks"
- Security design often involve trade-offs between security assurance and ease of use
- Security is still often an afterthought

Security Concepts and Relationships

How to approach security



Attacks and their classifications

- **Passive**
    - Attempt to learn or make use of information from the system that does not affect resources:
        - Eavesdropping
        - Monitoring of transmissions
    - Two types
        - The release of message contents
        - Traffic analysis
- **Active**
    - Attempt to alter system resources or affect their operation
        - 4 Categories:
            - Replay
            - Masquerade
            - Modification of messages
            - Denial of Service
            - **Inside**
    - Initiated by an entity inside the security perimeter.
            - **Outside**
    - Initiated from outside the security perimeter.

FIPS PUB 200

Federal **I**nformation **P**rocessing **S**tandard **P**ublication

Emphasizes more security during the design, development, implementation, and operation of more secure information systems.

- **Technical Measures**

  - Access control, ID & Auth, System & Communications protection, etc.
- **Management controls and procedures**
  - Awareness & Training, audit, accountability, certification, security assessment, risk assessment, etc.
- **Overlapping technical & management**
  - Configuration management, incident response, media protection.

Fundamental Security Design Principles

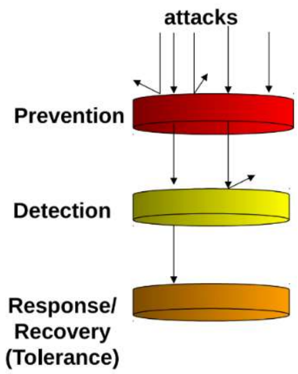
- **Economy of mechanism**
  - KISS -> Security mechanisms should be as simple as possible.
    - Simple design is easier to test & validate.
    - Fewer vulnerabilities.
- **Fail-Safe Defaults**
  - Unless given explicit access to an object, should be denied.
    - Computing systems save default is generally no access
      - ACL -> Access Control List
      - **Complete Mediation**
  - All access to objects should be checked to ensure permitted.
    - Done every time an access occurs
      - **Open Design**
  - Security of a mechanism should not depend upon the secrecy of its design/implementation.
  - Should be open for scrutiny
    - Critical observation or examination
  - E.g. cryptography and openness
    - Should have unclassified designs e.g. AES (Often used for Wireless)

      - **Separation of Privilege**
  - The system should not grant permission based on a single condition.
      - **Least Privilege**
  - An entity should only be given the privileges required to finish a task.
      - **Isolation**
  - Public access should be isolated from critical resources.
  - User files segregated except when desired behaviour.
  - Security mechanisms should be isolated (i.e. firewalls, etc).
      - **Modularity**
      - **Layering -> Defense in depth**
  - Use of multiple, overlapping, protection approaches.

Computer Security Strategy

- **Policy/Specs**
  - Value of protected assets.
  - System vulnerabilities.
  - Potential threats & likelihood of attacks.
  - Ease of use vs security.
  - Cost (recovery vs security).
- **Implementation/Mechanism**
  - Prevention, Detection, Response, Recovery.
- **Correctness/Assurance**
  - Does it really work?

Mechanisms/Implementation



- **Prevention**
  - I.e. encryption, access control.
- **Detection**
  - I.e. auditing, intrusion detection.
- **Response**
  - I.e. halt detected attack and minimize damage.
- **Recovery**
  - I.e. data backups, intrusion tolerance.

Chapter 2 - Cryptographic Tools

Basic Terminology

- **Plaintext (P)** - the original message
- **Ciphertext (C)** - the coded (or encrypted) message
- **Cipher (Encryption/Decryption Algorithm)** - algorithm for transforming from plaintext to ciphertext (plaintext) e.g. Caesar, DES, AES
- **Key** - information used in cipher known only to sender/receiver in symmetric cipher
  - 64 bits size - weak/insecure/easy to crack
  - 128 bits size - more secure
- **Encrypt (encipher) E(P)** - converting plaintext to ciphertext
- **Decrypt (decipher) D(C)** - recovering ciphertext from plaintext

Symmetric vs Asymmetric Cryptography

Shown in the following table:

	Symmetric	**Asymmetric**
Key relation	Enc. Key = Dec. Key	Enc. Key ≠ Dec. Key
Encryption Key	Secret	Public, {Private}
Decryption Key	Secret	Private, {Public}
Algorithm	Classified/Open	Open
Example	DES(56 Bits), AES	RSA (1024 bits)
Key Distribution	Required	Not Required
Number of Key	Many (Mbits/second)	Small(kbits/second)

Performance	Fast	Slow

Extra Notes

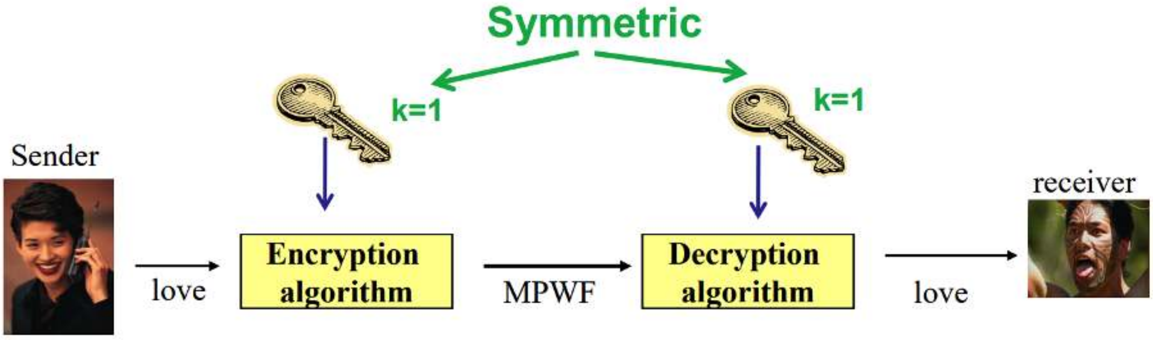
- Asymmetric Cryptography ensures that if a user has access to the key to decrypt the encrypted message, you know that the key must have come from you as you are the only person who has the key as it is private.
- Confidentiality is thus achieved.

Classical Ciphers

- **Substitution Ciphers**
  - Monoalphabetic
  - Polyalphabetic
- **Transposition Ciphers**

Caesar Cipher

Q: If key is 5 (k = 5), what will cipher text of ‘love’ be?



A:

C ≣ E(P) ≣ P + kmod(26)
L O V E

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

L ------------------->Q
O ------------------->T
>A (Wraps Around) V

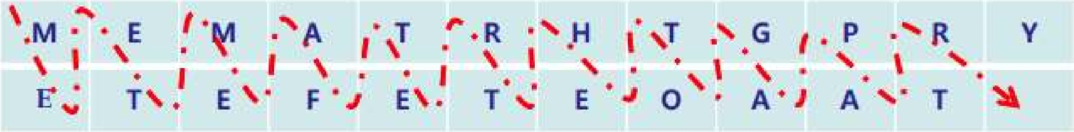
E ------------------->J

With k = 5, you shift every letter over by 5 letters
Therefore answer is: QTAJ

Transposition (permutation) Cipher

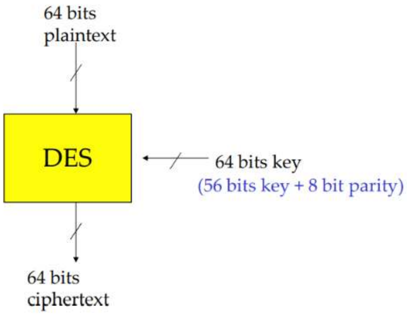
- Hide the message by rearranging the letter order without altering the actual letters used
- Can recognise these since have the same frequency distribution as the original text
- Eg. Rail-fence cipher
  - Write message letters out diagonally over a number of rows(with the depth 2 (with a key of 2))
  - Then read off cipher row by row
  - PlainText: meetmeafterthetogaparty

- CipherText: MEMATRHTGPRYETEFETEOAAT



- Why are ciphers not secure?
  - Because of language characteristics/weak key size. Brute force attacks are so easy and effective
- Any ideas for improvement?
  - Use both substitution and transposition together. Brute-force attacks are still so easy and effective if encryption strategies can be guessed

Data Encryption Standard (DES)



Insecure - Weak Due to Small Key Size.
Security Concern:

- 56 Bit Key is too short
  - There are 2^56 possible keys
  - Moore's law: processors get faster, will find it eventually
  - 1998: DES broken in 4 days

In Jan 1999 - 22.25 hours in total.
Now considered insecure due to Brute Force.
NSA made sure that the key size was small enough for them to be able to perform brute force attacks.

Brute Force Search

- Always possible to simply try every key
- Most basic attack, proportional to key size
- If you triple the key size, time increases exponentially.

Multiple Encryption and DES

- DES is not secure enough
- The once large key space, 2^56, is now too small.
- But users in commerce and finance are not ready to give up on DES
- Solution: to use multiple DES with multiple keys

DES, Triple DES

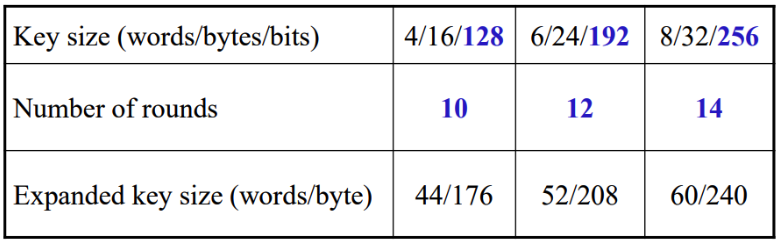
- DES has been withdrawn as a standard by the NIST
- The algorithm is believed to be practically secure in the form of Triple DES
  - Attacking cost is lot more than triple with triple DES

Triple DES

- The standards define three keying options:
    - Keying Option 1:
        - All three keys are independent.
        - Is the strongest with 168 bits (3 x 56)
        - E.g. PGP
    - Keying Option 2:
        - K1 and K2 are independent and K3 = K1
        - Provides less security, with 2 x 56 = 112 key bits
    - Keying Option 3:
        - All three keys are identical, K1 = K2 = K3
        - Equivalent to DES, with only 56 key bits
        - Provides backward compatibility with DES

Advanced Encryption Standard

- DES cracked, Triple-DES slow
- 1997 NIST called for algorithms
- Final five
    - Rijndael
    - Serpent
    - Twofish
    - RC6 (RSA)
    - MARS
- 2000 Rijndael won; 2002 Rijndael became AES
    - An iterative rather than Feistel cipher
    - Operates on entire data block in every round
- Rijndael allows many block sizes and key sizes
    - AES restricts block length to 128 bit
    - The key size can be independently specified to 128, 192 or 256 bits



Key size (words/bytes/bits)	4/16/**128**	6/24/**192**	8/32/**256**
Number of rounds	**10**	**12**	**14**
Expanded key size (words/byte)	44/176	52/208	60/240

- Adopted by US government - superseding DES
- Symmetric-key algorithm
- Public in May, 2002
- Examples of use:
    - WinZip, 7Zip, RAR
    - VPNs

Public-Key (Asymmetric) Encryption

"is any cryptographic system that uses pairs of keys: *public keys* which may be disseminated widely, and *private keys* which are known only to the owner. This accomplishes two functions:

authentication, where the public key verifies that a holder of the paired private key sent the message, and encryption, where only the paired private key holder can decrypt the message encrypted with the public key."

Requirements

- Computationally easy to generate pair of public and private keys
- Easy to generate ciphertext with public key
- Easy to decrypt ciphertext with private key
- Infeasible for someone to derive private key from public key
- Infeasible for someone to recover plaintext message with ciphertext and public key

DH Applications

- Diffie-Hellman Algorithm
- Used in:
  - Internet Protocol Security
  - Secure Sockets Layer (SSL)
  - Secure Shell
  - Public Key Infrastructure (PKI)

RSA

- 1st public key cryptosystem
- Believed to be secure if IFP (Integer Factorization Problem) is hard and worldwide standard for last 30 years
- Most widely used asymmetric encryption algorithm
- this asymmetry is based on the practical difficulty of the factorization of the product of two large prime numbers, the "factoring problem".
- The acronym RSA is made of the initial letters of the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman, who first publicly described the algorithm in 1978.
- Clifford Cocks, an English mathematician working for the British intelligence agency Government Communications Headquarters (GCHQ), had developed an equivalent system in 1973, but this was not declassified until 1997"
- Used in SSH
- How Secure is an n-bit RSA key?
  - Recommendations
    - For high-security applications or for data that needs to remain confidential for more than a few years, you should use at least a 2048-bit key.
    - To keep data confidential for more than two decades, RSA recommends a key size larger than 2048 bits.
- RSA key length vs performance
  - With every doubling of the RSA key length, decryption is 6-7 times slower

Pretty Good Privacy

"PGP encryption uses a serial combination of hashing, data compression, symmetric-key cryptography, and finally public-key cryptography; each step uses one of several supported algorithms. Each public key is bound to a user name or an e-mail address. The first version of this system was generally known as a web of trust to contrast with the X.509 system, which uses a hierarchical approach based on certificate authority and which was added to PGP implementations later. Current versions of PGP encryption include both options through an automated key management server."

