



Information Security for Technical Staff

Module 8:

Threats, Vulnerabilities, and Attacks

Networked Systems Survivability

**CERT[®] Coordination Center
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890**

© 2002 Carnegie Mellon University
® CERT, CERT Coordination Center and Carnegie Mellon are registered in the
U.S. Patent and Trademark Office



Instructional Objectives

- Define common terminology used to describe threats, vulnerabilities, and attacks
- Describe the range of threats to information and information systems
- Identify the classes of threat actors
- Discuss the range of vulnerabilities found in information systems
- Introduce the range of attacks used against systems
- Discuss an attack scenario used to breach information systems

This module focuses on the range of threats, threat actors, vulnerabilities, and attacks leveraged against information system. Common terminology is defined in order to present a consistent language when relating threats and vulnerabilities. This module further describes the concept of an intruder – someone who has the means, motive and opportunity to affect attacks against a victim. A scenario is presented toward the end of this unit to help construct a typical example of the attack sequence and decisions used by an attacker.



Module Overview

Overview

- Common terms
- Threats to information systems
- Threat actors
- Vulnerabilities
- Forms of attack
- Network intrusion scenario example

This module provides an introduction into threats and technical vulnerabilities through a discussion of threats to information security requirements, classifications of threat actors, known technological vulnerabilities, and known attack strategies. The primary focus in this module is network intrusions. Note, however, that more avenues of attack are often leveraged and will therefore also be discussed. Finally, a scenario example is introduced to examine a realistic attack approach. Taken from the intruder's point-of-view, this scenario will discuss the plausible course of action as seen through a timeline.

Common Terms

Vulnerability – feature allowing compromise

Threat – circumstance leading to potential loss of C. I. A.

Safeguard – means to reduce vulnerability

Incident – event which violates security policy

Attack – attempt to breach security

Intrusion – successful breach of security

Compromise – system placed into an insecure state



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 4

To understand fully the picture of the information security threat landscape, we must have a common language in which to describe events, actions, and actors. The following section introduces the basic definitions of threats, vulnerabilities, and attacks.

Vulnerability: A feature or a combination of features of a system that allows an adversary to place the system in a state that is contrary to the desires of the people responsible for the system and increases the probability or magnitude of undesirable behavior in or of the system

Threat: Any circumstance or event with the potential for causing undesirable destruction/loss, disclosure, modification, and/or interruption

Safeguard: An action, device, procedure, technique, or other measure that reduces the vulnerability of an information system

Incident: An event (or set of related events) in which the information security policies of an organization are violated

Incidents are sometimes found to be a collection of data representing one or more related attacks.

These attacks can be analyzed by categorizing the related attributes of an incident and by grouping the incident around: attacker (threat actor), type of attack, motivations and objectives, sites or locations involved, or timing and sequences.

Attack: An attempt to breach the security of an information asset or resource, or the method in which the attempt is made

Attacker: A person who deliberately attempts to breach the security of an information asset or resource

Intrusion: A breach in the security of an information asset or resource resulting from a successful attack

An action conducted by one adversary, the intruder, against another adversary, the victim. The intruder carries out an attack with a specific objective in mind. From the perspective of an administrator responsible for maintaining a system, an attack is a set of one or more events that may have one or more security consequences. From the perspective of an intruder, an attack is a mechanism to fulfill an objective.

Intruder: A person who deliberately breaches the security of an information asset or resource

More intuitively, the intruder is the actor or person who carries out an attack. Attacker, in most respects, is a common synonym for intruder with a few exceptions. The word intruder applies only after an attack has occurred. A potential intruder may be referred to as an adversary or an attacker. The majority of ambiguity between these two terms exists because the victim of the intrusion assigns the label of intruder or attacker. Therefore contingent on the victim's definition of encroachment, there can be no ubiquitous categorization of actions as being intrusive or not, intruder or attacker.

Compromise: Disclosure of or access to information by unauthorized persons (see also Intrusion) where the effects are a loss of confidentiality, integrity, and/or availability

Compromise of an information system resulting in access by an intruder at a level equivalent to that of a user or an administrator (a.k.a. root, superuser) of the system


Networked Systems Survivability

Threats Overview

... to Confidentiality

... to Integrity

... to Availability



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 5

The illustration shows a person sitting at a desk with a computer monitor and keyboard. A thought bubble above the person's head contains the word 'THREATS' in large, bold, orange letters. The background of the thought bubble is a stylized landscape with a sun and clouds. The person has a concerned expression on their face.

Threat exists to any and all forms of asset (physical, intellectual, and informational resources). With respect to information systems, threats provide the potential to destroy, disclose, modify, and/or denial availability of an information asset. Threats act both directly and indirectly toward an asset and the results are not mutually exclusive. Here are a few examples:

- Information under direct assault by a threat actor. In this case, the attacker may destroy, disclose, modify, or interrupt access to the specific, targeted asset.
 - For example, when an attacker simply destroys electronic files through file deletion, the threat compromises the availability as well as the integrity of the information.
- Information under indirect assault by a threat actor. In this case, the attacker may destroy, disclose, modify, or interrupt access to supporting infrastructure, operating system and application software, or key configuration and support files.
 - For example, an attacker may cause any number of events that create a denial-of-service toward file access, application use, or network use, thereby denying access to the resource or asset but not causing direct destruction, modification, or disclosure.
 - In other circumstances, an attacker may weaken the operating system by destroying or hiding critical system files (dynamic libraries, run-time function, configuration files), causing an inability to interact properly with the desired resource or asset.
- Information under indiscriminate but deliberate assault by a threat actor. In this case, the attacker uses some form of automation or third party to cause a destruction, disclosure, modification, or interruption of access to an information asset.
 - For example, a virus delivered via e-mail, and executed by an unintended user, acts in an automated manner. The threat actor exhibits limited to no control over the attack after the initial release of the virus.

In the next few pages, we will discuss other threats as they relate to specific properties of information (confidentiality, integrity, and availability). These threats comprise a relative sample of the range of threats and damages, but do not cover the subject in an exhaustive manner.

Networked Systems Survivability

Threats to Confidentiality

- Unauthorized access
 - Observation, eavesdropping
 - Copying, theft
- Inappropriate disclosure



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 6

Threats to confidentiality include any circumstances where an information asset is disclosed through inappropriate, inadvertent, or otherwise unauthorized means to an unauthorized party. An unauthorized party is any person who does not have the authority to view, read, or discuss the information, as defined by the organization; and this includes both external and internal people.

Disclosure events include:

- Disclosure of sensitive network infrastructure information (topologies, technologies, vendors, etc).
- Disclosure of intellectual property, contracting / acquisition information, and strategic plans.
- Disclosure of organizational data (human resource, financial, departmental hierarchy).

Inappropriate (yet deliberate) disclosure involves events where information that is restricted by policy, training, or technical mechanism is disclosed to an unauthorized party. Examples include:

- Discussing sensitive, internal or company-specific information in newsgroups or Web discussion forums.
- Disclosure of sensitive, internal or company-specific information at conferences, in articles, or in reports to external customers.

Inadvertent disclosure involves events where information is provided to an unauthorized party in an accidental manner. Examples include:

- Printing information to the wrong printer where unauthorized persons can read or view the information.
- Information disclosure through observation (unencrypted network communications on the public Internet, etc).
- Providing content or information in a manner that is publicly available but through seemingly private sources (hiding development web pages on the production web server).

Negligent disclosure involves events where insiders provide information that may be sensitive to the organization without knowing whether the information is being disclosed in an unauthorized manner. A good example of this is when company employees fill out trade periodical or journal subscriptions. In

Student Workbook – Module 8: Threats, Vulnerabilities, and Attacks

these events, insiders potentially disclose harmful information regarding the management and operational practices of an organization, to include:

- The number of system, security, or network administrators
- The number or types of systems owned by the organization
- The organization's budget toward information technology or security
- The types or revisions of operating systems used
- The types of special function systems used (network infrastructure, IDS, firewalls, anti-virus servers, etc)

Unauthorized disclosure involves events where insiders purposefully and intently disclosure proprietary, sensitive, or company-specific information to unauthorized external or internal parties. Most of these cases involve a compromised insider who is disgruntled, bribed, and/or threatened.

Finally, the security property of confidentiality is unique from integrity and availability for reasons of recovery. Integrity and availability are properties that are recoverable through practices and procedures (like redundancy, restoration from source or backup media, etc) but disclosure events can only be recovered from when all parties to whom the information was disclosed are prevented from further disclosure.

Networked Systems Survivability

Threats to Integrity

Unauthorized modification or destruction
Loss of means to authenticate or verify integrity



© 2002 Carnegie Mellon University Module 8: Threats, Vulnerabilities, and Attacks - slide 7

Threats to integrity include any circumstances where an information asset is destroyed, modified, or otherwise adversely affected through inappropriate, inadvertent, or unauthorized means by an authorized or unauthorized party.

The distinction being:

- An unauthorized party describes both external and internal persons who do not have the authority to modify the information in a manner inconsistent with its purpose, as defined by the organization
- An authorized party describes both external and internal persons who have the authority to modify the information, but considers whether the result of their actions has exceeded the authority granted or is strictly prohibited by the organization -- making this party a inside threat actor

For example, an internal employee may have the authority (as an authorized party) to create, modify, and delete records in a financial database, but that person becomes a confirmed threat actor when they deliberately abuse their position in order to destroy, modify, or otherwise corrupt the information.

Integrity events include corruption of:

- Sensitive network infrastructure information (topologies, technologies, vendors, etc)
- Intellectual property, contracting / acquisition information, and strategic plans
- Organizational data (human resource, financial, departmental hierarchy)

Threats to integrity extend beyond the actual information to the metadata and other safeguards use to protect or describe the information. Meaning that the processes or technologies used to verify the integrity of information can be modified to have similar adverse effects as if the information was modified directly.

Some examples are corruption of:

- Cryptographic hash values used to verify the integrity of files and directories (i.e. Tripwire information)
- PGP keys used to verify the integrity of files and email messages
- Access control lists (ACLs) in operating systems used to restrict and allow access by users, programs, and services

Networked Systems Survivability

Threats to Availability

Denial of service


Loss / theft / destruction

Threats to integrity

- Availability of reliable data

Loss of the means to access data

- Passwords, encryption keys, backup technology



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 8

Threats to availability include any circumstances where an information asset has been denied access to parties authorized to interact with the information by intentional or unintentional means of an unauthorized or authorized party. An unauthorized party is any person who does not have the authority to interrupt access to the information, as defined by the organization; this includes both external and internal people.