Chapter 3 - User Authentication

Means of Authentication

- Something the individual knows

    - Password, PIN, questions
- Something individual possesses
    - Tokens, keys
- Something the individual is/has:
    - Fingerprint, retina..
- Something the individual does
    - Voice, handwriting, …

(Text-based) Password Vulnerabilities

- Offline dictionary attack
    - Attacker gains access to the password file and compares password hashes to common passwords.
    - Countermeasures
        - Add controls to prevent unauthorised access to the password file
        - Add intrusion detection methods to identify compromises
        - If the password file is corrupted passwords are reissued
- Specific account attack
    - Attacker submits passwords until the correct one is submitted
    - Countermeasures
        - Put limit on the number of password attempts allowed (typical practise is no more than 5)
- Popular password attack
    - Attacker tries the same common password on multiple accounts
    - Countermeasures
        - Add policies that inhibit users selecting from common passwords
        - Scan the IP address of requests and client cookies, looking for submission patterns.
- Password guessing
    - Attacker gathers information about the user and the system to guess the users password
    - Countermeasures
        - Training and enforcement of password policies (making passwords difficult to guess). These address:
            - Secrecy
            - Minimum length of passwords
            - Character set
            - Prohibitors against known user identifiers
            - Length of time before the password is changed
            - Workstation hijacking
    - The attacker waits until a logged-in workstation is unattended.
    - The standard countermeasure is automatically logging the workstation out after a period of inactivity. Intrusion detection schemes can be used to detect changes in user behavior.
            - Shoulder surfing/Exploiting user mistakes
    - If the system assigns a password, then the user is more likely to write it down because it is difficult to remember.
    - This situation creates the potential for an adversary to read the written password.
    - A user may intentionally share a password, to enable a colleague to share files, for example.
    - Also, attackers are frequently successful in obtaining passwords by using social engineering tactics that trick the user or an account manager into revealing a password.

    - Many computer systems are shipped with preconfigured passwords for system administrators.
    - Unless these preconfigured passwords are changed, they are easily guessed.
    - Countermeasures include user training, intrusion detection, and simpler passwords combined with another authentication mechanism.
        - Exploiting multiple password use
    - Attacks can also become much more effective or damaging if different network devices share the same or a similar password for a given user.
    - Countermeasures include a policy that forbids the same or similar password on particular network devices.
        - Electronic monitoring
    - If a password is communicated across a network to log on to a remote system, it is vulnerable to eavesdropping.
    - Simple encryption will not fix this problem, because the encrypted password is, in effect, the password and can be observed and reused by an adversary.

Browsers often perform the role of password managers

Best Practices for Enforcing Password Policies

- Enforce password history
- Maximum password age
- Minimum password age
- Minimum password length
- Passwords must meet complexity requirements
- Store password using reversible encryption for all users

/etc/passwd File

- Text-based database of information about users that may log into the system or other operating system user identities that own running processes.
- In many operating systems this file is just one of many possible back-ends for the more general passwd name service.
- The `/etc/passwd` file typically has file system permissions that allow it to be readable by all users of the system (*world-readable*), although it may only be modified by the superuser or by using a few special purpose privileged commands.
- The `/etc/passwd` file is a text file with one record per line, each describing a user account. Each record consists of seven fields separated by colons. The ordering of the records within the file is generally unimportant.
- An example record may be:
    - `jsmith:x:1001:1000:Joe Smith,Room 1007,(234)555-8910,(234)555-0044,email:/home/jsmith:/bin/sh`

  The fields, in order from left to right, are:[1]

  1. User name: Must be unique
  2. Information used to validate a user's password; in most modern uses, this field is usually set to "x" (or "*", or some other indicator) with the actual password information being stored in a separate shadow password file. On Linux systems, setting this field to an asterisk ("*") is a common way to disable direct logins to an account while still preserving its name, while another possible value is "*NP*" which indicates to use an NIS server to obtain the password.[2] Without password shadowing in effect, this field would typically contain a cryptographic hash of the user's password (in combination with a salt).
  3. user identifier number, need not be unique.
  4. group identifier number, which identifies the primary group of the user; all files that are created by this user may initially be accessible to this group.

5. Gecos field, commentary that describes the person or account. Typically, this is a set of comma-separated values including the user's full name and contact details.
6. Path to the user's home directory.
7. Program that is started every time the user logs into the system. For an interactive user, this is usually one of the system's command line interpreters (shells).

/etc/shadow File

- is used to increase the security level of passwords by restricting all but highly privileged users' access to hashed password data. Typically, that data is kept in files owned by and accessible only by the super user
- Each entry is as follows:
  - User login name
  - salt and hashed password OR a status exception value e.g.:
    - "$id$salt$hashed", the printable form of a password hash
    - Empty string – No password, the account has no password
    - "!" – the account is password locked
    - "*LK*" or "*" – the account is locked
    - "!!" – the password has never been set
  - Days since epoch of last password change
  - Days until change allowed
  - Days before change required
  - Days warning for expiration
  - Days before account inactive
  - Days since epoch when account expires
  - Reserved

“Unix Salt”

- Prevents duplicate passwords from being visible in the password file
- Greatly increases the difficulty of offline dictionary attacks, for a salt of length b bits, the number of possible passwords is increased by a factor of 2^b.
- Becomes nearly impossible to find out whether a person with passwords on two or more systems have used the same password on all of them

Typical Password Rules

- Trade-offs among enhanced security vs ability to remember
  - Site-specific rules add confusion

How Password Crackers Work

- Build a large dictionary of possible passwords and try each of these against the password file
  - Each password must be hashed using each salt value in the password file and then compared to stored hash values
  - Password cracking program may attempt variations on all the words in its dictionary of likely passwords
  - Requires huge computational overhead

Rainbow Table

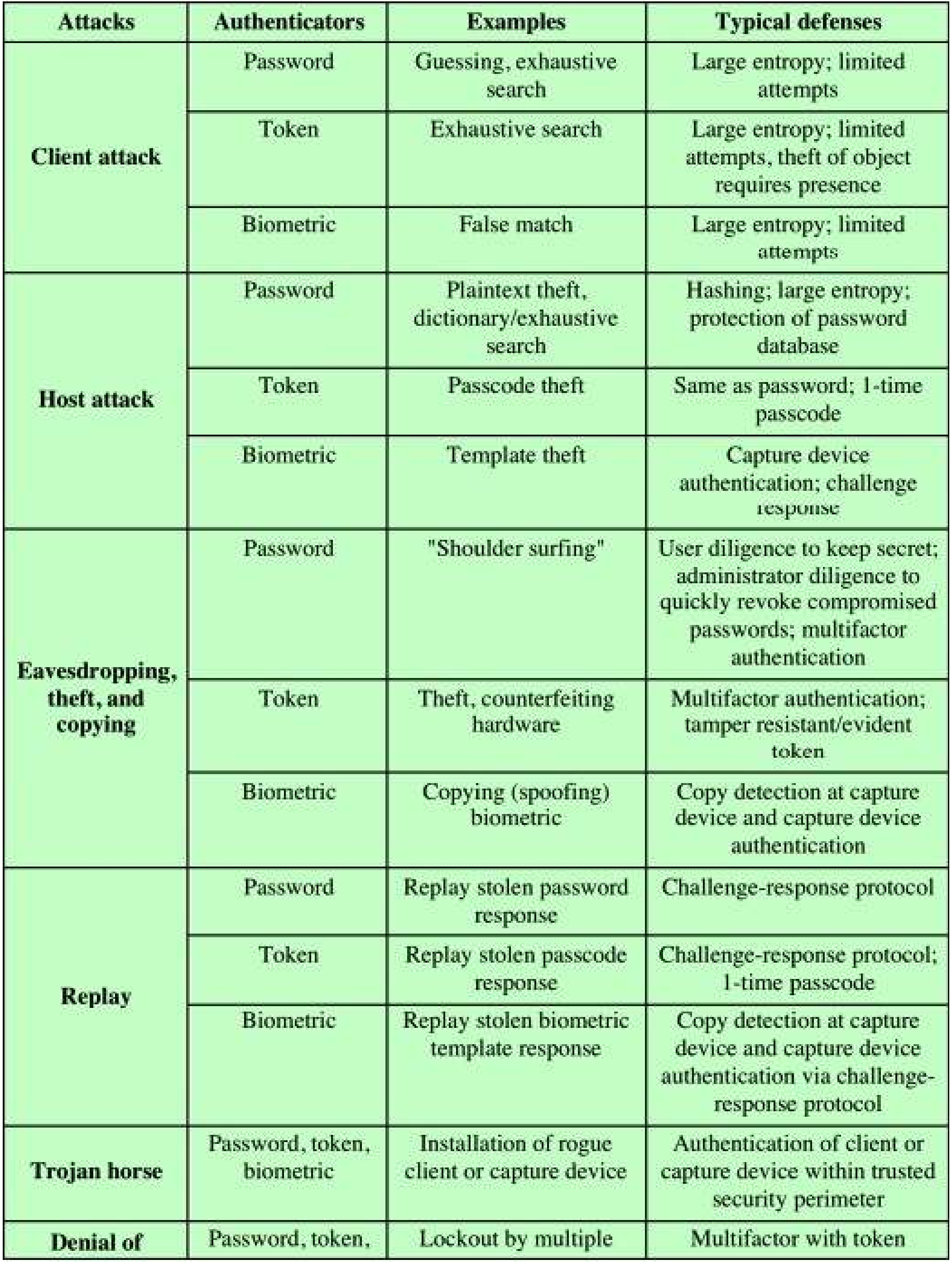
- Potential hash values can be precomputed
- Attacker generates a large dictionary of possible passwords. For each password, the attacker generates the hash values associated with each possible salt value
- Larger salt values and a sufficiently large hash value can make such attacks less practical
- Rainbow tables and/or password crackers are readily available

Password Selection Strategies

	How it works	Pros	Cons
User Education	Users can be told the importance of using hard-to-guess passwords and can be provided with guidelines for selecting strong passwords.	Can often work when a rule is applied to generate passwords and users follow a certain technique to do so.	Passwords may still be easy to crack and not all users may be educated or get educated. Guidelines may be ignored.
Computer-generated passwords	Uses an algorithm which generates random characters	Hard to crack.	If the passwords are quite random in nature, users will not be able to remember them. Even if the password is pronounceable, the user may have difficulty remembering it and so be tempted to write it down.
Reactive password checking	System periodically runs its own password cracker to find guessable passwords. The system cancels any passwords that are guessed and notifies the user.	Notified if password is easy to crack more often? Faster user signup and less user confusion with password requirements?	First, it is resource intensive if the job is done right. Because a determined opponent who is able to steal a password file can devote full CPU time to the task for hours or even days, an effective reactive password checker is at a distinct disadvantage. Furthermore, any existing passwords remain vulnerable until the reactive password checker finds them
Proactive password checking	a user is allowed to select his or her own password. However, at the time of selection, the system checks to see if the password is allowable and, if not, rejects it	with sufficient guidance from the system, users can select memorable passwords from a fairly large password space that are not likely to be guessed in a dictionary attack	Must strike a balance between user acceptability and strength. If the system rejects too many passwords, users will complain that it is too hard to select a password. If the system uses some simple algorithm to define what is acceptable, this provides guidance to password crackers to refine their guessing technique.

Security Issues for User Authentication

Following Diagram is what he said we should learn:



Attacks	Authenticators	Examples	Typical defenses
Client attack	Password	Guessing, exhaustive search	Large entropy; limited attempts
	Token	Exhaustive search	Large entropy; limited attempts, theft of object requires presence
	Biometric	False match	Large entropy; limited attempts
Host attack	Password	Plaintext theft, dictionary/exhaustive search	Hashing; large entropy; protection of password database
	Token	Passcode theft	Same as password; 1-time passcode
	Biometric	Template theft	Capture device authentication; challenge response
Eavesdropping, theft, and copying	Password	"Shoulder surfing"	User diligence to keep secret; administrator diligence to quickly revoke compromised passwords; multifactor authentication
	Token	Theft, counterfeiting hardware	Multifactor authentication; tamper resistant/evident token
	Biometric	Copying (spoofing) biometric	Copy detection at capture device and capture device authentication
Replay	Password	Replay stolen password response	Challenge-response protocol
	Token	Replay stolen passcode response	Challenge-response protocol; 1-time passcode
	Biometric	Replay stolen biometric template response	Copy detection at capture device and capture device authentication via challenge-response protocol
Trojan horse	Password, token, biometric	Installation of rogue client or capture device	Authentication of client or capture device within trusted security perimeter
Denial of	Password, token,	Lockout by multiple	Multifactor with token

Two Factor Authentication

- Apple
- How it works:
    - Your account can only be accessed on devices you trust, like your iphone, ipad, or mac. When you want to sign in to a new device for the first time, you'll need to provide two pieces of information - password and six-digit verification code that's automatically displayed in your trusted devices. By entering the code, you're verifying that you trust the new device.
- Improves general security as you now don't just need a password to log in.

Other Approaches to Authentication

- Graphical Passwords
    - Using faces as identifiers
- CAPTCHA - Completely Automated Public Turing test to tell Computers and Humans Apart
    - Often Complements Text-Based password
        - Strengths security in user authentication
        - Reduce likelihood of service abuse - bot accounts
- Tools
    - Fingerprint
    - Retina
    - Face

Authentication Challenges

- Not necessarily limited to passwords
- Remote user authentication is another challenge
- Potential attacks include:
    - Client
    - Host
    - Eavesdropping, theft, copying
    - Replay
    - Trojan horse
    - DDOS

Simplified User Authentication State

- Based on user's image selection, and text responses, there are several different cases to consider

Preventive Actions

- It is crucial to effectively estimate, based on patterns of failed login attempts
    - Which "element" of the password might have been compromised
    - System might choose of ask text password associated with specific image

Automated Attackers

- Bots are assumed to be equipped with deep learning algorithms,access to powerful search engines, "private" database on actions taken on past failures, etc.
- Most powerful and nasty attack scenarios imaginable are to be investigated

Current Status

- Implement a prototype system and perform preliminary user study
    - Ease of use
    - Successful Recall: 1. Image and 2. Texts
    - Failure Analysis

- Simulate “social engineering” attacks
- Simulate automated/nasty attacks

Conclusions

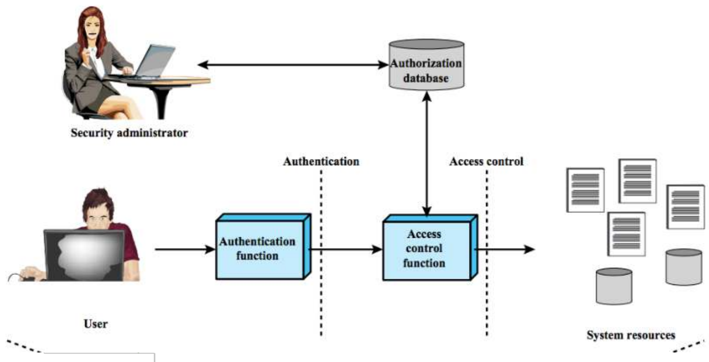
- Combine the best of both image and text based authentications
    - Built-in password attack and leak detection mechanisms via trap images
    - Highly personalised association between user’s own images and text labels
    - Easier to create and remember passwords on multiple sites
- Guidelines to enhance security strength and defeat face recognition algorithms?
- Plan to develop to become secure, scalable, easy-to-use two factor authentication

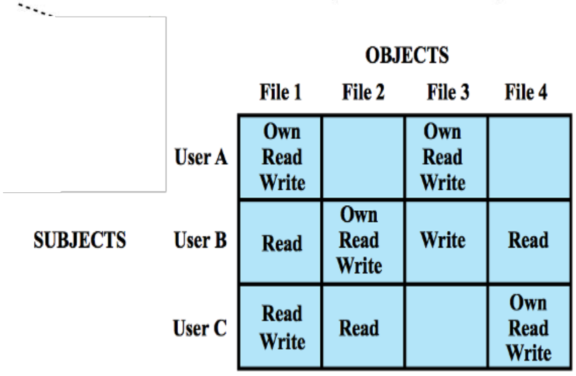
Chapter 4 - Access Control

Access Control

- Subject, Object
- DAC (Discretionary Access Control)
- MAC (Mandatory Access Control)
    - Deployed primarily in military applications
    - Applies to OS, DBMS, etc
    - Bell La-Padula MLS (Multi-Level Security)
- Role-based Access Control

Authentication vs Access Control - Access Control Matrix





		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

(a) Access matrix

Subject, Object, and Access Rights

- Subject: An entity capable of access objects
- Object: A resource to which access is controlled
    - Further divided into:

        - Owner
        - Group - user's who are in file's 'group'
        - Everybody else (except sudo)
- Access Right: Describes the way in which a subject may access an object (read, write, execute)
- It is essential to accurately and completely identify all the subjects, objects, and access rights
    - Read, Write, Execute, Delete, Create, Search, Append, Changing permissions

DAC in Unix

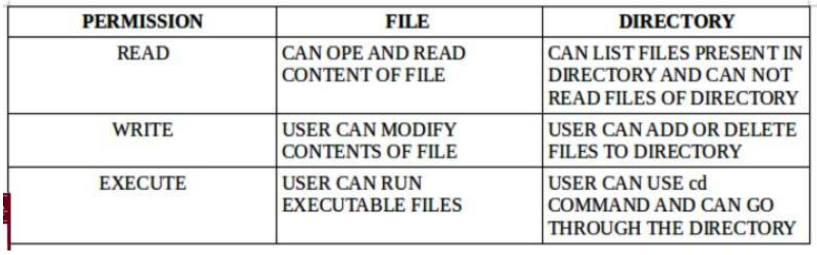
- Commonly known as file permission
    - chmod() - three numbers which define user, group world permissions as binary - table below shows the different totals based on permissions:

read	write	execute	Total Value
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Unix File Permission

- Some file permission strings might be "valid or legal" but may make little logical sense
    - - --- --- --- 0,0,0
        - Who can do anything now?
        - Can sudo do anything?
        - Answer: root can do anything but execute the file (outside removing the file if the file-system is mounted read-only or the file has some immutable flag set).
          Non root users might change the file permission if they own it. They can still access the file if ACLs are set to allow it.
        -
    - - --- rwx ---
        - Is the owner also a group member?
        - If the owner is also the group member, they are not allowed to rwx as they owner bit specifically disallows them access to them, which takes priority.
    - - --- --- rwx
        - What is the precise definition of "other"?

        - Means that neither the owner or group members can read, write, or execute the file, while everyone else can do all three.
- What does rwx mean when applied on directories?



PERMISSION	FILE	DIRECTORY
READ	CAN OPE AND READ CONTENT OF FILE	CAN LIST FILES PRESENT IN DIRECTORY AND CAN NOT READ FILES OF DIRECTORY
WRITE	USER CAN MODIFY CONTENTS OF FILE	USER CAN ADD OR DELETE FILES TO DIRECTORY
EXECUTE	USER CAN RUN EXECUTABLE FILES	USER CAN USE cd COMMAND AND CAN GO THROUGH THE DIRECTORY

Superuser

- A.k.a root has an effective UID 0
    - Process control
        - Change priority of any process
        - Set “hard limits” on resource usage
    - Device Control
        - Access any working device, shutdown or reboot, read or modify any memory location, create new device
    - Network Control
        - Reconfigure network, run network services, etc.
    - Filesystem control
        - Read, modify, or delete any file or program
        - Add, create,or change user account
        - Mount of unmount filesystem
- Can’t perform certain tasks such as:
    - Make a change to a filesystem mounted as read-only
    - Unmount a filesystem that contains open files
    - Decrypt the password stored in the shadow password file
- Superuser is the main security weakness in UNIX operating system
    - Some restrictions can be enforced
    - Sulog is often provided
- Any user can become superuser with /bin/su program

Subject & Object: Not-so-apparent

- Processes, Files, Directories : no brainer
- Can processes be objects too?
- Hard link? Symbolic link?
- Device files?
- How about memory, buffer, etc?
    - Becomes more complicated when both DAC and MAC are in place

Hard Links and Symbolic Links

- A Unix file is “stored” in two different parts of the disk - the data blocks and the inodes.
- The data blocks contain the “contents” of the file.
- Information about the file is stored elsewhere - in the inode. Inode is a file structure on a file system. Contains all file information but file contents and file name.
- Symbolic link contains a reference to another file or directory in the form of an absolute or relative path
    - Must possess in depth knowledge to perform security analysis.

- Hard Link associates a name with a file on a file system. All directory-based file systems must have at least one hard link giving the original name for each file.

SUID and SGID

- Mechanism to allow unprivileged users to perform tasks that require privileges
  - Without ability to directly modify /etc/passwd file
- UNIX allows programs to be endowed with privileges. Processes executing these programs can assume another UID or GID when they are running.
  - When a SUID program is run, its effective UID becomes that of the owner of the file, rather than the user who is running it.
- Setuid and setgid - allows for temporary elevated privileges in order to perform a specific task - by using owner's id or group id to access

chown and chgrp

- File's owner and group can also be changed
- Use commands such as chmod, chown, chgrp with caution
- Semantics may differ from OS to OS.

Access Control Lists

- List of access control entries defining a trustee and their specific access rights
- Must always check default setting to see what users are given at the start
  - Review trade-offs and decide
  - Systems are likely to choose similar setting, but you must verify to be sure

Mandatory Access Control (MAC)

- the operating system constrains the ability of a *subject* or *initiator* to access or generally perform some sort of operation on an *object* or *target*.
- In practice, a subject is usually a process or thread; objects are constructs such as:
  - files,
  - directories,
  - TCP/UDP ports,
  - shared memory segments,
  - IO devices
- Subjects and objects each have a set of security attributes.
- Whenever a subject attempts to access an object, an authorization rule enforced by the operating system kernel examines these security attributes and decides whether the access can take place.
- Any operation by any subject on any object is tested against the set of authorization rules (aka *policy*) to determine if the operation is allowed.

MLS and Bell-LaPadula

- MLS (Multi-level security)
  - Application of a computer system to process information with incompatible classifications, permit access by users with different security clearances and needs-to-know, and prevent users from obtaining access to information for which they lack authorization.
  - SS (Simple Security) Property - a subject at a given security level may not read an object at a higher security level - "no read up"
  - The * (star) property indicates that a subject at a given security level may not write to any object at a lower security level - "no write down"
  - Discretionary Security Property - use of an access matrix to specify the discretionary access control (DAC)
- MLS Policy Definition
  - Generally "No read up, no write down"
  - Subtle variations may exist from one implementation to another

        - Is it a security violation to allow “write up”?
        - Covert Channels - secret channels
- Bell-LaPadula
    - Users can create content only at or above their own security level
    - Users can view content only at or below their own security level

Security Levels

- Subjects and Objects are labelled with SLs, which are composed of two types of entities:
    - Sensitivity - A hierarchical attribute like “Secret” or “Top Secret”
    - Categories - Set of non-hierarchical attributes like “US Only” or “UFO”
- Must have one sensitivity and zero or more categories
- E.g. {Secret / UFO, Crypto}, {Top Secret / UFO, Crypto, Stargate}
- E.g. Can a process running with a clearance of {Top Secret / UFO, Rail Gun} write to a file classified as {Top Secret, UFO}?
    - YES. SL1 > SL2 as SL1 has more categories than SL2. SL2 is a subset of SL1.

DAC and MAC

- **Read textbook**
- Ds-property
    - An individual may grant to another individual access to a document based on the owner’s discretion, constrained by the MAC rules
    - In other words, both DAC and MAC tests must pass to access an object

;Role-based Access Control

- RBAC based on roles that users assume in a system rather than the user’s identity
    - Roles can be assigned statically or dynamically
- DAC and RBAC are NOT, IMHO, fundamentally different
- RBAC is not widely used.