Availability events include denial of access to information by:

- Attacks against the network infrastructure providing the conduit to information assets (topologies, technologies, vendors, etc)
- Theft or destruction of information assets
- Loss of authentication (i.e. passwords), encryption keys, and technology that enable access

Denial of availability events can involve theft, denial-of-service, or threats to integrity of the information asset, both directly and indirectly. Physical or electronic theft simply denies the authorized party from access to the information asset or system. Denial-of-service provides a temporary unavailability to the information asset or system to authorized parties.

Threats to integrity also render the information unreliable and, therefore, untrustworthy or useless to the authorized party – resulting in unavailability of the asset or resource. Just as threats to system, application, or supporting software are threats to integrity, they are also threats to availability. For example, if the cryptographic tool suite a party uses to interact with encrypted email messages is deleted or modified (to act in a manner that is inconsistent with its purpose), then not only is the cryptography tool suite unavailable, but any information stored, transmitted, or processed by the software.

Networked Systems Survivability

Other Threats

- Electromagnetic Interference
- Physical damage due to weather
- Natural disasters
- Armed conflicts
- Loss of power, water, network or phone connectivity



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 9

Other threats to information include events that happen beyond the reasonable control of the organization. These threats include:

- Environmental threats (natural disaster, flooding, fire, etc)
- Electromagnetic interference (solar flares, large power sources, nuclear weapons, etc)
- Armed conflicts
- Loss of required sources of utility (water, power, telecommunications, etc)

Other threats to information include the inadvertent, deliberate, or negligent actions completed by unassociated or related third parties. These events are separate from the traditional threats because the party causing the threat is typically outside of the span of control and lacks the direct motivation.

Threat Actors

Any person or party who has the potential to do harm

May cause undesirable outcomes:

- Independent of their motivations:
 - Deliberately
 - Inadvertently
 - Negligently
- Independent of their technical abilities



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 10

Expanding upon the definition of threat – *Any circumstance or event with the potential for causing undesirable destruction/loss, disclosure, modification, and/or interruption* – is the actor who deliberately, inadvertently, or negligently causes the undesirable outcome. Threat actors are the persons and parties who leverage, create or affect threats against information and systems. In reality, the list of threat actors includes the malicious and the harmless as well as the legal and unlawful.

In the following list the traditional threat actors are listed as well as those parties who, through their involvement with an organization, additionally become capable of leveraging destructions/losses, disclosures, modifications, and interruptions to information assets.

Activists	Digital Forensic Specialists	Police
Competitors	Global coalitions	Professional thieves
Crackers	Government agencies	Reporters
Customers	Hackers	Spies
Cyber-gangs	Maintenance personnel	System and Network Administrators
Deranged people	Nature	Terrorists
Drug Cartels	Organized criminals	Tiger teams
Employees	Paramilitary groups	Vandals

The difference between many threat actors is motivation. Crackers, “people who maliciously break into information systems and intentionally cause harm in doing so [1]” are diametrically opposed to customers in their motivations to cause harm deliberately. This, however, does not exclude customers as a threat source if they inadvertently view, modify, or destroy information that the organization considers a sensitive or critical asset.

So, threat actors are not all seemingly malicious and unlawful. In these examples we see intent in another manner, to include a legal or legitimate intent. Police, customers, vendors, tiger teams, government agencies, and reporters are commonly overlooked as a threat actor. But these groups of people certainly

Student Workbook – Module 8: Threats, Vulnerabilities, and Attacks

act as threat actors when, in an official and legitimate capacity, cause the destruction, disclosure, modification, or denial of availability of an information or asset.

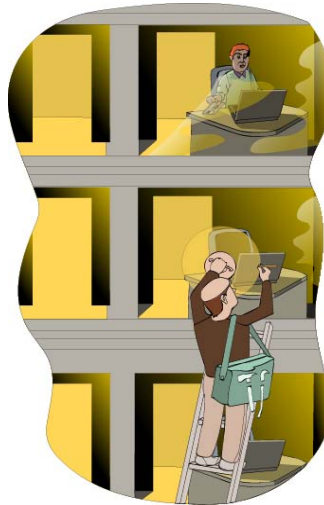
For example, any organization calling in the police to investigate a computer crime gives up (sometimes unknowingly) the ability to maintain confidentiality and availability. The reason being that through course of the investigation, typically unauthorized personnel from outside of the organization (in this case, police officers), will be able to view sensitive, proprietary, and company-specific information. The investigators may even require or have the legal right to demand an off-site forensic analysis of the compromised data or systems – rendering the ability of the organization to maintain availability or confidentiality implausible.



Intruders Overview

Internal

External



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 11

An intruder is a person who deliberately attempts to breach the security of an information asset or resource, through direct or indirect means. Intruders include all threat actors who act deliberately to thwart, bypass, or breach the policies and procedures of an organization. Direct manners include attempts by an intruder to access login information, crack passwords, create denial-of-service attacks, or leverage any direct technological mechanism at a target. Indirect manners include attempts by technologies acting on behalf of the intruder but which work independent of direct controls. For example, many malwares (Trojan horse, worms, and virus programs) are created by an individual but can and do exist in an autonomous fashion, exploiting vulnerabilities and leveraging system compromises without the direct control of the intruder.

Intruders have a means in which to employ an attack, malicious or purposeful motivation in which to carry out an attack, and seek opportunities in the form of specific and indiscriminate targets. An important facet of intruders is to realize that intruders who have discrete motivations toward breaching the policies of a specific organization are not necessarily more dangerous than those intruders who attack organization's assets (networks, systems, and data) as targets of opportunities.

For more information on intruder means, motives, opportunities, and sophistication, please refer to the student workbook section "The Challenge of Survivability," pages 21 through 28.



Internal Intruders

- Employees
- Contractors
- Service personnel
- Visitors
- Covert agents



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 12

Internal intruders include employees, contractors, service personnel, visitors, and covert agents (aka co-opted insiders). This group of threat actors is particularly dangerous because of their potential for causing a direct, large-magnitude impact on the organization.

Insider intruders, like employees and contractors, typically know which assets are important and where they are located, how the assets are protected, how the assets are at risk (to disclosure, modification, destruction, and interruption), and how to circumvent the protection mechanisms in place to protect the assets. Other insiders, like visitors and service personnel, are granted close quarters with the assets of the organization giving them the ability to leverage physical and electronic attacks.



External Intruders

- Former employees
- Contractors
- Clients and customers
- “Crackers”
- Vandals
- Thieves and organized crime
- Competitors
- Political opponents and insurgent groups
- Foreign agents

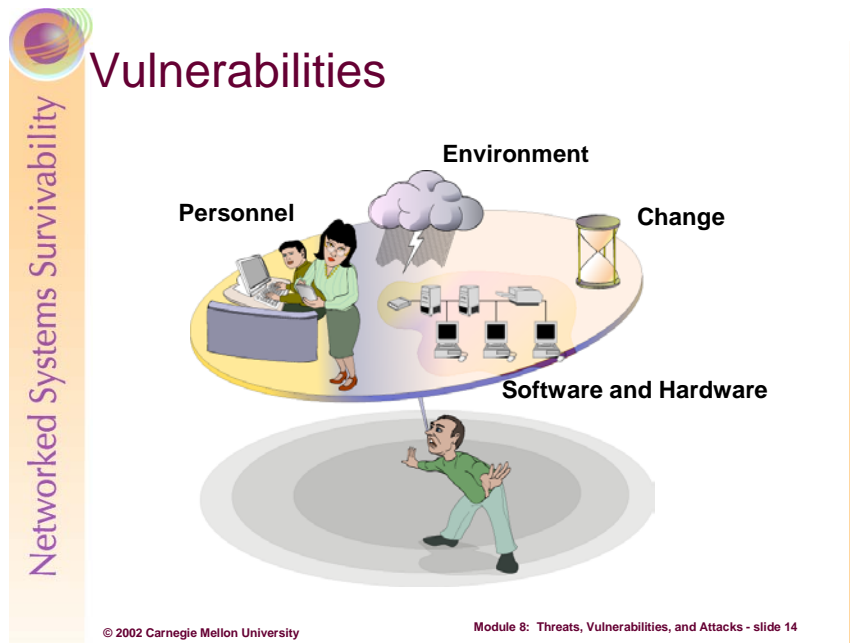


© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 13

External intruders pose the most obvious form of attack group, or at least the ones the most organizations prepare against. External intruders include former employees, contractors, clients, customers, and competitors. This group also includes the traditionally malicious actors, such as vandals, crackers, thieves, organized criminals, political opponents, insurgent groups, and foreign agents.

Typically, this group of intruder will employ technical mechanisms from the outside of an organization, but they are also the group who co-opt insiders using bribery, blackmail, or extortion. External intruders gain access to information systems, networks, and data of the target organization or party through electronic means (wired, dial-in and wireless network penetrations, etc).

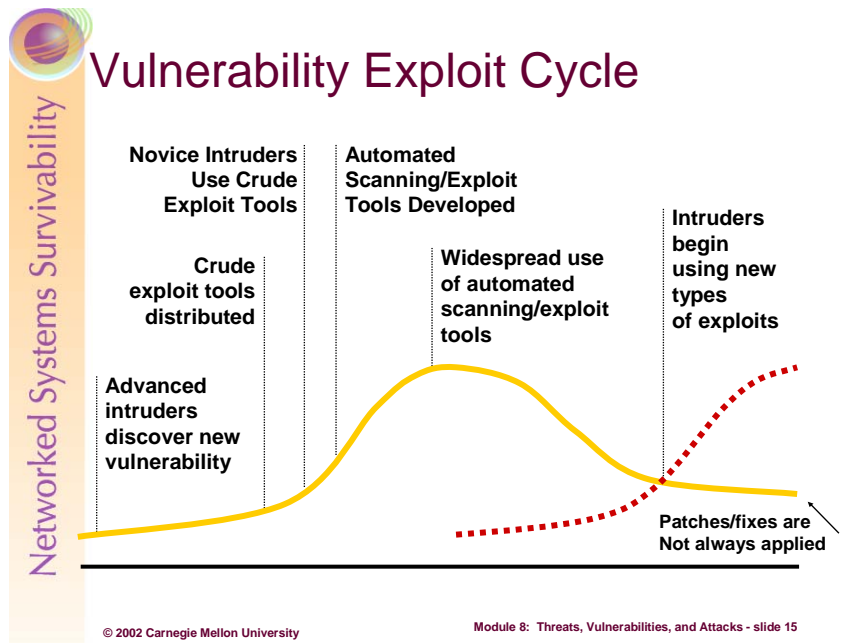


To refresh, Vulnerabilities in the technical sense are weaknesses in technologies that would allow an intruder to place a system or application into a state that is not desired by an organization. Vulnerabilities, however, do exist outside of technology in how personnel perform their duties, the natural and artificial environments, and through change.