Chapter 6 - Malware

Malware

- “Malicious Software”
- Harmful or intrusion software
- Computer Viruses, worms, trojans, ransom ware, rootkits, spyware, botnets
- Takes form of: executable code, scripts, …..
- Defined my malicious intent
- [SOUP13] Definition: Program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system or otherwise annoying or disrupting the victim
- We are concerned with:
    - Threat malware poses to:
        - Applications programs
            - Editors
            - Compilers
            - Kernel-Level Programs
        - Use on compromised or malicious websites and servers
        - Use in especially crafted spam e-mails or other messages which aim to trick users into revealing sensitive personal information

Terminology

Advanced Persistent Threat	Cybercrime directed at business and political targets - using intrusion technologies and malware - applied to specific targets over an extended period - often attributed to state-sponsored organisations
Adware	Advertising integrated into software - pop ups or redirects
Attack Kit	Set of tools for generating new malware automatically using supplied propagation and payload mechanisms
Auto-rooter	Malicious hacker tools used to break into machines remotely
Backdoor	Mechanisms that bypass normal security check - allows access to functionality in a program or a compromised system
Downloaders	Code that installs other items on a machine that is under attack - normally included in the malware code first inserted onto a compromised system to then import a larger malware package
Drive-by Download	Attack using code in a compromised web site that exploits a browser vulnerability to attack a client system when the site is viewed.
Exploits	Code specific to a vulnerability/vulnerabilities
Flooders	Used to generate large volume of data to attack networked computer systems - DDoS attack
Keyloggers	Captures keystrokes on a system
Logic Bomb	Code inserted into malware by an intruder - lies dormant until a predefined condition is met - code then triggers an unauthorized act
Macro Virus	Uses macro or scripting code - embedded into a document - triggers when document is viewed or edited - to run and replicate itself into other such documents
Mobile Code	Software that can be shipped unchanged to heterogeneous collection of platforms and execute with identical semantics
Rootkit	Set of hacker tools used after attacker has broken into a computer system and gained root-level access
Spammer programs	Used to send large volumes of unwanted email
Spyware	Software that collects information from a computer and transmits it to another system by monitoring keystrokes, screen data and/or network traffic; or by scanning files on the system for sensitive information
Trojan Horse	Program that appear to have a useful function but has a hidden and malicious function that evades security mechanisms, sometimes by exploiting legitimate authorizations of a system entity that invokes the

	Trojan Horse program
Virus	Malware that tries to replicate itself into other execute machine or script code; when it succeeds the code is said to be infected. When the infected code is executed , the virus also executes
Worm	Program that can run independently and can propagate a complete working version of itself onto other hosts on a network, usually by exploiting software vulnerabilities in the target system
Zombie, bot	Program activated on an infected machine that is activated to launch attacks on other machines

Notable Historic Malware

- CIH Virus - 1998
- Melissa Worm - 1999
- Code Red Worm - 2001
- Slammer Worm - 2003
- SoBig.F Worm - 2003
- My Doom Worm - 2004
- Stuxnet Worm - 2010
- Cryptolockler Trojan - 2013
- ZeroAccess Botnet - 2013
- Superfish Adware0 - 2014
- Locky Ransomware - 2016
- WannaCry RansomWare - 2017

Rise of Attacks

- Attack Sophistication vs Intruder Tech Knowledge

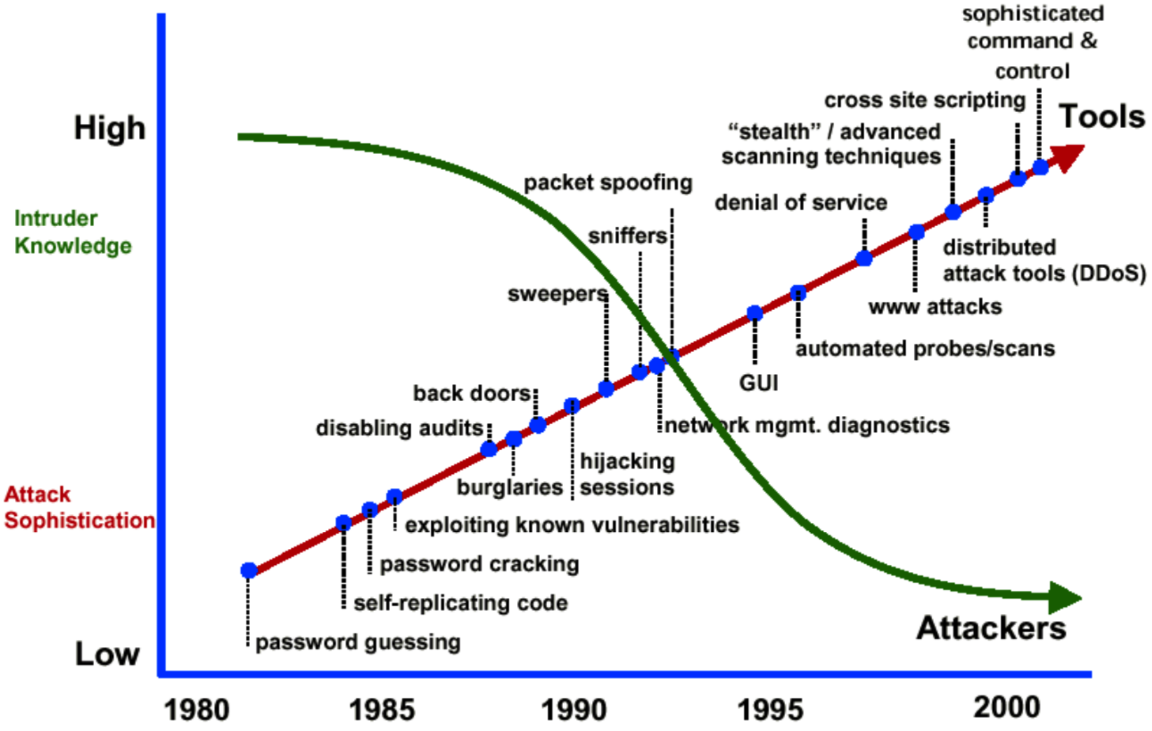
Factors Affecting Attack Trend

- Increased use of the Internet
- Increasing software complexity
- Abundance of attack tools - increasing sophistication and complexity
- Increased use of broadband home access
- Slow adoption of good security practices

Microsoft Prevention Strategies

- Outlook prevents many file types from being sent (e.g. .app, .exe, .js, .com ……..)

- Save the file to the cloud



- Use a file compression utility (WinZip)
- Rename the file to use an extension that doesn't block

Understanding the Threat

- Attackers are not a monolithic group of people
- They can be categorized based on types of attacks and capabilities
- Attack Tools are easy to find

Attacker Types

Script Kiddies	• Use crime kits to make spending money • Little to no business or technical expertise
Gray-Hats	• Believe that are offering legitimate services - however their customers can be both "legitimate" and criminal • Ran as a business
Black-Hats	• Threats cybercrime as a business • Business and technical expertise • Often works in a closed group of other professional cybercriminals • Criminal reputation is everything
Hactivists	• Individuals or groups who hack for a social cause, without economic motivation • Has both technical people and minions
State Sponsored	• National security and/or economic motivation • Technical expertise

	• Work in a closed group of other professionals • Often uses Black-Hat resources and/or techniques to make their identity

Keylogging

- Piece of software or hardware that can intercept and record the keystrokes of a compromised machine. Digital tap that captures every keystroke from the keyboard
- Often the keylogger function is embedded in another piece of malware

RootKits

- Originally created to obtain root privilege on a system as well as hide elements such as processes, files, and network connections
- With the advent of spyware, rootkits have been designed to hide the aforementioned elements with the specific intent of remaining residents
- Rootkits can be:
    - Persistent
        - Execute each time system starts or user logs in
        - Memory-based
        - Does not survive reboot
        - User-mode
        - Intercepts commands such as file listing
        - Kernel-model
        - Hide from kernel list of active processes

Attack Kits

- Super Easy to get if you search on the web
- From the table above - Set of tools for generating new malware automatically using supplied propagation and payload mechanisms

Detection of Rootkits

- Possible to hide spyware or virus that will not be detected by traditional antivirus products
- E.g.
    - F-Secure blackLight RootKit Eliminator
    - Published Rootkits
    - Exposure Example
    - SONY BMG copy protection rootkit scandal
        - First4Internet XCP copy protection software
        - Design flaw in Sony's web-based uninstaller
            - CodeSupport to download and run code from URL

Payload - (Stealthing) Rootkit

- Set of hidden programs installed on a system to maintain covert access to that system
- Hides by subverting the mechanisms that monitor and report on the process, files, and registries on a computer
- Gives administrator (or root) privileges to attacker
    - Can add or change programs and files, monitor processes, send and receive network traffic, and get backdoor access on demand

"Macro" Virus

- Automated Scalping Bots often buying tickets to events and scalping other ticket buyers
  (what the fuck is this definition lol)

virus written in the same **macro** language used for software programs, including Microsoft Excel or word processors such as Microsoft Word. When a **macro virus** infects a software application, it causes a sequence of actions to begin automatically when the application is opened.

Worms

- Program that actively seeks out more machines to infect and each infected machine serves as an automated launching pad for attacks on other machines
- Exploit software vulnerabilities in client or server programs
- Can use network connections to spread from system to system
- Spreads through shared media
- Email worms spread in macro or script code included in attachments and instant messenger file transfers
- Upon activation the worm may replicate and propagate again
- Usually carries some form of payload
- First know implementation was done in Xerox Palo Alto Labs in the early 1980s

Top 10 Worst Computer Worms/Viruses

- Jerusalem (Blackbox) - 1987
- Michelangelo - 1991
- Storm Worm - 2008
- Sobig - 2003
  - Exploited open proxy servers to turn infected machines into span engines
  - MSBlast - 2003
  - Melissa - 1998
  - Email worm
  - First to include virus, worm and trojan in one package
  - Code Red - 2001
  - Exploited Microsoft IIS bug
  - Probes random IP addresses
  - Consumes significant Internet Capacity when active
  - Code Red 2 - 2001
  - Microsoft IIS
  - Installs a backdoor for access
  - Nimda - 2001
  - Worm, virus and mobile code characteristics
  - Spread using email, windows shares, web servers, web clients and backdoors
  - SQL Slammer - 2003
  - Exploited a buffer overflow vulnerability in SQL Server
  - Compact and spread rapidly
  - ILOVEYOU - 2000
  - Mydoom - 2004
  - Mass mailing email worm
  - Installed a backdoor in infected machines
  - Warezov - 2006
  - Creates executables in system directories
  - Sends itself as an email attachment
  - Can disable security related products

    - Conflicker (Downadup) - 2008
    - Exploits a Windows buffer overflow vulnerability
    - Most widespread infection since SQL Slammer
    - StuxNet - 2010
    - Restricted rate of spread to reduce chance of detection
    - Targeted industrial control systems
    - Morris Worm - 1988
- The Concept Virus - 1995
- CIT (Chernobyl Virus)
- Anna Kournikova Worm
- Blaster Worm
- Netsky and Sasser
- OSX.R SPlug Trojan - 2007

What can virus or malware do?

- Pretty much anything!
- Real-world viruses are much more sophisticated and dangerous
    - May have complex mechanisms to evade detection

Malware 'Patterns'

- There is no such thing, but many share similar characteristics
- Significant variations on structure, complexity, vulnerabilities being exploited, what it does, propagation patterns, hiding mechanisms, ….

Morris Worm

- Earliest significant worm infection
- Robert Morris - 1988
- Designed to spread on UNIX Systems
    - Cracked local password file to use login/password to logon to other systems
    - Exploited a bug in the finger protocol which reports the whereabouts of a remote user
    - Exploited a trapdoor in the debug option of the remote process that receives and sends mail
    - Successful attacks achieved communication with the operating system command interpreter
    - Sent interpreter a bootstrap program to copy worm over

Virus Protection

- Effective protection is anti-virus software which:
    - Scans email attachments
    - Checks for virus signatures
    - E.g.
    - Norton
    - McAfee
    - V3

Worm Replication

Electronic Mail or Instant Messenger Facility	• Worm emails a copy of itself to other systems • Sends itself as an attachment via an instant message service

File Sharing	• Creates a copy of itself or infects a file as a virus on removable media
Remote execution capability	• Worm executes a copy of itself on another system
Remote file access or transfer capability	• Worm uses a remote file access or transfer service to copy itself from one system to another
Remote login capability	• Worm logs onto a remote system as a user and then uses commands to copy itself from one system to another

E.g. Slammer Worm (W32.Slammer)

- Aliases: SQL Slammer, Sapphire, W32.SQLExp.Worm
- Released: January 25, 2003 - 5.30am
- Fastest worm in history: Spread world-wide in under 10 mins
- Doubled infections every 8.5 seconds
- Size: 376 bytes long

Worm Countermeasures

- Considerable overlap in techniques for dealing with viruses and worms
- Once a worm is resident on a machine anti-virus software can be used to detect and possible remove it
- Perimeter network activity and usage monitoring can form the basis of a worm defense
- Worm defense approaches include:
    - Signature-based worm scan filtering
    - Filter-based worm containment
    - Payload-classification-based worm containment
    - Threshold random walk (TRW) scan detection
    - Rate limiting
        - Used to control rate of traffic sent or received by a network interface controller and is used to prevent DDoS attacks
        - Rate halting

Buffer Overflow

- Trying to fill a buffer beyond its capacity in C
- Specially crafted malicious input values adapted to the architecture and environment could yield to arbitrary code execution
- Fixes:
    - Make sure that the memory auditing is done properly in the program
    - Use fgets() instead of gets()
    - Use strncmp() instead of strcmp(), strncpy() instead of strcpy() and so on.

Vulnerable Code

- strcpy()
    - Does not check buffer lengths and may very well overwrite memory zone contiguous to the intended destination
    - Mitigation:
        - Use str|copy if it is readily available or define it yourself
    - sprintf()
        - Does Not check the buffer boundaries and is vulnerable to overflows

        - Mitigation:
        - Prefer using snprintf, which has the double advantage of preventing buffer overflows and returning minimal size of buffer needed to fit the whole formatted string.

Top 10 Secure Coding Practices

- Validate Input
- Heed Compiler Warnings
- Architect and design for security policies
- Keep it simple
- Default deny
- Adhere to the principle of least privilege
- Sanitize data sent to other systems
- Practice defense in depth
- Use effective quality assurance techniques
- Adopt a secure coding standard
- Define security requirements
- Model threats

Social Engineering

- Persuade someone to disclose sensitive information
- Persuade someone to run/install malicious or subverted software
- Invite someone to log into a bogus website such as a spoofed bank web site
- Impersonating new employee who has forgotten userid/password
- Impersonating a technical support staff member and requesting a user login to ‘check’ accounts
- “Tricking” users to assist in the compromise of their own systems
- E.g.
    - Spam
        - Bulk Email
        - Malware
        - Phishing attacks
        - Trojan Horse
        - Program containing harmful hidden code
        - Used to accomplish functions that the attacker could not accomplish directly
        - Mobile Phone Trojans
        - 2004
        - Target is smartphone

Phishing

- Electronic Fishing
- Mass distribution of ‘spoofed’ email
- Baits users to malicious content
- Appears to come from banks, insurance agencies, retailers or credit card companies
- Email messages that guide recipients to legitimate-looking but fake websites and try to get them to supply personal info like passwords
- Look “official” - up to 5% of recipients may respond, resulting in financial losses, theft
- Easy to identify once its true identity is known, but can do real harm to some people
- Senior citizens more vulnerable

Typosquatting

- Aka URL hijacking or fake URL
- Relies on mistakes such as typos made by Internet users when inputting a website address into a web browser

- When they enter the wrong address - they get lead to malicious fake site
- One of five types:
    - Common Misspelling - e.g. foreign language
    - A misspelling based on typos - e.g. examlpe.com
    - A differently phrased domain name - e.g. examples.com
    - A different top-level domain - e.g. example.org
    - Abuse of Country Code Top-Level Domain - i.e. something that is not your country code
    - Stay alert - remain justifiably suspicious
    - Fake site may prompt you to download additional software intended to "enhance security"

Pharming

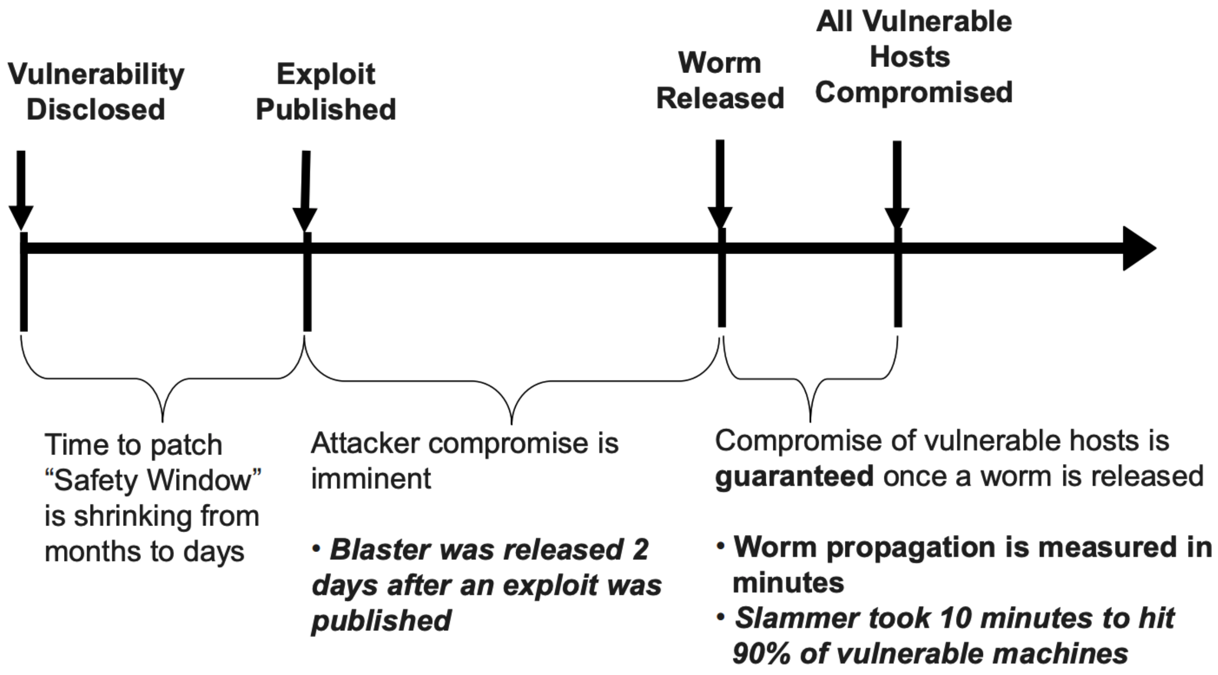
- Hacker's attack aiming to redirect a website's traffic to another bogus website
- Even though browser seems to be displaying web address you wanted to visit
- Tampers with DNS system so that traffic to a website is secretly redirected to a different site altogether
- Become of major concern to businesses hosting Ecommerce and online banking websites

Ransomware

- E.g. 'WannaCry'
- Similar to a worm, compromises hosts, encrypts files stored on ports then demanding a ransom payment in the form of bitcoin.
- " threatens to publish the victim's data or perpetually block access to it unless a ransom is paid"

New Trend - "Zero-day" Attacks

- Takes advantage of software vulnerability for which there are no available fixes
- Take advantage of flaws before software makers can fix them
- 2008+
- Emphasizes importance of safe configuration policies and good incident reporting systems
- Hackers are getting faster at exploiting flaws
- 2003 Blaster Worm
    - Hit the internet barely a months after microsoft released a patch for the flaw it exploited
    - Took EIGHT MONTHS to appear after vulnerability it targeted was disclosed
    - Zero day gets closer!



Time to patch "Safety Window" is shrinking from months to days

Attacker compromise is imminent

- ***Blaster was released 2 days after an exploit was published***

Compromise of vulnerable hosts is **guaranteed** once a worm is released

- **Worm propagation is measured in minutes**
- ***Slammer took 10 minutes to hit 90% of vulnerable machines***

Bug Bounty Program

- Deal offered by many websites and software developers by which individuals can receive recognition and compensation for reporting bugs, especially those pertaining to exploits and vulnerabilities

Software Testing and Malware

- Black-box Testing
  - What can it detect? - Only surface level bugs - Front End bugs
  - White-Box Testing
  - What can it detect? - Code-level bugs - Deepest level

Malware and Code Obfuscation

- Obfuscation
  - Deliberate act of creating source or machine code that is difficult for humans to understand
  - Used to conceal code purpose or logic or implicit values embedded to prevent tampering and deter reverse engineering
  - Control Flow Flattening
  - Changing lines to code to have exact same operation and output but appear much differently with a much more complex approach potentially appearing to behave differently
  - Binary Obfuscation

Chapter 7 - Denial of Service Attacks

Denial of Service

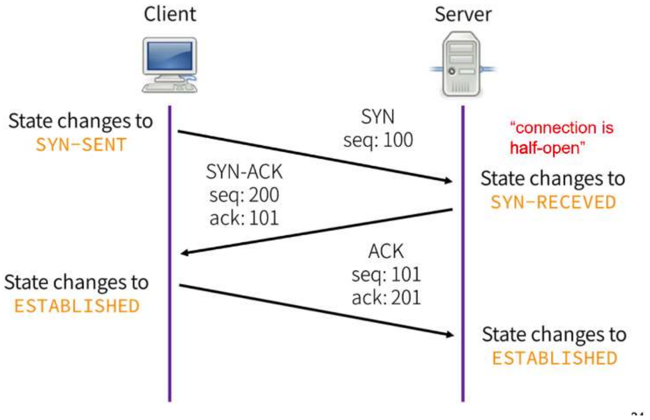
- A form of attack on the availability of a service
- Prevents or impairs authorized use of networks, systems, or applications by exhausting resources such as CPU, memory, bandwidth and disk space

- Resources that can be attacked:
    - Network bandwidth
        - Relates to capacity of the network links connecting a server to the internet
        - This is connection to ISP
        - System resources
        - Aims to overload or crash network handling software
        - Application resources
        - Involves a number of valid requests - each consume significant resources - limiting the ability of the server to respond to requests from other users
    - Popular DoS attacking tools:
        - LOIC (Low Orbit Ion Cannon)

Flooding Attacks

- Classified based on network protocol used
- Intends to overload network capacity on some link to a server
- Any type of network packet can be used
- Types:
    - **ICMP Flood**
        - Ping flood using ICMP echo request packets
        - Traditionally network admins allow such packets into their networks because ping is a useful network diagnostic tool
        - Aim is to overwhelm capacity of network connection to the target org.
        - Traffic can be handled by higher capacity links on the path, but packets are discarded as capacity decreases
        - Source of the attack is clearly identified unless a spoofed address is used
        - Network performance is noticeably affected
        - **UDP Flood**
        - Uses UDP packets directed to some port number on the target system
        - Sends a large number of packets to random ports of a remote host
        - Host checks:
            - For application listening on that port
            - Sees that no application is listening on that port
            - Replys with ICMP Destination Unreachable packet
            - For many packets, the host has to send an excessive amount of ICMP response packets back, flooding the network.
            - Sender can also spoof IP address ensure that response packets will not return
            - Mitigated by using a rate limiting service on the response packets
        - **TCP SYN Flood**
            - Sends TCP packets to the target system
            - Total volume of packets is the aim of the attack rather than the system code
            - Host becomes so overwhelmed with SYN segments, which initiate incomplete connection requests, that it can no longer process legitimate connection requests
            - Fills the memory buffer of the victim
            - Host can then no longer process new TCP connection requests

        - Solution: impose a limit on the number of SYN segments that are permitted to pass through the firewall per second



        - Store backed up SYN requests in a queue
        - TCP 3-way handshake

    - **Teardrop Attack**
        - Conducted by targeting TCP/IP fragmentation reassembly codes.
        - Causes fragmented packets to overlap one another on the host receipt
        - Host attempts to reconstruct them during the process but fails.
        - Gigantic payloads are sent to the machine that is being targeted, causing system crashes.
    - Naive and brute-force attack is unlikely to be successful if adequate OS patches or updates have been implemented

Source Address Spoofing

- Use forged source addresses
    - Via the raw socket interface on operating systems
    - Makes attacking systems harder to identify
    - Attacker generates large volumes of packets that have the target system as the destination address
    - Congestion would result in the router connected to the final, lower capacity link
    - Requires network engineers to specifically query flow information from their routers
    - Backscatter traffic
    - ICMP echo response packets generated in response to a ping flood using randomly spoofed source addresses is a good example
    - Advertise routes to unused IP addresses to monitor attack traffic
    - **TCP SYN Spoofing**
    - Common DoS attack
    - Attacks the ability of a server to respond to future connection requests by overflowing the tables used to manage them, thus legitimate users are denied access to the server
    - Attacker directs a very large number of forged connection requests at the targeted server
        - Rapidly fills known TCP connections table on the server
        - Once full, any future requests, including legitimate requests from other users, are rejected
        - Table entries will timeout and be removed, which in normal network usage corrects temporary overflow problems

        - However, if attacker keeps a sufficient volume of forged requests following, this table will be constantly full and the server will be effectively cut off from the internet, unable to respond to most legitimate connection requests

Privacy, Anonymity, Freedom from Tracking/Censorship, Security, etc..

- When it comes to malware and cyber attacks, there could be conflict with each other
    - Technical solutions alone will likely be insufficient
    - Often good to use many together however there may be conflicts

Anonymous Web Browsing

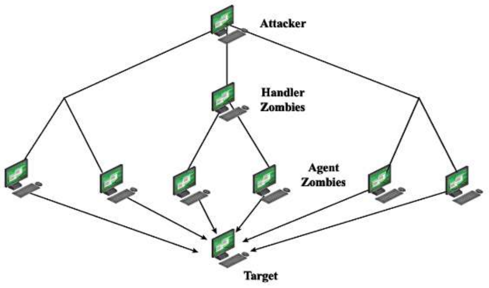
- Utilization of the web that hides a user's personally identifiable information from websites visited.
- Done using:
    - Proxy Servers
    - VPNs
    - Tor
    - These programs send info through a series of routers in order to hide the source and destination information
    - Never a guarantee of anonymity - programs still susceptible to traffic analysis

Tor

- Free software for enabling anonymous communication
- Directs traffic through a worldwide overlay network consistent of more than 7000 relays
- Conceals user location and usage from anyone conducting network surveillance or traffic analysis

DDoS

- Use of multiple systems to generate attacks
- Attacker uses a flaw in operating system or in common application to gain access and installs program on it
- Large collection of such systems under the control of one attacker's control can be creating, forming a botnet



Bots and Botnets

- A bot is a software agent which interacts with other services intended for people as if it were a real person
- Typical bot use is information gathering
- A botnet is a collection of bots which run autonomously
    - Usually a collection of compromised machines running worms, trojans, backdoors, etc.