Personnel, for example, may not be aware of or follow the correct procedures for completing a task or safeguarding information; hence creating a vulnerability that may be exploitable by an intruder. Not knowing that as a system user you should not give away your password or let another user use your account are some examples.

The environment, both naturally and artificially created, present conditions where systems and networks are at risk. Loss of control over humidity, power, cooling systems as well as tornadoes and other natural disaster presents ample opportunities for environmental vulnerability. Finally, what was true and correct today may not be the best course of action, procedure, management practice, or technological mechanism tomorrow. Change can introduce much vulnerability, both technical and non-technical.

The next several pages discuss technological and system management vulnerabilities.



A relationship described by the CERT/CC is the “Vulnerability Exploit Cycle.” This cycle depicts how most vulnerability magnitude relates to a function of (unknown) time. Above, and starting from the left, is a typical life cycle of any known and exploited vulnerability.

The timeline of a vulnerability starts when advanced intruders discover a new vulnerability through software testing and code examination. At this point, knowledge of the vulnerability is restricted to the exploit writer/tester.

After some unknown time, the exploit may be distributed in the form of a script or as a collection of one or more command-line inputs. Again, the distribution is mostly limited to the exploit finder, but this person may start to disseminate the information to a collection of known repositories, publications, Web sites, or peer attackers.

At some time later, the exploit, in script or command-line form, is picked up and learned by novice intruders (some who may not fully understand the exploits, its properties, or its use). The volume of exploit activity related to the vulnerability starts to increase and may be learned by those defending against attacks, such as computer security incident response teams (CSIRTs), system and security administrators, or software vendors.

Again, after some unknown time, the scripts or commands may be developed and/or coded into an automated scanning and exploit tool. These tools are sometimes graphically based and can be distributed via anonymous FTP (File Transfer Protocol) servers, Web sites, bulletin board systems, and/or physical media, between threat actors. Contrary to their original purpose, these tools are also adopted and learned by legitimate system and security professionals in an attempt to determine the exploitability of their systems and networks. Generally, at this point in the cycle, the vulnerability is well known to the attack and defense communities, and specific remedies and work-around may be available to mitigate against the vulnerability.

Generally at some time in the cycle time, widespread use of the automated attack tools (which exploit the vulnerability) occurs and the amount of exploit use, attempt, and success is at its peak. Most system and security administrators have already patched or updated their systems and networks through updates or work-arounds supplied by vendors at this point; however, the volume of attempt as well as volume of compromise on poorly configured or maintained systems is also at its peak. Here the vulnerability exploit code may be fully contained within a library of vulnerabilities within an attacker’s suite of tools.

Student Workbook – Module 8: Threats, Vulnerabilities, and Attacks

Over time, the volume of unreported and reported attacks decreases in volume as more systems are patched and the defense community learns more. This decrease, though, tends to never be come negligible or reach a “zero-incident” state for many reasons -- including poorly maintained systems that are never patched, systems redeployed with default configurations or reloads of systems with un-patched software/firmware, etc.

Finally, the cycle repeats or begins anew with the discovery of a new vulnerability. Many times these discoveries happen simultaneously, or over collapsed or expanded amounts of time.



Vulnerable Configurations

Default accounts

Passwords

Services

Remote access

Logging and auditing

Access controls



Demo – Aironet 340

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 16

The nature of computers today makes and demands that they be somewhat generic in scope and purpose so that the individual organization can manipulate these devices to fulfill a more specific operation or objective. This demand has prompted operating system and hardware systems manufacturers to make highly configurable products, but at the same time create vulnerable default configurations. Some of the most prominent weak and default conditions found in operating systems for example are:

- Guest and other default accounts left active or available
- Empty or well-known vendor passwords on accounts
- Unnecessary features, services, network ports, and applications enabled
- Remote access with little or no inherent security enabled
- Logging and auditing features disabled
- Incorrect default access controls on files, directories, and other objects

Continuing this discussion, let's examine the opportunities presented by several of these known and default configuration conditions. When incorrect system access controls are employed, access to administrative systems, applications, programs, and configuration data is available to those who would normally be prohibited from access them (to include employees who could be intruders).

This means that intruders could access files, directories, and storage volumes that may contain sensitive or private assets that are critical to the mission of the organization or the information technology staff. This also allows access to backup information, remote services, file and object ownership, and administrative services (account creation and deletion, etc) can be made available to intruders and others.

When these problems are coupled with incorrect network access controls, organizations and system administrators lose control over access to administrative capabilities of networked systems and components, router and switch configurations, firewall and intrusion detection configurations, network monitor configurations, and relationships between networked systems.

Problems Maintaining System and Network Software

Software maintenance

Software versioning issues

- Different versions across enterprise

Default settings

- Example: Windows shares



Demo – Windows Shares

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 17

Vulnerabilities are not only presented by inadequate or weak default conditions but also in the degradable state of applications, operating systems, and network protocols. Software presents and solves numerous vulnerabilities through their continuing need for patches and updates. Software updates like device drivers, new system tools, and improved applications are developed by vendors to enhance functionality but also to repair improper or poorly coded existing functions and routines. This requires system and network administrators to continually search and apply updated device drivers and software patches to fix functional and security deficiencies.

Potentially exploitable and vulnerable conditions are created by:

- Failing to keep software up-to-date regarding security fixes
- Assuming old configuration files will work for updated versions of software
- Assuming that new versions of software will have all the security fixes included
- Accepting unwritten default settings

This problem is easily compounded and exacerbated with each increase in scale of deployed systems in an organization. The potential for inconsistent of software versions and configurations across all systems and network infrastructure components becomes more likely.



Attacks Strategies

- Below threshold attacks
- Coordinated attacks
- Distributed coordinate attacks



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 18

Intruders, whether using known technical exploits and vulnerabilities or non-technical means, employ attacks through a number of strategies. The most commonly used tactics and strategies are described below.

Below-Threshold Attacks

Below-threshold attacks deal with strategies aimed at preventing the victim from being alerted to the attack even when security safeguards are in operation. The attacker develops a profile of the target victim by determining the tolerance and sensitivity of the detection mechanisms to levels of malicious activity.

For example, an attacker may watch for increased network traffic of a certain control protocol (like SNMP) during the course of scanning activities and then reduce the amount of scanning or lengthen the interval of scanning until no alerting traffic is detected. In this manner, the attacker learns the alerting threshold and sensitivity of intrusion detection, network monitoring, and firewall devices. Other examples include:

- Scanning multiple systems for evidence of one open port type (e.g. TCP port 80) versus in depth scanning of one target
- Attempting to compromise a target organization by trying a known vendor password one specific type of network device versus brute force password cracking on one system

Coordinated Attacks

In coordinated attacks, an attacker uses one or more types of attack to leverage a concerted attack on a target site. For example, an attacker may develop a specific Trojan Horse program to simulate the logon screen of a specific organization, send the program to a system administrator within the organization, and wait for the program to be executed. Here the intruder has used many attacks, including: information gathering, service enumeration, human engineering, Trojan Horse, etc.

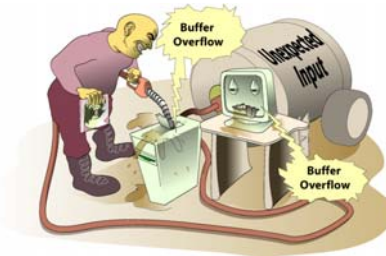
Distributed Coordinated Attacks

Distributed coordinated attacks involve one or more attackers who use one or more sets of vulnerable systems to attack one or more victims. For example, attacks involving:

- The redirection of Web browser traffic to attack a victim site
- A set of simultaneous attacks by a coordinated group of attackers to try to overwhelm defenses
- Misinformation provide to thousands of intermediaries in order to have them attempt to gain access to a victim site (i.e. DNS cache poisoning)

Commonly Exploited Attacks

- Buffer overflows
- Error insertion and analysis
- Insertion in transit
- Modification in transit
- Privileged program misuse
- Race conditions
- Replay attacks
- Residual data gathering
- Spoofing and masquerading



Demo – Buffer Overflows

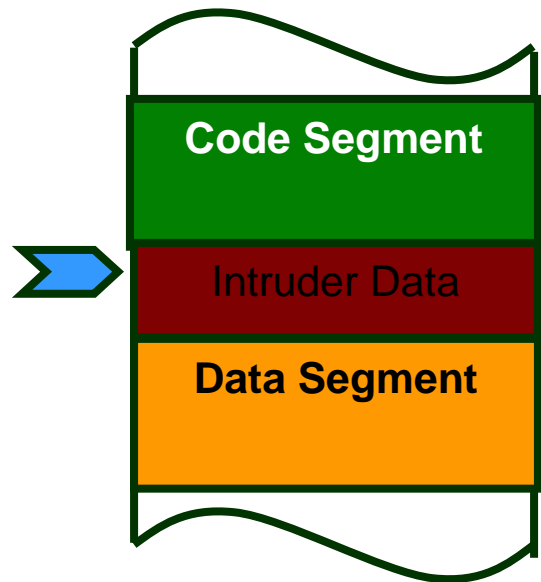
© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 19

A large number of attacks leverage vulnerabilities that have emerged as pre-existing conditions. This section will introduce the reader to the most common and widespread type of these conditions but not present an exhaustive look into all possible cases.

Buffer Overflows (aka -- input overflow)

“A buffer overflow occurs when a program or process tries to store more data in a buffer (temporary data storage area) than it was intended to hold. Since buffers are created to contain a finite amount of data, the extra information - which has to go somewhere - can overflow into adjacent buffers, corrupting or overwriting the valid data held in them. Although it may occur accidentally through programming error, buffer overflow is an increasingly common type of security attack on data integrity. In buffer overflow attacks, the extra data may contain codes designed to trigger specific actions, in effect sending new instructions to the attacked computer that could, for example, damage the user's files, change data, or disclose confidential information. Buffer overflow attacks are said to have arisen because the C programming language supplied the framework, and poor programming practices supplied the vulnerability. [3]”



As of January, 2002, CERT Coordination Center (CERT/CC) statistics showed that one half of the ten most seriously vulnerabilities reported (see <http://www.kb.cert.org/vuls/> listed by “metric severity”) included programs with buffer overflows [5].

Error Insertion and Analysis

Error insertion and analysis is when attempts are made to exploit the problems an application or operating system has supporting the execution, transmission, and storage of data objects. At the simplest level, errors are introduced purposefully to cause the safeguards protecting information to act inappropriately. For example, methods are commonly available for inducing errors in cryptographic processing in order to

reveal cryptographic keys stored in smart cards and devices. As another example, malformed network packets can be targeted at a network appliance to introduce errors in processing network traffic – causing an interruption of communications between two systems [1].