DoS Attack Defenses

- Attacks cannot be prevented entirely
- High traffic volumes may be legitimate
    - Very popular site?
    - Flash event?
    - High publicity at once?
    - Four lines of defense
    - Attack prevention and preemption - Before attack
    - Attack detection and filtering - During attack
    - Attack source traceback and identification - during and after attack
    - Attack reaction - after attack
    - Mitigation
    - Identify normal conditions for network traffic - "traffic patterns" - necessary for threat detection and alerting
    - Also requires identifying incoming traffic to separate human traffic from bots and hijacked web browsers - done by comparing signatures and examining different attributes of the traffic
    - Then filter packets based on above criteria
    - Best practices:
        - Create a DDoS Response Team and Plan
        - Maintain continuous vigilance
        - Protect your DNS servers
        - Learn how to detect and understand a DDoS attack

Dos Attack Prevention

- Block spoofed source addresses
- Filters may be used to ensure path back to the claimed source address is the one being used by the current packet
    - Filters must be applied in traffic before it leaves the ISP's network or at the point of entry to their network
    - Use modified TCP connection handling code
    - Cryptographically encode critical information in a cookie that is sent as the server's initial sequence number
        - Legitimate client responds with an ACK packet containing the increment sequence number cookie
        - Drop an entry for an incomplete connection from the TCP table when it overflows
    - Block IP directed broadcasts
    - Block suspicious services and combinations
    - Manage application attacks with a form of graphical puzzle (captcha) to distinguish legitimate human requests
    - Good general system security practices

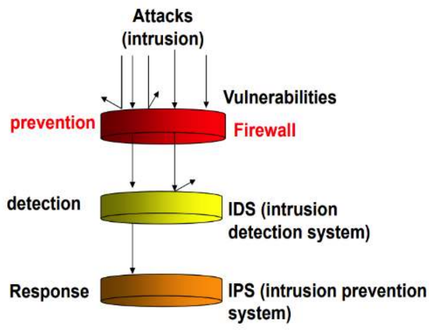
    - Use mirrored and replicated servers when high-performance and reliability is required

Responding to DoS Attacks

- Good Incident Response Plan
    - Details on how to contact technical person for ISP
    - Need to impose traffic filtering upstream
    - Details of how to respond to the attack
    - Anti Spoofing, directed broadcast, and rate limiting filters should have been implemented
    - Ideally have network monitors and IDS to detect and notify abnormal traffic patterns
    - Process:
    - Identify type of attack
        - Capture and analyse packets
        - Design filters to block attack traffic upstream
        - Identify and correct system/application bug
        - Have ISP trace packet flow back to source
        - May be difficult and time consuming
        - Necessary if planning legal action
        - Implement contingency plan
        - Switch to alternate backup servers
        - Commission new servers at a new site with new addresses
        - Update incident response plan
        - Analyse the attack and the response for future handling

Chapter 8/9 - IDS, IPS & Firewall

Overview



Intrusion and IDS

- IDS
  - An app which detects attacks against your computer or network and lets you know when they occur
  - Combination of software and hardware attempts to perform “intrusion detection”
  - Raises alarm when possible intrusion or suspicious patterns are observed
  - Benefits
    - Detects Attacks - notifies about compromised systems
    - Enforces policies - IDS can monitor an internal network for behavior that violates your organization's network security or acceptable use policies
    - Providing an audit trail - IDS can provide post-mortem audit trail for seeing how far an attacker got, and where it came from
    - Resource justification - IDS can provide how well your firewall is working and how many people are trying to intrude
    - Requirements
    - High Detection Rate with low false alarms
    - Minimum overheads to computer system and IDS itself for processing audit data

Classification of IDS

- **Intrusion Detection Model**
  - **Misuse detection (signature-based, knowledge-based)**
    - Use patterns of well-known attacks to identify intrusions
    - Records the specific patterns of intrusions
    - Monitor current audit trails and pattern matching
    - Report the matched events as intrusions
    - It is heuristics based <- I think maybe this stuff should be in anomaly based bc it is to do with weird behaviour Steve describes it as heuristics based in the lecture slides (slide 9)
      - Examples of uncommon activities
        - read files in other users’ personal directories
        - writing to other user’s files
        - logging in after hours often access the same files they used earlier
        - open disk devices directly but rely on higher-level OS utilities
        - logged in more than once to the same system
        - make copies of system programs
    - **Anomaly detection (behaviour-based)**
      - Use deviation from normal usage patterns to identify intrusions
      - Establish normal behaviour profiles
      - Observing and comparing current activities with the normal profiles
      - Reporting significant deviations as intrusions
      - Statistical measures as behaviour profiles
      - Disadvantage
        - Relatively high false positive rate - anomalies can just be new normal activities
  - **Data Source (input - audit data)**
    - **Host Based Detection**
      - Command logs and system calls
    - **Network Based Detection**

    - Using Network packet header, contents, traffic, etc
- **Hybrid**

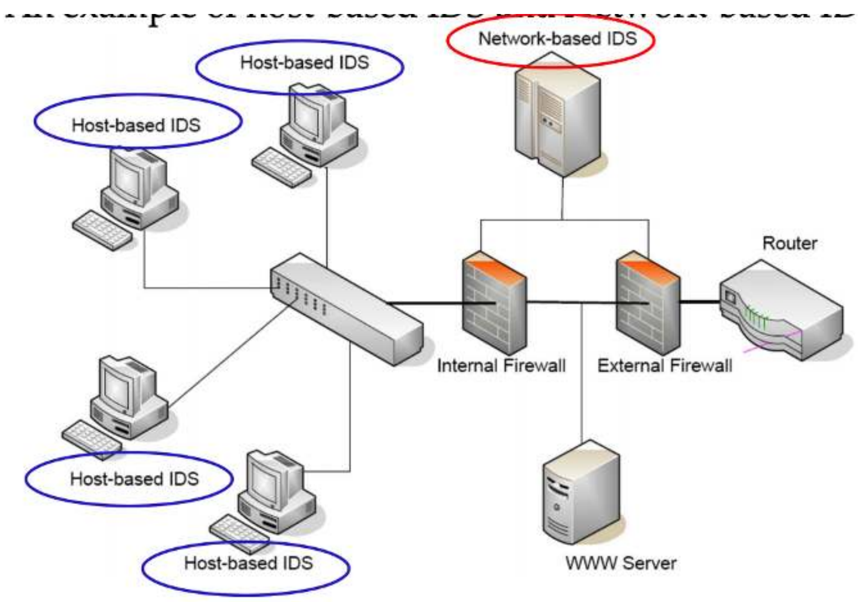
Measures that May Be Used for Intrusion Detection

Measure	**Model**	**Type of Intrusion Detected**
	Login and Session Activity	
Login frequency by day and time	Mean and standard deviation	Intruders may be likely to log in during off hours
Frequency of login at different locations	Mean and standard deviation	Intruders may log in from a location that is a particular user rarely or never uses.
Time since last login Elapsed time per session	Operational Mean and standard deviation	Break-in on a "dead" account. Significant deviations might indicate masquerade.
Quantity of output to location	Mean and standard deviation	Excessive amounts of data transmitted to remote locations could signify leakage of sensitive data.
Session resource utilization	Mean and standard deviation	Unusual processor or I/O levels could signal an intruder.
Password failures at login	Operational	Attempted break-in by password guessing.
Failures to login from specified terminals	Operational	Attempted break-in.
Command or Program Execution Activity		
Execution frequency	Mean and standard deviation	May detect intruders, who are likely to use different commands, or a successful penetration by a legitimate user, who has gained access to privileged commands.
Program resource utilization	Mean and standard deviation	An abnormal value might suggest injection of a virus or Trojan horse, which performs side-effects that increase I/O or processor utilization.
Execution denials	Operational model	May detect penetration attempt by individual user who seeks higher privileges.
File Access Activity		
Read, write, create, delete	Mean and standard	Abnormalities for read and write access for individual users may signify masquerading or

frequency	deviation	browsing.
Records read, written	Mean and standard deviation	Abnormality could signify an attempt to obtain sensitive data by inference and aggregation.
Failure count for read, write, create, delete	Operational	May detect users who persistently attempt to access unauthorized files.

Host vs Network

- HIDS (Host-Based Intrusion Detection System)
    - Based on host - installed inside the system
    - Surveillances (surveils ;) the system users and detects any hacking attempts
    - Single monitoring system
    - OS Dependent
    - NIDS (Network Intrusion Detection System)
    - Based on the packet capturing of the network
    - Analyze packets travelling through the network and detects any intrusions
    - Monitor according to every network unit



Snort

- NIDS
- Started its life as tcpdump
- packet capture agent that takes raw data straight off the wire
- enhanced to compare that data to a list of known attack patterns
- alerting you when traffic patterns match attack patterns
- Mainly signature-based, uses a combination of rules and preprocessors (anomaly)
- Benefits
    - Configurable
        - Inner workings, config files, rules are all laid bare to you
        - You can fine tune Snort to your specific network architecture
        - FREE
        - Widely used
        - Tens of thousands of downloads
        - De facto standard worldwide in the industry
        - Runs on multiple platforms
        - Runs on all Unix OS and Windows
        - Constantly updated
        - Maintenance releases come out regularly - every few months
        - Rules are regularly updated with new attack signatures
    - Preprocessors
        - IP defragmentation, port-scan detection, web traffic normalization, TCP stream reassembly
        - Can analyze streams not only a single packet at a time

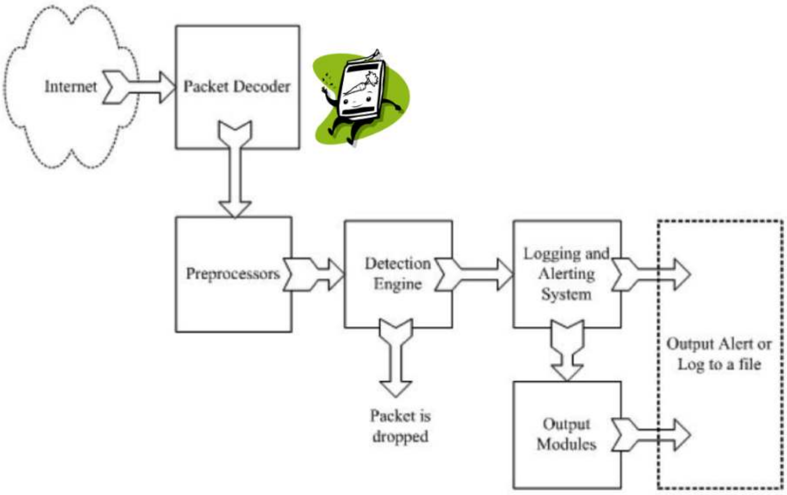


Figure 1-5 Components of Snort.

Packet Logger Mode

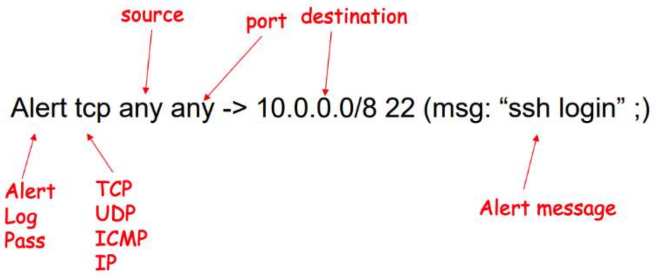
- Multi-mode packet logging options available
- Log all data and post-process to look for anomalous activity
- Saves packets to disk

Sniffing, Casey Netstat (network status)

- Shows all network connections and can show addresses as IP address being used

SNORT Rules

- Set of instructions defined to take action after matching some sort of signatures (atomic or composite)
- Categorized
    - Low-level protocols (icmp, netbios, tcp, udp)
    - High-level protocols (http, ftp, dns, pop3, imap)
    - Web server specific (web-attack, web-cgi, web-client)
    - Exploit specific (shellcode, backdoor, exploit)
    - Service impacting (dos, ddos)
    - Policy specific (policy, info, misc, porn)
    - Scanning and probing activities (scan, bad-traffic)
    - Viruses, worms, other malware (virus)
    - E.g.



```
Alert tcp any any -> 10.0.0.0/8 22 (msg: “ssh login” ;)
```

- Actions

Action	Description
alert	Generate an alert using selected alert method - log the packet
log	Log the packet
pass	Ignore packet

activate	Alert and then turn on another dynamic rule
dynamic	Remain idle until activated by an activate rule, then act as a log rule
drop	Make iptables drop the packet and log the packet
reject	Make iptables drop the packet, log it, send a TCP reset if protocol is TCP or an ICMP port unreachable message if the protocol is UDP
sdrop	Make iptables drop the packet but does not log it

Examples of Snort Rule Operations
- DO WE NEED IT?
- We may be asked what a Snort rule does and whether certain packets will trigger them, but we won't be asked to write them

Signature-based IDS Rules
- Source of 'intelligence' of product's competitiveness
- Can only detect attacks who's patterns are known, not any variations of the rules

Attack Patterns and False Alarms
- Packet sniffing and string pattern matching may be simple but real-world deployment is quite challenging
  - E.g. False positives and false negatives

Signature Database
- Extremely important that signature database (aka attack patterns) are kept current at all times
- Compromised or contaminated signature database may cause failure in detecting known attacks

Snort Output
- Each alert/notification of a potential intrusion (based on the rule) includes
  - Date and Time
  - SID (Snort ID) - identifier indicating which rule was tripped
  - Brief text message
  - Classification and priority of the attack
  - Protocol of the packet that tripped the rule
  - Source and destination IP addresses involved
  - Alert_fast
  - Show the flavor of an attack
  - Its classification
  - Destination
  - Timestamp
  - Super fast as low overheads to generate output
  - Alert_full
  - Full output of all details of the alert
  - Visualization Tools
  - ACID (Analysis Console for Intrusion Detection)
  - Powered by MS-SQL

Successful IDS Deployment Requirements
- IDS sensor must be installed at the right place in your network

- Real-time monitoring and “handling” essential
- IDS output must be protected
- Rules must be kept current and up to date at all times

Application Domains for IDS

- Host IDS
    - Linux, Windows, Web Servers, etc
    - Network IDS
    - Spam, DDoS Detection, Malware Detection
    - Wireless Ad Hoc Networks
    - Wireless Sensor Nets
    - Cyber Physical Systems
    - Smart Phones
    - Insider Threats

Highly-scalable IDS

- Most widely used software ids is unable to read network packets at the rate of more than a few Gbps - designed to utilize only a single CPU core for attack pattern matching
- A 12-core machine with two GPUs handles up to 33 Gbps of normal traffic and achieved 9 - 10 Gbps even when all packets contain attack signatures
- Two basic techniques for higher performance are
    - Batch processing
    - Parallel execution with an intelligent load balancing algorithm

Intrusion Detection Alert Correlation - not examinable

- Hard to make sense out of large pile of alerts coming from many different sources (e.g. Snort, System Call Traces (HIDS), File System Integrity Checkers)
- Many false alarms!
- Up to potentially 20,000 a day!
- General categories
    - Alert Clustering
        - Join alerts together in some meaningful groups
        - Matching Predefined Attack Scenarios (i.e.TCP SYN Flood)
        - Prerequisites/Consequences
        - Pre: necessary condition for the attack to be successful
        - Con: possible outcome of the attack

IDS vs IPS

- Many products provide both features
- Is prevention really possible?

IDS - Passive Detection	IPS - Inline Prevention
Connected to a tap of switch span port	Directly inline
Receives a copy of traffic	Traffic actually flows through system
Creates alerts	Creates alerts
Can’t block attacks	Can block attacks
Detection errors can result in false alarms	Detection errors can result in service

	disruption
Device malfunctions will cause a cessation of alarms	Device malfunctions can result in service disruption

- Why use an IPS instead of an IDS:
  - IDS only alerts attacks are happening
  - Network attacks can happen fast and some of them only need a small amount of time to complete the attack
  - So once a network technician gets into the network equipment the attacker could be long gone and got what they needed
  - You can now fix the problem but it's a little bit too late
  - IPS
  - Combines IDS and improved Firewall technologies
  - Make access control decisions based on application content, rather than IP address or ports as traditional firewalls had done
  - IDS is cheaper though!
  - Lots of Open Source IDS
  - IPS is EXPENSIVE

IPS

DONT NEED IT? - WOULDN'T PUT IT PAST OLE STEVE TO THROW US UNDER THE BUS

As if he hasnt been doing exactly that up to now

Firewall - The need for it!

- Internet connectivity is essential - creates a means of threats
  - Protects all computers in an office of limited size in a single location.
  - Network traffic screening, reporting and management features
  - Enterprise firewall
  - For large organizations, graphically dispersed
  - Consolidated reporting for multiple firewalls
  - Management tools enable you to configure multiple firewalls in a single step
  - All-in-one tools

Firewall Functionality

  - Positive: allows to pass only packets that meet a specific criteria
  - Negative: Reject any packet that meets certain criteria
- Firewall may examine
  - Protocol headers in packet
  - Packet payload
  - Pattern generated by a sequence of packets

Firewall Rules

- Enforces rules about what network traffic is allowed to enter or leave personal network/computer
- Often you may have to add your own rules - already has preconfigured rules
- E.g.
  - Allow everyone to access all Websites
  - Allow outgoing email from the internal mail server

    - Drop all outgoing network traffic unless it matches first two rules
    - Drop all incoming network traffic except for connections to the public web server
    - Log all connection attempts that were rejected by the firewall
    - Log all access to external web sites
- Effective means of protecting LANs
- Inserted between the premises network and the Internet to establish a controlled link
    - Can be one computer or multiplier working together
    - Used as a perimeter defense
    - Single choke point to impose security and auditing
    - Insulates internal systems from external networks

Firewall Characteristics

1. All traffic from inside to outside must pass through the firewall
    1. Physically block all access to LAN except via firewall

b. Only authorized traffic as defined by local security policy will be allowed to pass

c. Firewall itself is immune to penetration

What do Firewalls do?

- Block incoming network traffic based on source or destination
- Black outgoing network traffic based on source or destination
    - e.g. preventing employees from accessing inappropriate websites
- Block network traffic based on content
    - e.g. firewalls equipped with virus scanners can prevent files that contain viruses from entering your network.
    - Other firewalls will screen out unacceptable emails
- Make internal resources available
    - e.g. public web server
- Allow connections to internal network
    - e.g. salespeople can use VPN to connect to the corporate network from a remote destination
- Report on network traffic and firewall activities
    - Giving details of who tries to access material from inside the firewall and who may be trying to access inappropriate information from the internet going through the firewall

Types of firewalls

- Personal firewall
    - Installed on a single computer and protects only that computer or a small number of computers
    - Often have limited reporting and management features
- Departmental or small organization firewall

Basic Functions of a Firewall

- Packet filtering
    - All packet headers are inspected are thus determines if a packet is accepted or not
- Network Address Translation (I did NAhT)
    - Outside world only sees one or more outside IP addresses of the firewall
    - The internal network can use any address in the private IP address range.
    - Source and destination addresses are automatically changed ("translated") back and forth by the firewall.
- Application proxy

  - Requires understanding of specific application protocol
- Monitoring and logging
  - Allows analysis of possible security breaches and gives feedback on performance and actual filtering done

Advanced Functions of a Firewall

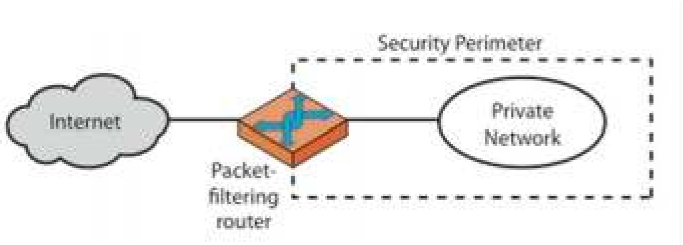
- Data caching
  - Caches data retrieved from web sites
  - Saves from repeatedly getting data from the same website
- Content filtering
  - Restrict data from certain inappropriate websites based on URLs, keywords, or content type (e.g. video streams)
- Intrusion detection
  - Detects certain patterns of network traffic and can choose to block those IP addresses if it deems them suspicious
- Load balancing
  - Need to have more than one point of entry from an availability standpoint
  - Allows incoming and outgoing requests to be distributed among two or more cooperating firewalls to help even the load.
- Static Address Mapping
  - Allows access attempts to the public firewall address to be redirected to the internal server
- Encryption
  - To prevent others from intercepting and reading information sent on the network
  - Also serve to prevent modifications of IP packets while they travel on the network

What Firewalls CANNOT Prevent

- Inside Attack
  - These users have already passed the firewall.
  - Firewall cannot do anything to stop internal network attacks from within.
  - Other security measures will need to be put in place to prevent this type of attack (i.e. restricted permissions and auditing of network access)
- Social Engineering
  - Attack in which hackers obtain information by calling employees and pretending to be someone else from the firm i.e. a trusted person with legitimate credentials
  - Hacker asks for sensitive information such as IP Addresses, passwords, etc.
  - Employees must know of these types of attacks and know never to distribute sensitive information
- Viruses and Trojan Horse Programs
  - Change constantly
  - The ability to distinguish between acceptable and malicious email attachments and content is a continuing problem for modern computer users
  - Good precautions should be taken to prevent spread of viruses and to minimize the damage that virus can do
  - Trojans horse programs are harder to spot as they may just open a backdoor and they don't spread like viruses do
- Poorly Trained Firewall Administrators
  - Firewall doesn't know what is or isn't acceptable unless admin tells it
  - Competent firewall admins should correctly specify which network traffic should be blocked

Packet Filter Firewall – 1st Gen

- Network level firewall, screening router firewall
- Operates on transport and network layers of the TCP/IP stack
- Applies a set of rules to **each IP packet** then forwards or drops (or rejects) the packet



- Setup as a list of rules based on matches to fields in the IP or TCP header
- If there is a match to one of the rules, that rule is invoked to determine whether to forward or discard packet
- Two default policies
  - Discard (everything is blocked)
    - More conservative, controlled, visible to users
  - Forward (everything is allowed)
    - If there is no match to any rule, then a default action is taken
    - Easier to manage and use but less secure

Packet Filtering Policy/Rules

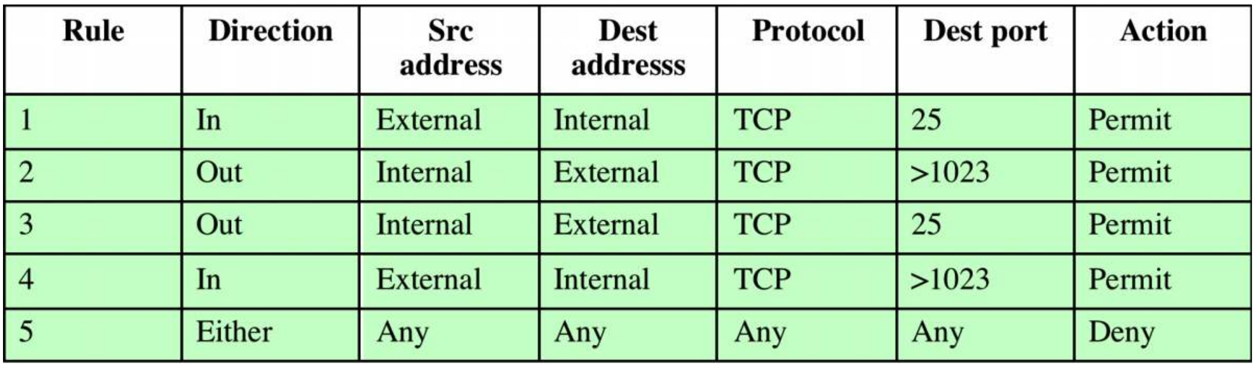
- Examples
- We use the notation in the textbook as "default"

Textbook Notation - IS THIS RIGHT?

Port/Content	Users	Time	Action
Port 80/except video	All	Always	Allow
Port 80/video	Trainers	Day	Allow
All ports, except 80	All	Always	Deny
Port 80/video	All, except trainers	Always	Deny
Port 80/video	Trainers	Night	Deny

Red - Deny-All Strategy
Green - Allow-all Strategy

Firewall for Dummies Notation - IS THIS RIGHT?



Rule	Direction	Src address	Dest addresss	Protocol	Dest port	Action
1	In	External	Internal	TCP	25	Permit
2	Out	Internal	External	TCP	>1023	Permit
3	Out	Internal	External	TCP	25	Permit
4	In	External	Internal	TCP	>1023	Permit
5	Either	Any	Any	Any	Any	Deny

DIY - Study some firewall rules - I definitely think he will ask us about this

Packet Filter Firewall: Advantages and Weaknesses

Advantages:

- Simple
- Typically transparent to users and very fast
- Network attack defence against DoS attacks

Weaknesses:

- Cannot prevent attacks that employ application specific vulnerabilities or functions
- Limited logging functionality
- Doesn't support advanced user authentication
- Vulnerable to attacks on TCP/IP protocol bugs
  - IP spoofing, source IP address - an internal IP address
- Improper configuration can lead to breaches

- WHY?
  A packet filter makes filtering decisions on an individual packet basis and doesn't take into consideration any higher-layer context

Stateful Packet Filtering

- Firewall remembers "state" about expected return packets
- Blocks all traffic on ports greater than 1023 - only allows network traffic that matches the response port of previously sent IP packet
- Firewall maintains a table of information on which ports it may expect traffic
- Firewall determines that a communication exchange has finished and removes information from the table
- Also automatically removes information after a short time period if unable to detect communication has ended
- Stateful Firewall Connection State Table
  - Tightens up rules for TCP traffic by creating a directory of outbound TCP connections
    - One entry for each currently established connection

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	210.9.88.29	80	Established
...	..	..	..	...