Insertion and modification in Transit

Insertion in transit attacks include TCP sequence attacks as discussed in Module 5 – TCP/IP Security. With respect to this specific attack, the intruder inserts network packets into a network, which have been crafted to insert data into a current TCP connection. The attacker’s forged packets use the same sequence numbers of the current connection in an attempt to trick the receiver’s TCP/IP stack into disregarding the other packets with the same sequence number that arrive later. In UDP connections, the intruder may attempt to intercept the datagram and modify the existing IP addressing information with other information in order for to misdirect the information.

Privileged Program Misuse

Privileged program misuse involves exploiting the condition where a typical user has access to functionality normally reserved for the system administrator. Here, the user exceeds their authority to create, modify, or destroy information because the operating environment has not expressly denied them the ability.

For example, a user allowed to create a rescue disk because of a misconfiguration of this privilege might be able to extract the password database – which could be used by the user to crack system password. Another example is when users are allowed to use backup programs and devices to create and restore data to and from archives. If the user deliberately restores the wrong information, they may be able to overwrite existing access permission and disable safeguards like system logging [1].

Race Conditions

Race condition attacks are when an “interdependent sequences of events” are interrupted or exploited by another “sequences of events that destroy critical dependencies.” Examples include exploiting the steps used by one process in order to introduce conditions not expected by another, subsequent, process. For example, if we are given that:

- 1) Process A checks for the existence of a file
- 2) Process B creates a file if none existed (as returned by A)

Then, an attacker could possibly insert commands between the between when the file existence is checked and the file is created. If the attackers command is to create a link between the file created and an existing file, such as the password database, then when B is executed the existing file will be overwritten. The result is that the password database is now zeroed (an empty file), thereby denying access to the operating environment or shared resources.

Replay Attacks

Replay attacks exist when information communicated via a network is captured and then later replayed to cause malfunction or errors. For example, if an authentication process is captured and then replayed by the attacker while the legitimate user is logged on, it may introduce errors into the existing session. Other examples include, replay of:

- Encrypted messages in order to cause the receiver to question the validity or confidentiality of the original message
- Alerting messages from IDS or firewalls in order to distract the network security administrator

Residual Data Gathering

Residual data gathering involves the reconstruction of data from processing or devices that were ineffective at completely erasing the information. For example, using supplemental file utilities or operating system commands to undelete information objects in operating systems. Residual data on physical media (CD-ROMs and diskettes) may be examined through electromagnetic analysis in order to reconstruct information even when it has been written over multiple times.

Spoofing and Masquerading

Spoofing and masquerading involve creation of “false or misleading information in order to fool a person or system into granting access” or act on the information presented. Examples include:

- IP source address spoofing in order to cause a person or system to believe the traffic has originated from one or more (false) locations
- Replacement of authentication methods (such as a login screen) in order to cause users to provide account and password information
- Electronic mail forgery generated to deceive the receiver in to acting in a certain manner
- Addition of log events, or the modification of log event times, in order to create false representations of attacker sequences, attempts, or methods

Specific Attacks

- Social engineering attacks
- Physical attacks
- Network intrusions
- Information gathering attacks
- Observation attacks
- Malicious code attacks
- Denial of service attacks
- Other attacks



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 20

Intruders often use many types of specific attacks to compromise systems, cause denial-of-service, or intercept communications. A subset of specific attacks are defined and described in the following paragraphs. This information is not meant to be exhaustive but a representative sample of the arsenal many intruders used to exact system compromises and other attacks. Additionally, because of the frequency and number of incidents involving certain attack types, several of these will be described in further detail in the next few sections.

Social Engineering Attacks: Social engineering is a form of spoofing and masquerading attack. The attacker exploits the weaknesses in a user's comprehension of security policy, awareness, or training, usually by pretending to be an insider or someone with legitimate access to information or resources. For example:

- Usurping a person into disclosing or changing their authentication information (usernames and passwords)
- Exploiting physical security by befriending an insider or physical security guard

Physical Attacks: Physical attacks are direct physical acts leveraged against systems, network media, storage locations, buildings, etc. For example:

- Causing a water line break on the first floor of a building in order to flood a basement-level server room
- Theft of storage devices or backup tapes from an off-site facility

Network Intrusions: Network intrusion attacks include any number of attacks that originate or use data and voice networks as a conduit for attack.

Information Gathering Attacks: Information gathering attacks involve the non-technical and technical techniques intruders use to acquire information on which to base future attacks.

Observation Attacks: Observation attacks use technical means to observe, record, and copy electronic communications between host systems on networks.

Malicious Code Attacks: Malicious code (or malware) includes viruses, Trojan horses, and worms.

Denial-of-Service Attacks: Denial-of-service attacks exploit the scarce resources common to most

systems and networks – that being processing power, bandwidth, physical data storage, number of service requests handled simultaneously, etc.

Other Attacks: Other attacks include advanced techniques or synergistic approaches used by attackers. Describing these forms of attack in detail would expose the individual elements (each attack method) used, which are similar to those already discussed. It is, however, beneficial to understanding they're relationship to individual attacks and how or why they are employed.

Combinations and Sequences Attacks [1]

Combined and sequence attacks use several attack methods concertedly in order to affect their goal of destruction, modification, disclosure, or interruption of information or systems. Examples include:

- Exploiting an emergency response to a fire, flood, or other physical threats in order to gain entry into secure locations where physical access is controlled
- Using malicious code to cause access control failures that allow subsequent exploitation by a controllable Trojan horse program designed from remote access
- Creating of false identities of organizational personnel in positions where they would be automatically granted access to critical information or systems
- Developing enticing Web sites designed to exploit users who visit them by sending their browsers content-based attacks or configure them to leak information through firewalls

Content-Based Attacks [1]

Content-based attacks involve sending information to any mechanism that performs an interpretation of the received information causing that mechanism to act undesirably. Examples include:

- Web-based URLs that bypass firewalls by causing the browser within the firewall to launch attacks against other inside systems
- Macros written in spreadsheet or word processing languages that cause those programs to perform malicious acts
- Compressed archives that contain files with name clashes causing key system files to be overwritten when the archive is decompressed

Cryptanalysis Attacks

Cryptanalysis attacks, which were discussed in Module 6 – Cryptography, are techniques used to analyze and break mathematical codes used to obfuscate information. Examples include:

- Frequency analysis for breaking mono-alphabetic substitution ciphers
- Index of coincidence analysis for breaking polyalphabetic substitution ciphers
- Historical breaking of the Enigma cipher in World War II through mathematical and optical techniques combined with knowledge of keys and key usage
- Exhaustive brute-force attacks on the Data Encryption Standard (DES) encryption standard
- Improved factoring for breaking cryptosystems based on modular arithmetic

Social Engineering Attacks

Bribes and extortion
Collaborative misuse
Fictitious people
False updates
Getting a job
Repair-replace-remove information
Shoulder surfing



Audio – Social Engineering

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 21

Social engineering, or human engineering, is the exploitation of social contexts to gain access to information, resources, systems, and network – to include critical assets of an organization. This type of manipulation by intruders is done through direct contact as well as communications via phone, e-mail, and paper mail. The following material describes some well-known techniques used to affect human engineering attacks.

Bribes and Extortion: Bribes and extortions are usually in the form of promises or threats that cause trusted parties to violate their trust established with the organization [1]. Examples include:

- Bribing a guard to gain entry into a building
- Kidnapping a key employee's family members to gain access to a computer system
- Using sexually explicit photographs to convince a trusted employee to provide insider information.

Collaborative Misuse: Collaborative misuse is when several persons or accounts (tied to individuals) are used in order to abuse or gain access to systems or information. Examples include:

- Creating a false account by one person and entering a computer database using the false identity by a second person
- Procuring attack software from an outsider for use in information theft, destruction, modification, or interruption
- Collaboration in covering the audit trail of an attack by a system administrator working with an outsider

Fictitious People: Fictitious people attacks involve impersonating or using a false identity to bypass physical access controls, manage perception, or create conditions conducive to attack.

False Updates: Providing seemingly legitimate software updates to unknowing system or network administrators in order to cause them to install the software on behalf of the attacker [1].

Getting a Job: The form of attack requires that the attacker apply for and secure employment within the target organization, as a direct employee or potential third-part collaborator [1]. Examples include:

- Getting a maintenance job to gain physical access to the assets

Student Workbook – Module 8: Threats, Vulnerabilities, and Attacks

- Serving as a spy for in intelligence agencies of competitors

Repair-Replace-Remove Information: Acting in a capacity as to create the need for repair services or rendering the maintenance service for a target organization in order to gain access to the information assets [1]. Examples include:

- Computer repair shops or service organizations copying information for resale to competitors or direct profit
- Maintenance people introducing computer viruses


Shoulder Surfing: Shoulder surfing is the direct physical attack where attacker's view information, such as authentication credentials, from the person interacting with a computer system. Examples include:

- Observing account name and password entry
- Observing laptop screens while in public areas such as airports or conferences
- Observing users in their job performance to understand standard operating procedures

Networked Systems Survivability

Physical Attacks

- Cable cuts
- Dumpster diving
- Environmental control loss
- Fire
- Flood
- Power failure
- Wire closet attacks



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 22

Physical attacks are acts leveraged against systems, network media, storage locations, buildings, etc, through direct physical means. These types of attacks are described in further detail in the following material.

Cable cuts: Cable cuts are attacks that cause any physical separation of networking or communications equipment in order to disrupt normal operations.

Dumpster diving: Dumpster diving involves examining refuse and recycled materials in an attempt to breach the confidentiality of information assets.

Environmental control loss: Environmental control loss attacks involve causing failures to operating environments such as heating, cooling, and humidity through deliberate tampering or sabotage.

Fire and flood: Fire and flood attacks involve direct arson or physical release of liquids in order to cause physical damage, destruction, or otherwise disrupt normal operations of systems and networks.

Power failure: Power failure is an attack caused by physical attacks to a power system or the conduit for transmission of power to computers, networks, or building facilities -- in order disrupt normal operations.

Wire closet attacks: Wiring closet attacks involve breaking into communications and cabling closets in order to observe (by adding technology to a network), disrupt, damage, or destroy network and communications equipment.



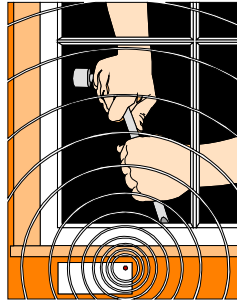
Network Intrusions

Achieved in a matter of seconds

By intruders interested in access and private data

Through a series of compromised systems

Can create jurisdiction problems



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 23

This module has introduced the range of threats, vulnerabilities, and attacks that intruders embody and deploy to compromise systems and networks. With the sophistication of attack methods, tools, and resources, intruders can affect network intrusions with minimized direct actions and requisite time, sometimes in a matter of seconds.

Intruders have many means, motives, and opportunities to exploit, corrupt, deny availability, disclose, and compromise the many assets of an organization or individual. These attacks often involve multiple geographically disparate sites and multiple sites in order to obfuscate the criminal trail of the ultimate attacker or intruder. Sometimes these attacks may even originate from outside, or traverse, your country's, state's, or region's jurisdictional boundaries, potentially inhibiting the prosecution or identification of the intruders and attackers.

The next several sections examine a possible scenario in order to reiterate the major points in this module. However, before we can present a plausible, realistic scenario we should revisit some of the pre-work techniques intruders use to perform target selection and identification. More information on this topic can be found in Module 7 entitled "Prelude to a Hack."



Information Gathering Attacks

Dumpster diving
Social engineering
Probes
Network scans
Network mapping
Keystroke monitoring
Packet sniffing



Probes and network scans are the most commonly reported intruder activity

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 24

Before we can discuss a scenario example, we must be familiar with some of the non-technical and technical techniques used by intruders to perform target selection and intelligence gathering. Intruders commonly use the following tactics to gather information about a potential site, system, or network to attack:

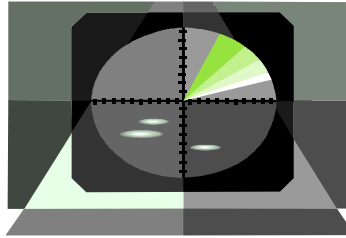
- Direct physical searches of refuse and waste, called “dumpster diving” to learn names of individuals, assets, intellectual properties, system and network types, etc
- Social engineering techniques to learn secrets from the organization’s employees, visitors, contractors, etc
- Network probes and scans to identify open ports, available connections, and the number and types of services used (as well as the potential for wireless eavesdropping)
- Network mapping using scanning techniques to allow intruders to find similar, trusted, and critical systems
- Keystroke monitoring and packet sniffing to learn usernames and passwords as well as network hosts, hostnames, firewall rules, intrusion detection system identification, etc

By far, the most prevalent type of information gathering reported to the CERT/CC at Carnegie Mellon University’s Software Engineering Institute are network-based probes and scanning activities by intruders.



Scans

- Automated tools use
- Scanning detection
- Stealth scanning
- “War dialing”
- “War driving”



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 25

As a major form of target reconnaissance, intruders use automated and manual scanning tools and techniques to locate networks, determine device configurations, enumerate open ports and services, and determine operating systems deployed, at the target location(s).

In the early days of network scanning, patterns of scanning activity were readily identifiable in network logs – denoted as a series of progressive and consecutive probes to a range of system addresses or port numbers. As more advanced tool suites became available or were developed by intruders, stealth techniques developed to spread probes out over time and network geography in order to appear as inconspicuous traffic – usually within the normal network traffic patterns.

Additionally, in the earlier days of scanning, automated tools were deployed to dial telephone number ranges in search of dial-up access pools and modems – called “war dialing”. This attack was dramatized in the movie “War Games.” Today this attack has been augmented by another form of scanning, called “war driving.” Here, the intruder uses a laptop computer and wireless network interface card to drive around or near locations that use wireless networks. Intruders do this for any number of purposes – including attempts to learn network information, view network traffic, interject traffic on the network, and/or deny service availability.



Observation Attacks

- Infrastructure observation
- Observation in transit
- Electronic emanations



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 26

Observation attacks use technical means to observe, record, and copy electronic communications between host systems on networks. These types of attacks are described in the following material.

Infrastructure Observation

“Examining the infrastructure in order to gain information. Examples include watching air ticketing information in order to see when particular people go to particular places and using this as an intelligence indicator, tapping a PBX system in order to record key telephone conversations, and watching for passwords on the Internet in order to gain identification and authentication information to multiple computers. [1]”

Infrastructure observation attacks involve human analysis of the information captured from data, voice, and human networks. This form of attack goes far beyond simple data collection on the network, such as account and password information, and involves more of an intelligence gathering exercise. The attacker, to form an improved picture of the target victim’s systems, networks, and information, would compile information gathered from these multiple sources. Infrastructure attacks may yield such things as:

- Determining what systems across local or wide area networks are involved in the transmission and delivery of network communications between systems of interest to the attacker
- Determining the operational schedule of a network (i.e. at what point in time is network traffic heaviest, what are the maintenance periods and frequency, how often are central log host purged and reviewed, etc)
- Determining the operational abilities of the network operations staff (i.e. how fast does the I/T staff respond to an incident involving X problem, etc)

Observation in Transit

“Examination of information in transit. Examples include network tapping and I/O buffer watching. [1]”

Observation in transit is a subset of infrastructure observation attacks involving primarily data network or system process observation. This form of attack most likely involves an attacker who employs network “sniffer” programs. The intruder places the program on a particular network segment of interest in the hopes of gaining information that can be forwarded on to his/her local or remote location. More technical observation attacks can include observation and capture of system processing information as the system

hands information from one device to another. For example, watching the buffer of a disk array as it hands information to the system bus of a computer.

Electronic Emanations [1]

Electronic emanations, or emissions, attacks involve the collection of electronic radio frequencies from the air in order to reconstruct information, screen-shots, or network packets. This form of attack involves sophisticated knowledge and equipment, as well a requirement of close proximity to the system of interest or network. One of the simplest forms of this attack seen today is observation of 802.11b network information by those who use Wireless networks.

In the early days of emanation attacks, attackers would use sensitive electronic receivers to capture things like keyboard strokes (or rather the frequency exhibited when a key is pressed), monitor/video screen emanations, and system bus emanations.

By far the most common and easiest to execute types of observation is network packet sniffing, which is described in the next section.

Packet Sniffing

When a packet sniffer is present, a copy of all packets that pass by it on the network are covertly captured.

Router

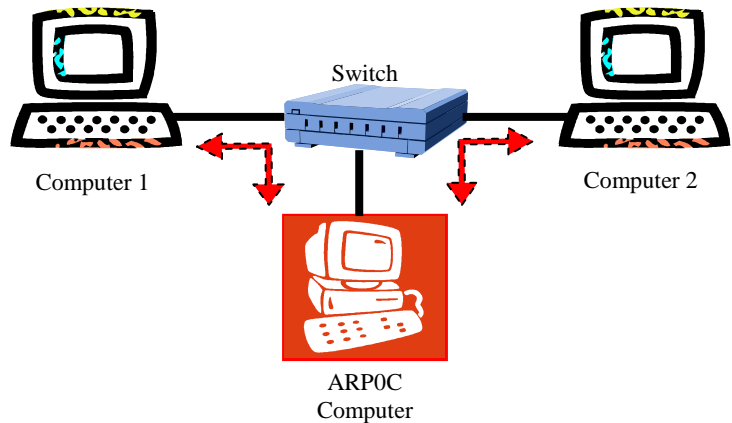
Packet Sniffer Executing

Demo – WebspY & Mailsnarf

© 2002 Carnegie Mellon University Module 8: Threats, Vulnerabilities, and Attacks - slide 27

Packet sniffing is simple observation of information in transit on a network. Most packet sniffing is completed in a passive mode meaning that the information (network packets) is copied and read by a system as it passes by the system on the network.

More direct methods of packet sniffing employ man-in-the-middle attacks. In these cases, a communication is taken over or covertly intercepted by means of network protocol manipulation and then passed on to one of the two endpoints where the original packet was bound. ARPOC, from Phenoelit (<http://www.phenoelit.de/arp0c>), is one such program that manipulates the ARP (address resolution protocol) cache and messages between two systems to set up a “man-in-the-middle” situation. ARPOC, see in the diagram above, intercepts and relays messages between two systems, even across a switched network, because it tricks each machine into believing it is the correct destination for the network messages [4].



Malicious Code Attacks

Viruses

- File infectors
- System and boot infectors
- Macro viruses

Worms

Trojan Horse Attacks

- Downloaded files
- Active content from Web
- E-mail and attachments



Demo – ActiveX (Tiny)

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 28

Malicious code includes viruses, worms and Trojan horses. These types of attacks are described in the following material.

Viruses

A virus is a piece of programming code usually disguised as something else that causes some unexpected and usually undesirable event. The most classical virus definition comes from Dr. Fred Cohen, who coined the term as “a program that can infect other programs by modifying them to include a, possibly evolved, version of itself [7].” The process ultimately describes the replication process and the nature of many viruses to automatically spread to other computer system, users, and their files and programs.

Viruses enjoy a multitude of communication vehicles; they are transmitted as attachments to e-mail messages, embedded within downloaded programs, and exist on a diskette, CD, or other portable media. The Word Macro virus, concept, is a good example of this latter distribution method; some versions of Microsoft® Windows® 95 were distributed with this virus embedded, directly from the manufacturer – shrink-wrapped and all [9]. The sender of an infected mail message or the person who initially downloaded the virus-containing software is usually unaware of the virus’s existence.

The effects of virus infestations cover a range of severity, from the simple annoyance to severely destructive. Depending on the programming involved and the intent of the virus, malicious code can be programmed to execute its “payload” immediately; or remain in a dormant phase until circumstances (time triggers, running antivirus software, and other events) cause the payload to be executed. The innocuous viruses are benign in intent and often seek to inform or entertain (“This one’s for you, Bosco. [8]”). Malicious, malignant viruses can be quite dangerous to computer systems and data; often these viruses destroy, write-over, or hide key system and application files, cause physical damage to hard drives, BIOS firmware, or tamper with the operating system to change system performance. Certain, maliciously crafted viruses are also well aware of the anti-virus techniques (integrity checking, heuristics, signature database) used to detect them and seek to allude and disable commercial anti-virus products.

In general, there are three main classes of viruses:

File infectors

A file infector virus “attaches itself to, or associates itself with, a file.” These viruses usually append or prepend themselves to regular program files or overwrite program code. The file-infector class is also

used to refer to programs that do not physically attach to files but associate themselves with program filenames. Some file infector viruses attach themselves to program files, selecting .COM or .EXE files. Others infect any program for which execution is requested, including .SYS, .OVL, .PRG, and .MNU files. When the program is loaded, the virus is loaded as well. Still other file infector viruses arrive as wholly contained programs or scripts sent as an attachment to an e-mail note. [10]

System or boot-record infectors

These viruses infect executable code found in certain system areas on a disk. They attach to the DOS boot sector on diskettes or the Master Boot Record on hard disks. A typical scenario is to receive a diskette from an innocent source that contains a boot disk virus. When your operating system is running, files on the diskette can be read without triggering the boot disk virus. However, if you leave the diskette in the drive, and then turn the computer off or reload the operating system, the computer will look first in your A drive, find the diskette with its boot disk virus, load it, and make it temporarily impossible to use your hard disk. (Allow several days for recovery.) This is why you should make sure you have a bootable floppy.