- Allows incoming traffic to only high-numbered ports only for those packets that fit the profile of one of the entries in this directory
- Some also keep track of TCP sequence numbers to prevent attacks that depend on the sequence number - e.g. session hijacking
- Some even inspect limited amounts of application data for some well-known protocols like FTP, SIPS commands, to identify and track related connections
- Using proxy servers
  - Typically proxy servers sit between client and actual service
  - Both client and server talk to proxy rather than directly with each other

Static Inbound Translation

- Instead of statically mapping all ports of a specific public IP address to an internal private IP address
- Specify that only specific ports from the public IP address should be mapped to the internal IP address
- Known as port forwarding or server publishing

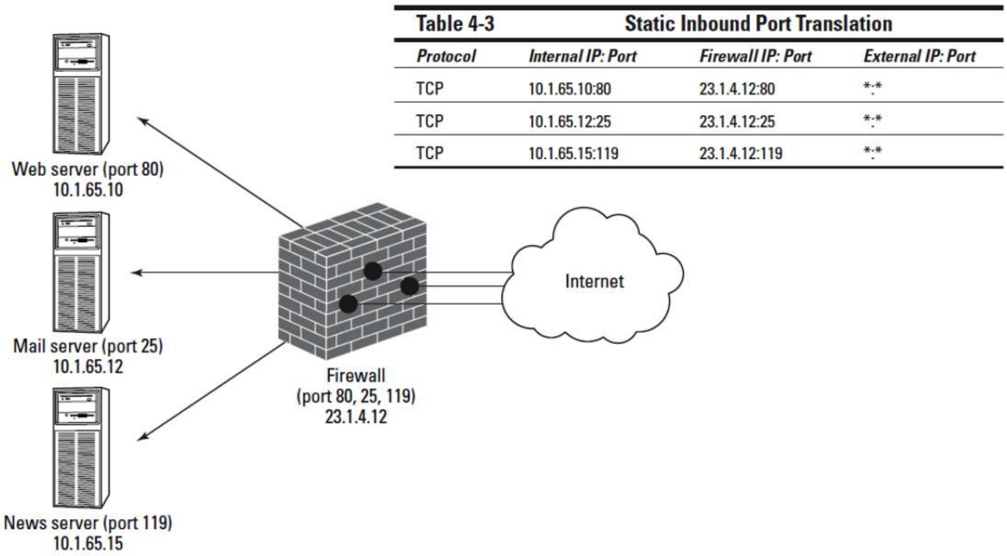


Table 4-3 Static Inbound Port Translation

Protocol	*Internal IP: Port*	*Firewall IP: Port*	*External IP: Port*
TCP	10.1.65.10:80	23.1.4.12:80	*:*
TCP	10.1.65.12:25	23.1.4.12:25	*:*
TCP	10.1.65.15:119	23.1.4.12:119	*:*

Content Filtering

- Based On HTTP content type, file name, file content/virus, keywords, SMTP email inspection, etc

- Also based on site name, site IP address, time of day, username, connection quota, data quota, etc
- E.g.

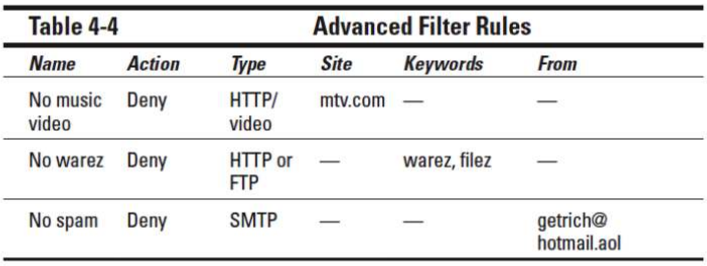


Table 4-4 Advanced Filter Rules

Name	*Action*	*Type*	*Site*	*Keywords*	*From*
No music video	Deny	HTTP/ video	mtv.com	—	—
No warez	Deny	HTTP or FTP	—	warez, filez	—
No spam	Deny	SMTP	—	—	getrich@ hotmail.aol

Firewall and Packet Encryption

- Renders your firewall unable to inspect data
    - Firewall cannot decipher encrypted content that’s sended to other particiaptns on the network when packets pass through the firewall
    - Firewall is supposed to make decisions based on information in the packets
- Firewall is unable to perform NAT
    - May break the integrity checksums used by the encryption protocol when the source or destination Ip address in the Ip header changes the TCP or UDP ports
    - Destination computer rejects packet because it discovers that the packet has changed after it left the source computer
    - Also, some network protocols include source or destination addresses in the application portion of the IP packet - if portion is encrypted, the firewall can't find the address and replace those during the NAT process
- Firewall can now provide a start or endpoint for VPN
    - Firewall is border between internal network and the untrusted external network
    - Convenient place to initiate a VPN connection or act as endpoint as VPN requires encryption

VPN and Firewalls

- Wrapped up in photo

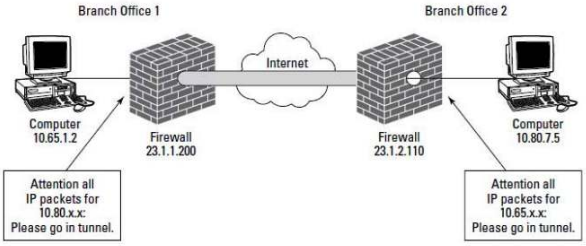


Figure 4-2: VPN tunnel between two branch offices.

- Other patterns are also possible. For example,

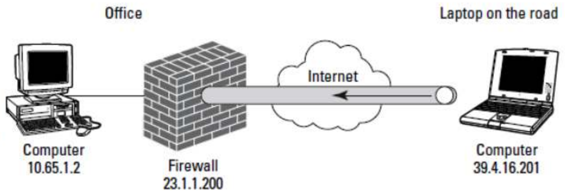


Figure 4-3: VPN tunnel to dial in to the office.

Security Policy Definition Is ESSENTIAL

- Cannot configure a firewall unless you know what network traffic is allowed and what is disallowed
- Create two policy documents
  - Internet Acceptable Use Policy
  - Security Policy

1

Internet Acceptable Use Policy

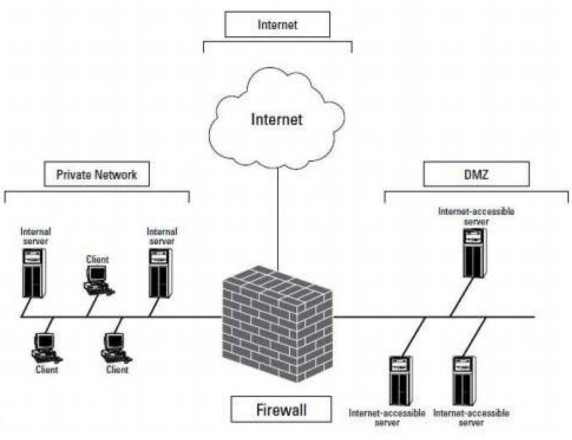
- Identifies what actions are considered acceptable when users access the internet - spells out the dos and don'ts!
- Defines:
  - All available services
  - Who can access the internet
  - Ownership of resources
  - Responsibilities of the employees
  - All unauthorized use of the internet
    - What purposes email are disallowed? (e.g. spam)
    - Web Content not able to be accessed
    - What types of files can’t be downloaded from internet
  - Actions that are taken if policy is not followed

Setting a Security Policy

- Defines the resources that a company deems important enough to secure
- Describes company's plan of action for security
- Define what needs to be protected and what actions must be taken in order to protect resources
- For employees, definites what is important to the company and what therefore must be secured from attackers
- Essentially a risk management activity
- "Generic" risks include unauthorized access, unauthorized disclosure, unavailability of the resource due to DoS attacks
- Example
    - Establish a project team to develop a security policy
    - Identify what resources require protection
    - Identify what potential risks exist for each resource
    - Decide the probability of each risk
    - Create mitigation plans that address each risk

Firewall and DMZ (Demilitarized Zone)

- All traffic that enters and exits is inspected
    - DMZ is probably the most secured segment of the network - all data that enters or exits the DMZ is inspected against a firewall rule listing to determine whether the traffic is approved to enter or exit the DMZ
- Resources in the DMZs are inspected to ensure that security is not compromised
    - Many companies use intrusion detection software in the DMZ, both on the network and at each device in the DMZ, to identify attacks launched against resources
    - Intrusion detection software immediately informs the firewall admin that a suspected attack is taking place
- DMZs act as a protective boundary to the private network
    - By placing Internet-accessible resources in the DMZ, firewall can be configured to prevent all access attempts to the private network from the internet
    - Only access attempts directed to the DMZ are permitted by the firewall, as long as the attempts use only approved protocols
- Network admins deploy two common configurations when deploying a DMZ to protect Internet Accessible Resources
    - Three-Pronged Firewalls
        - Use of three network cards in the firewall
        - Each network interface card is assigned to a zone of the network
            - Private Network Zone
            - Internet Zone
            - DMZ
    - Multiple Firewall DMZ



11-1: DMZ sing a three- onged ll.

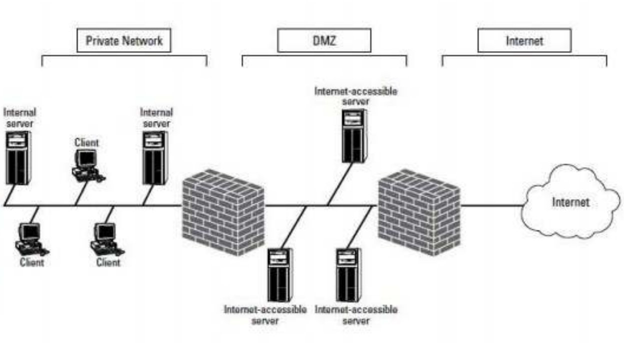


Figure 11-2: A DMZ using two separate firewalls.

Firewall and Private Network Addressing in the DMZ 555

- Scenario looks similar to this

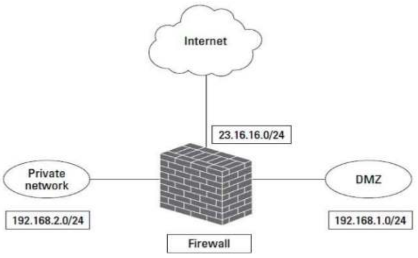


Figure 11-3: Implementing private network addressing in the DMZ.

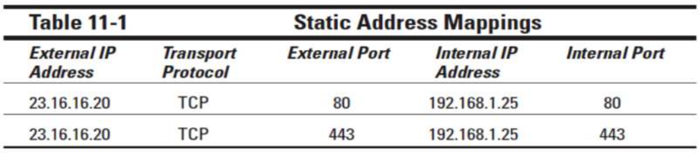


Table 11-1 Static Address Mappings

External IP Address	*Transport Protocol*	*External Port*	*Internal IP Address*	*Internal Port*
23.16.16.20	TCP	80	192.168.1.25	80
23.16.16.20	TCP	443	192.168.1.25	443

- The Static Address Mappings would appear as such:

External IP Address	Transport Protocol	External Port	Internal IP Address	Internal Port
23.16.16.20	TCP	80	192.168.1.25	80
..	..	..	..	..

Firewall Products

- There are many
    - Windows

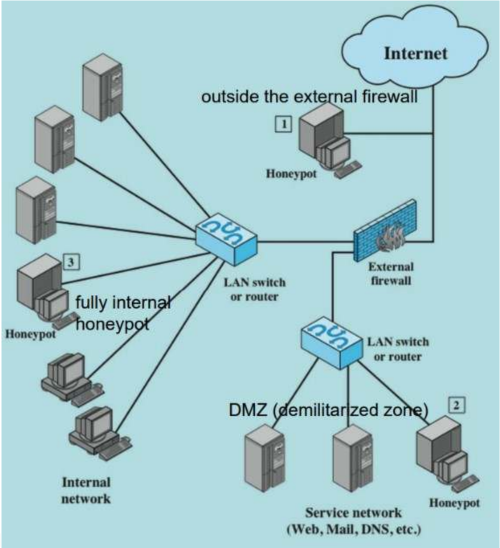
  - Linux
  - Personal Firewalls
  - Microsoft: Internet Security and Acceleration Server
  - Checkpoint Firewall-1

Firewall Capabilities and Limitations

- Capabilities
  - Defines a single choke point
  - Provides a location for monitoring security events
  - Convenient platform for several internet functions that are not security related
  - Can serve as the platform for IPSec
- Limitations
  - Cannot protect against attacks bypassing firewall
  - May not protect fully against internal threats
  - Improperly secured wireless LAN can be accessed from outside the organisation
  - Laptop, PDA, or portable storage device may be infected outside the corporate network then used internally

Anomaly Detection

- Intrusion Detection can be considered as a classification problem



  - Pattern matching
  - Feature Extraction (and selection)

Honeypot

- Decoy systems designed to
  - Lure a potential attacker away from critical systems
  - Collect information about the attacker’s activity
  - Encourage the attacker to stay on the system long enough for administrators to respond
- Filled with fabricated information that a legitimate user of the system wouldn’t access
- Once hackers are within the network, administrators can observe their behavior to figure out defenses

Conclusions

- IDS Itself (signature and anomaly based) should be improved further to counteract various challenges - evasion, encryption, new exploits, unknown exploits
- IDSs for new applications domain and new threats should be developed

- IDS should be operated tightly with other security solutions
  - IPS
  - SIEM
  - ESM