Macro viruses

These are among the most common viruses, and they tend to do the least damage. Macro viruses infect your Microsoft Word application and typically insert unwanted words or phrases.

Worms

Worms are a subcategory of viruses. Worms are similar but different from traditional viruses in that they self-replicate but do so entirely; this means that worms do not infect other files but reproduce themselves from one system to the next. Within recent years, many worms have become virus-worm hybrids – they replicate and propagate across a network as a traditional worm but leave destruction payloads as a traditional virus.

The following sections examine two specific worms, the Love Letter Worm and the Nimda Worm.

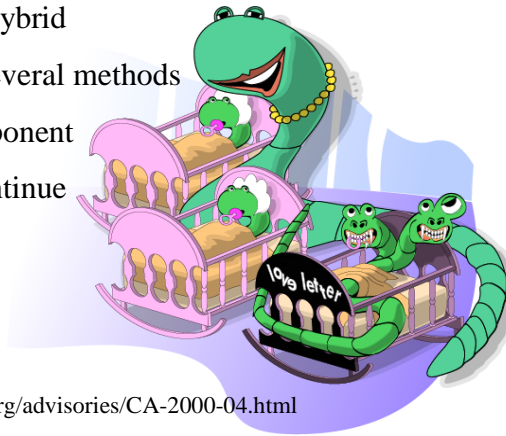
Trojan Horse Programs

“Unintended components or operations are placed in hardware, firmware, software, or wetware causing unintended and/or inappropriate behavior. Examples include time bombs, use or condition bombs, flawed integrated circuits, additional components on boards, additional instructions in memory, operating system modifications, name overloaded programs placed in an execution path, added or modified circuitry, mechanical components, false connectors, false panels, radios placed in network connectors, displays, wires, or other similar components. [1]”



Love Letter Worm

- Malicious code hybrid
- Propagates via several methods
- Uses social component
- New variants continue to be discovered



See: <http://www.cert.org/advisories/CA-2000-04.html>

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 29

The Love Letter worm is a good example of virus-worm hybrid form of malicious code (or malwares). This worm not only replicated itself via network means but also carried a malicious payload, acting like the traditional virus.

Love Letter had the potential to generate large amounts of e-mail and modify entries in the registry in Microsoft Windows operating systems. The worm propagated by several methods, including via: e-mail in an attachment to the message, infected files transmitted by network or physical media, or Internet Relay Chat (IRC) channel communications. Additionally, the worm was easily modified to avoid detection by anti-virus programs that are signature-based, making re-infections possible.

One of the symptomatic problems seen with viruses, Trojan Horses, and worms (like Love Letter), is their exploitation of human engineering to continually propagate. Love Letter takes some form of human interaction to execute and propagate, so the countermeasures that do not include security awareness training but rely on anti-virus software alone will never solve the problem entirely.

In the next section we will examine another hybrid worm, the “nimda” worm.



Nimda (aka Concept Virus)

Vulnerable on multiple operating systems

- Windows 9x, Me, NT, 2000

Multiple methods for propagation

- E-mail
- Web browser
- File system

Exploited known vulnerabilities



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 30

“Nimda,” so called because of its backwards spelling of the word “admin,” is another example of the multiple capabilities of malicious code. The worm, first identified in September, 2001, caused wide spread denial-of-service to computer systems because of its propagation and execution.

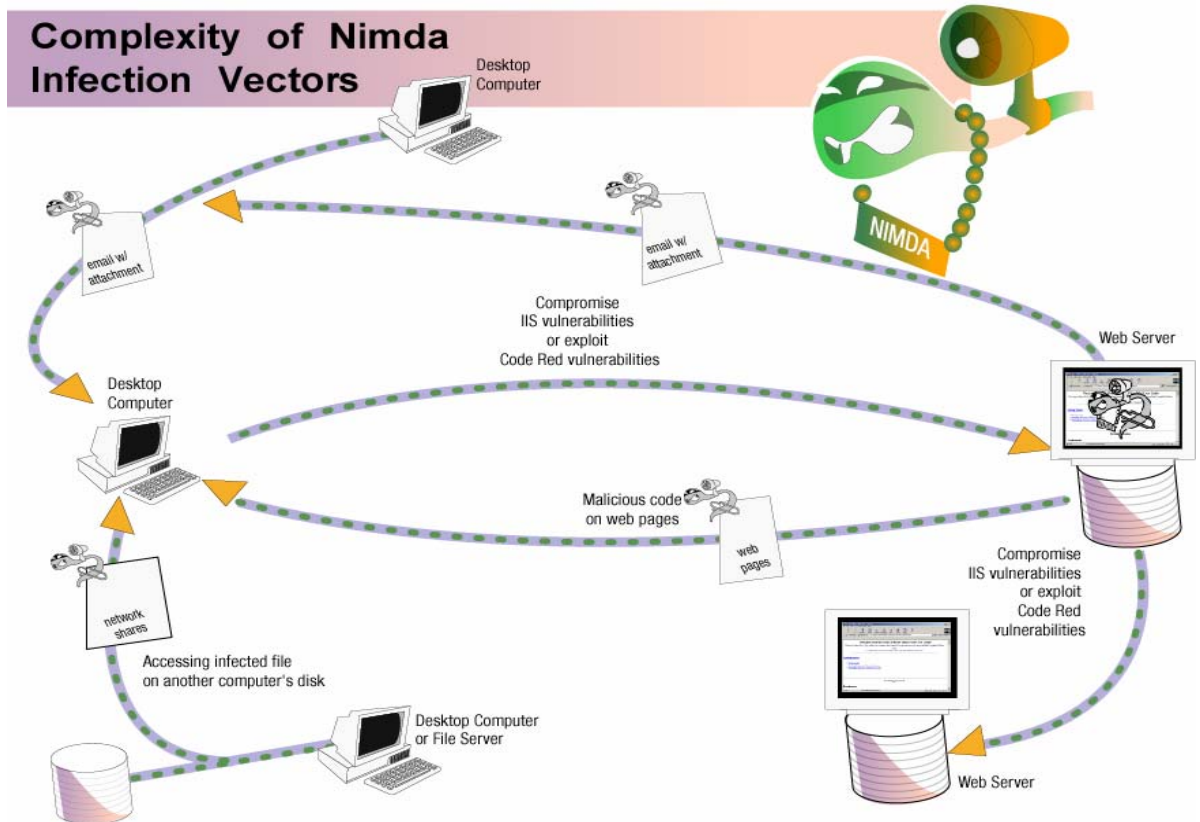
One of the main reasons for the worm’s impact was that the “worm had the potential to affect both user workstations (clients) running Windows ® 95, 98, ME, NT, or 2000 and servers running Windows NT and 2000 [12].” Another reason was that Nimda had the ability to propagate through e-mail, user Web browsing of infected file and Web servers, and file system infections.

The following technical information was provided by the CERT Coordination Center®, in an advisory on Nimda, initially published on September 18, 2001 (available at <http://www.cert.org/advisories/CA-2001-26.html>) [12]:

Overview

“The CERT/CC has received reports of new malicious code known as the "W32/Nimda worm" or the "Concept Virus (CV) v.5." This new worm appears to spread by multiple mechanisms:

- From client to client via email
- From client to client via open network shares
- From web server to client via browsing of compromised web sites
- From client to web server via active scanning for and exploitation of various Microsoft IIS 4.0 / 5.0 directory traversal vulnerabilities ([VU#111677](#) and [CA-2001-12](#))
- From client to web server via scanning for the back doors left behind by the "Code Red II" ([IN-2001-09](#)), and "sadmin/IIS" ([CA-2001-11](#)) worms



The worm modifies web documents (e.g., .htm, .html, and .asp files) and certain executable files found on the systems it infects, and creates numerous copies of itself under various file names.

The Nimda worm has the potential to affect both user workstations (clients) running Windows 95, 98, ME, NT, or 2000 and servers running Windows NT and 2000.

Email Propagation

This worm propagates through email arriving as a MIME "multipart/alternative" message consisting of two sections. The first section is defined as MIME type "text/html", but it contains no text, so the email appears to have no content. The second section is defined as MIME type "audio/x-wav", but it contains a base64-encoded attachment named "readme.exe", which is a binary executable.

Due to a vulnerability described in [CA-2001-06](#) (Automatic Execution of Embedded MIME Types), any mail software running on an x86 platform that uses Microsoft Internet Explorer 5.5 SP1 or earlier (except IE 5.01 SP2) to render the HTML mail automatically runs the enclosed attachment and, as result, infects the machine with the worm. Thus, in vulnerable configurations, the worm payload will automatically be triggered by simply opening (or previewing) this mail message. As an executable binary, the payload can also be triggered by simply running the attachment.

The email message delivering the Nimda worm appears to also have the following characteristics:

- The text in the subject line of the mail message appears to be variable.
- There appear to be many slight variations in the attached binary file, causing the MD5 checksum to be different when one compares different attachments from different email messages. However, the file length of the attachment appears to consistently be 57344 bytes.

The worm also contains code that will attempt to resend the infected email messages every 10 days.

Payload

The email addresses targeted for receiving the worm are harvested from two sources

- The .htm and .html files in the user's web cache folder
- The contents of the user's email messages retrieved via the MAPI service

These files are passed through a simple pattern matcher which collects strings that look like email addresses. These addresses then receive a copy of the worm as a MIME-encoded email attachment. Nimda stores the time the last batch of emails were sent in the Windows registry, and every 10 days will repeat the process of harvesting addresses and sending the worm via email.

Likewise, the client machines begin scanning for vulnerable IIS servers. Nimda looks for backdoors left by previous IIS worms: Code Red II [[IN-2001-09](#)] and sadmind/IIS worm [[CA-2001-11](#)]. It also attempts to exploit various IIS Directory Traversal vulnerabilities ([VU#111677](#) and [CA-2001-12](#)). The selection of potential target IP addresses follows these rough probabilities:

- 50% of the time, an address with the same first two octets will be chosen
- 25% of the time, an address with the same first octet will be chosen
- 25% of the time, a random address will be chosen

The infected client machine attempts to transfer a copy of the Nimda code via tftp (69/UDP) to any IIS server that it scans and finds to be vulnerable.

Once running on the server machine, the worm traverses each directory in the system (including all those accessible through file shares) and writes a MIME-encoded copy of itself to disk using file names with .eml or .nws extensions (e.g., readme.eml). When a directory containing web content (e.g., HTML or ASP files) is found, the following snippet of Javascript code is appended to every one of these web-related files:

```
<script language="JavaScript">
window.open("readme.eml", null, "resizable=no,top=6000,left=6000")
</script>
```

This modification of web content allows further propagation of the worm to new clients through a web browser or through the browsing of a network file system.

In order to further expose the machine, the worm

- Enables the sharing of the c: drive as C\$
- Creates a "Guest" account on Windows NT and 2000 systems
- Adds this account to the "Administrator" group.

Furthermore, the Nimda worm infects existing binaries on the system by creating Trojan horse copies of legitimate applications. These Trojan horse versions of the applications will first execute the Nimda code (further infecting the system and potentially propagating the worm), and then complete their intended function.

Browser Propagation

As part of the infection process, the Nimda worm modifies all web content files it finds (including, but not limited to, files with .htm, .html, and .asp extensions). As a result, any user browsing web content on the system, whether via the file system or via a web server, may download a copy of the worm. Some browsers may automatically execute the downloaded copy, thereby infecting the browsing system.

File System Propagation

The Nimda worm creates numerous MIME-encoded copies of itself (using file names with .eml and .nws extensions) in all writable directories (including those found on a network share) to which the user has access. If a user on another system subsequently selects the copy of the worm file on the shared network drive in Windows Explorer with the preview option enabled, the worm may be able to compromise that system.

Additionally, by creating Trojan horse versions of legitimate applications already installed on the system, users may unknowingly trigger the worm when attempting to make use of these programs.

More information on Nimda can be found at <http://www.cert.org/advisories/CA-2001-26.html>.

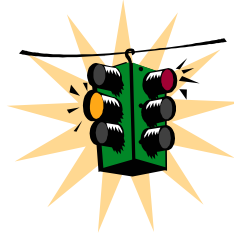
Denial-of-Service Attacks

Potential affects

- Loss of availability
- Loss of the ability to respond to requests
- Unstable operating environment
- Failure or shutdown

Vulnerabilities exploited

- Limited and consumable resources
- Security interdependencies
- Protocol weaknesses
- Environmental conditions
- Physical insecurities



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 31

The traditional intent and impact of DoS attacks is to prevent or impair the legitimate use of computer or network resources. Regardless of the diligence, effort, and resources spent securing against intrusion, Internet connected systems face a consistent and real threat from DoS attacks because of two fundamental characteristics of the Internet.

- The Internet is comprised of limited and consumable resources
The infrastructure of interconnected systems and networks comprising the Internet is entirely composed of limited resources. Bandwidth, processing power, and storage capacities are all common targets of DoS attacks designed to consume enough of a target's available resources to cause some level of service disruption. An abundance of well-engineered resources may raise the bar on the degree of an attack must reach to be effective, but today's attack methods and tools place even the most abundant resources in range for disruption.
- Internet security is highly interdependent

DoS attacks are commonly launched from one or more points on the Internet that are external to the victim's own system or network. In many cases, the launch point consists of one or more systems that have been subverted by an intruder via a security-related compromise rather than from the intruder's own system or systems. As such, intrusion defense not only helps to protect Internet assets and the mission they support, but it also helps prevent the use of assets to attack other Internet-connected networks and systems. Likewise, regardless of how well defended your assets may be, your susceptibility to many types of attacks, particularly DoS attacks, depends on the state of security on the rest of the global Internet

Denial-of-service attacks can include any physical or technological attack that renders a system or network unavailable for use by those who would have legitimate access. Denial-of-service involves any attacks that cause: a loss of availability, a loss of ability to respond to service or communication requests, a consumption of the scarce resources of a system or network that create an unstable system environment, or forcing failure or shutdown of a system that contains a needed information asset or conduit to an asset (required to satisfy or provide the delivery of the asset).

Student Workbook – Module 8: Threats, Vulnerabilities, and Attacks

The following are examples of the many types of physical and technological attacks that render unavailability:

- Forced power failures that cut power from systems and networks
- Theft or destruction of hardware used to retrieve, store, process, or transmit information
- Network bandwidth saturation by protocol manipulation (malformed TCP packet headers, Smurf, and UDP bounce attacks)
- Large-scale, highly coordinated distributed attacks via multiple networks, geographic locations, and/or persons or software agents

The following section lists some examples of specific technological denial-of-service attacks.

Examples of Denial of Service Attacks

- Mail bombs
- Ping floods (e.g. “Smurf” attacks)
- SYN attacks
- UDP bounce attacks
- Distributed denials of service



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 32

This section introduces some of the more common and recurring denial-of-service attacks, as well as the traditional attacks used to explain network-based DoS attacks.

Mail Bombs

Mail bombs flood a person’s or organization’s e-mail account with messages, in quantity and size, in such a manner as to deny the usefulness of the account or in an attempt to hinder the functionality of the user. Typically, this type of attack is done via a mail server that relays mail messages thereby obfuscating the attacker. Falsifying someone’s subscription to an e-mail message service (Listserv, mailing list, etc) is another way to increase the volume of undesired messages to a user.

One variant of mail bombing is called email “spamming.” Email “spamming” refers to situations when an attacker sends mail (either with a real or spoofed account) to hundreds or thousands of users (or to lists that expand to that many users). Email spamming can be made worse if recipients reply to the email, causing all the original addressees to receive the reply. It may also occur inadvertently, as a result of sending a message to mailing lists and not realizing that the list explodes to thousands of users or other lists, or as a result of an incorrectly set-up responder message (such as vacation(1)). [6]

Ping Floods

Ping (or Packaged INternet Gropper) floods overload a system or network segment with such an amount of ping request and replys that the volume consumes the available network bandwidth and/or ties up the targeted system from being able to respond.

SYN Attacks

SYN (or synchronize) attacks take advantage of the manner in which the TCP (Transmission Control Protocol) protocol begins a communication session. In the normal three-way handshake process of TCP: 1) the client sends a SYN packet to a server requesting a service connection, 2) the server, if available and offering the service, replies with a SYN/ACK packet (acknowledging the synchronization request), and 3) the client sends a final ACK packet to acknowledge communications and begin a session. Abnormal handshakes, called half-open connection requests (caused by SYN attacks), create a state problem where the server receives a client’s SYN packet but never receives a subsequent ACK packet from the same client. If the server receives enough SYN packets without follow-on ACK packets, the server will

allocate connection slots to a point where it can no longer respond to further communication requests, thereby resulting in an unavailability of a service.

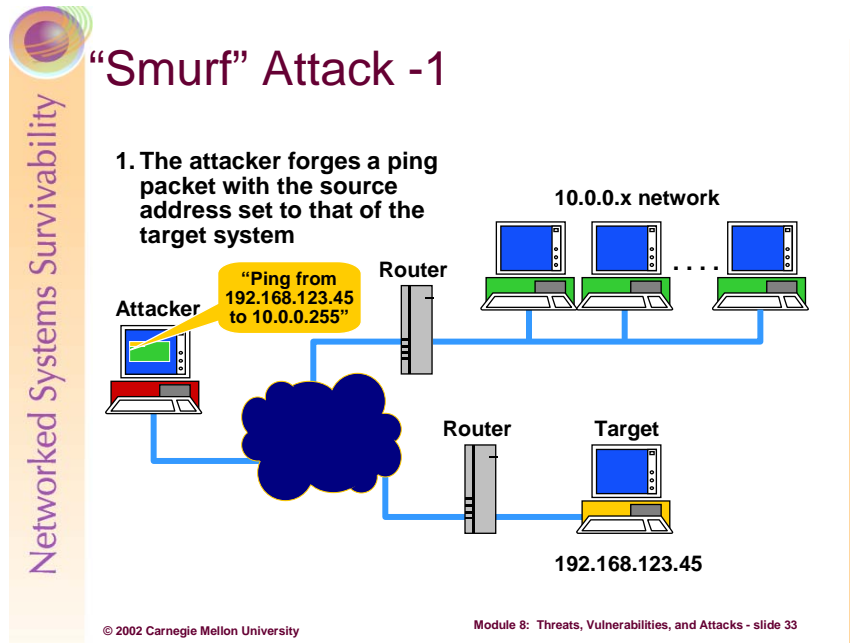
UDP Bounce Attacks

UDP (User Datagram Protocol) bounce attacks are one of the earliest forms of network-based denial-of-service attack. Here a forge (spoofed packet) is generated to cause a system to generate a rapidly increasing stream of network traffic between two hosts on a (typically) local network. This attack uses the Chargen and Echo ports of UDP.

Distributed Denial-of-Service (DDoS) Attacks

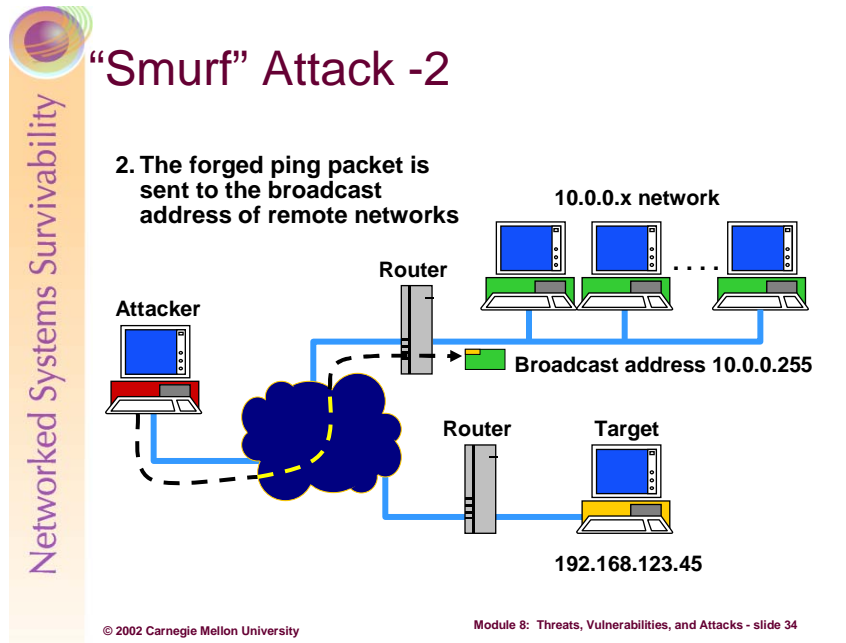
Distributed denial-of-service attacks employ one or more means of denial-of-service attack methods generally from multiple locations (both networked and geographically). A typical DDoS attack is discussed in more detail in a further section.

In the next few sections, we will discuss “Smurf,” SYN Flood, UDP Bounce, and DDoS attacks in more detail.



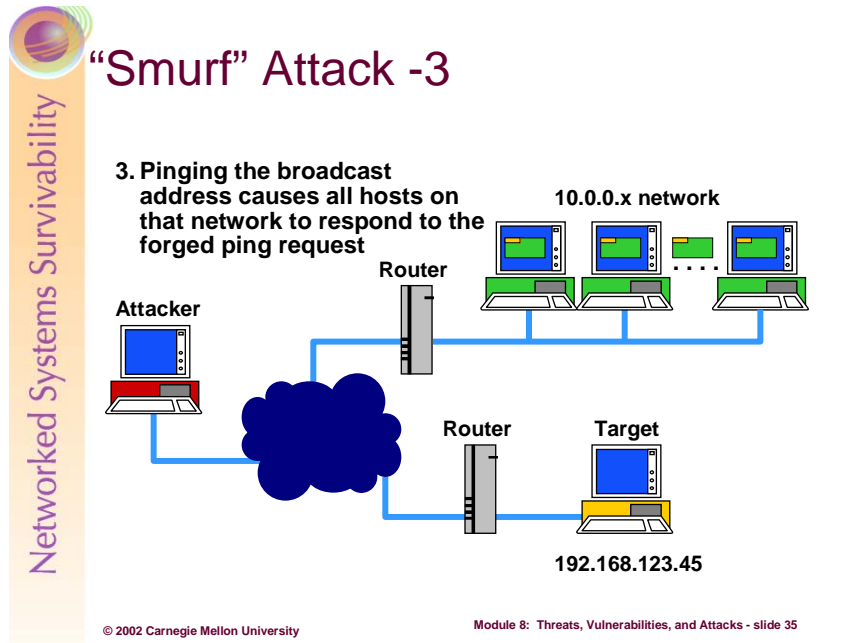
“Smurf” attacks get their name from the attack program originally created to cause this denial-of-service (DoS) attack. This attack abuses the ICMP protocol by directing a forged, spoofed, network packet at the broadcast address of a network. As a condition to this attack working effectively against a target system, the attacker must first find an intermediary router configured (willing) to direct broadcast request to the network(s) to which it connects [11].

Viewing the slide above, the attack is started when the attacker sends out a PING (Packaged INternet Gropper) packet using ICMP Echo (described in further detail in Module 5 – TCP/IP Security). This PING packet is directed at the broadcast address (10.0.0.255 – in this case) of the 10.0.0.x network, as shown above, and has a source IP address equal to the target (victim) computer system.



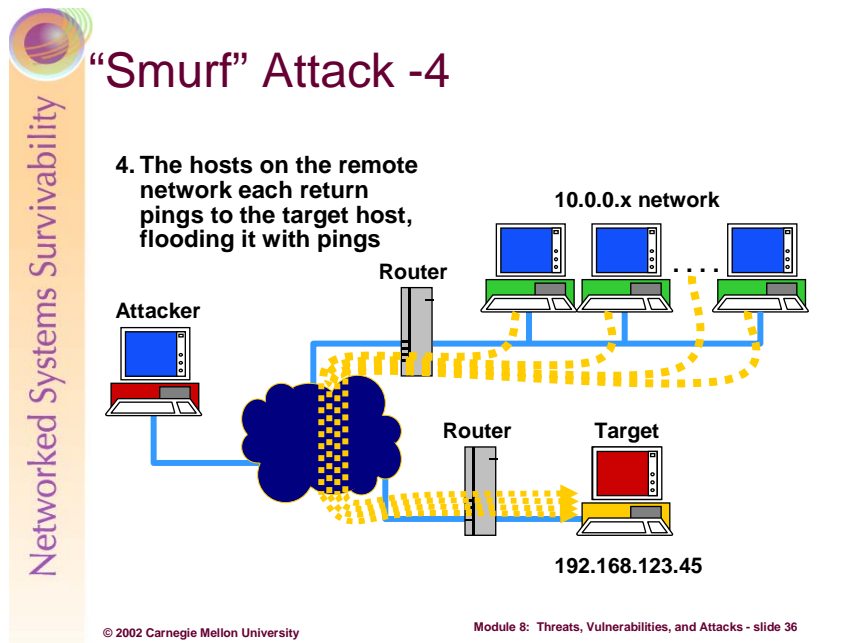
“Smurf” Attacks (Continued)...

Here we see the first part of the attack; as the packet is received and forwarded to the broadcast address of the 10.0.0.x (green) network.



“Smurf” Attacks (Continued)...

As seen above, each system receiving the ICMP Echo request inspects the packet and identifies that the target system (192.168.123.45) sent the packet. Of course, the attacker forged this information but the systems in the 10.0.0.x network have no manner in which to verify the authenticity of the network packets received. The effect is that the systems believe the packet information is factual and must reply to all broadcast packets received...

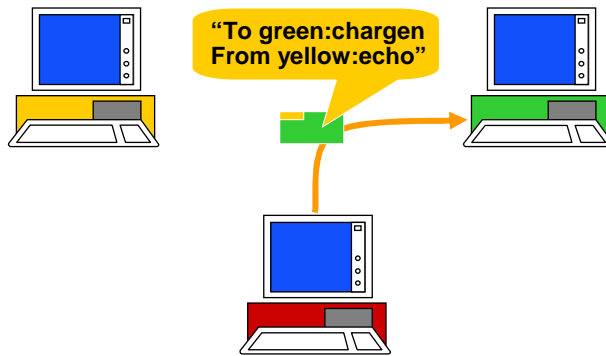


“Smurf” Attacks (Continued)...

Finally, for each initial ICMP Echo request received by the systems on the 10.0.0.x network, each system will reply with an ICMP Echo Reply packet to the source of the original request – our target system in this case. The result is a denial-of-service caused by the amplification of spoofed ICMP Echo requests. The magnitude of the DoS attack is dependent on the size of the network and number of ICMP Echo requests received by the network that it will respond to. In addition to the target system being affected by the DoS, the intermediary router can be “inundated with traffic, resulting in degraded network service availability. [11]



UDP Bounce Attacks -1



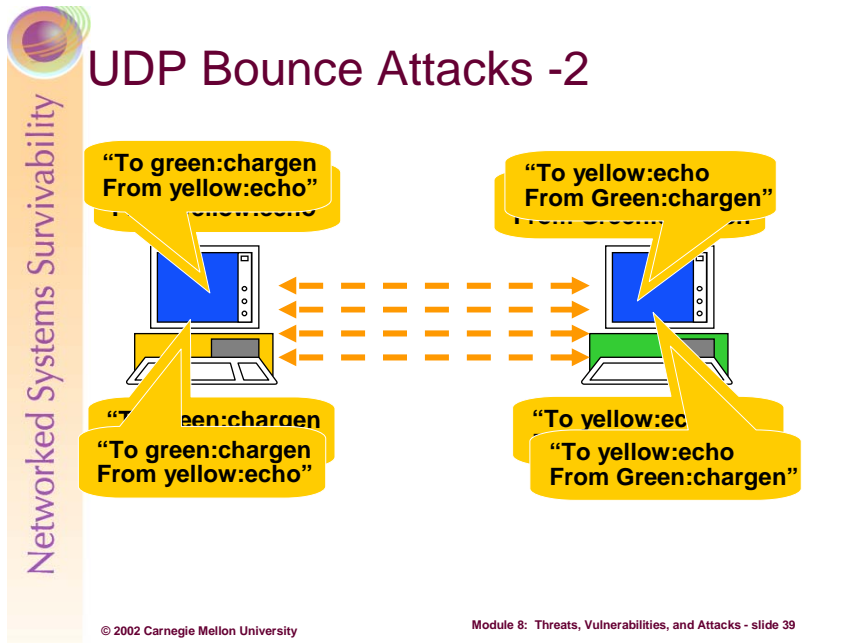
© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 38

A third type of DoS attack is the UDP Bounce attack. This type of attack is still seen within local area networks due to mis-configuration, inadequate host hardening, and default configuration in commercial operating systems.

The first step of the UDP Bounce attack involves the attacker sending a forged network packet that is addressed to the *chargen* port of one target system (in green above), claiming to originate from the *echo* port of the other target system (in yellow above.)

Note: The UDP Echo service should not be confused with the ICMP Echo / Echo Reply services. Especially for one important reason, the UDP Echo service does not respond with ICMP messages. The UDP Echo service sends exactly the same information out as it received.



UDP Bounce Attacks (Continued)...

The second, and subsequent, part of the attack basically involves the same properties, with the exception that each iteration of the attack becomes amplified. The target (in yellow above) receiving the initial forged (spoofed) network packet responds by sending a number of packets to the *echo* port of the other target (in green above). Every packet received on the *echo* port is returned back to the *chargen* port of the first target (green). Each packet sent the *chargen* (a UDP service that responds by generating a fixed amount of ASCII characters) port gets several packets back. The situation very quickly works from: one packet spawns many packets to many packets spawn even more packets, and so on.

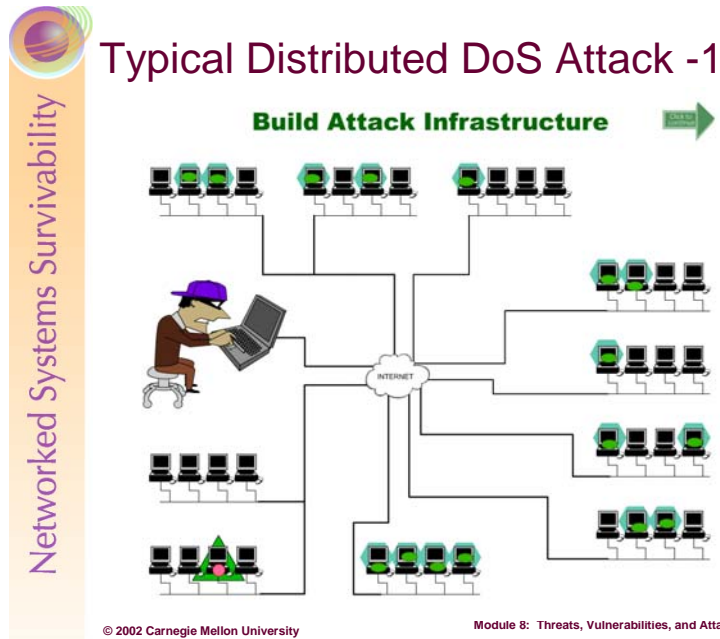
The net effect is a localized denial-of-service caused by an increasing flood of traffic between the two systems involved, rendering both systems unable to respond. In addition, the extreme volume of traffic generated between the targets affects network connectivity of other systems that share the same network segment.

Typical Distributed DoS Attack



Click for Animation





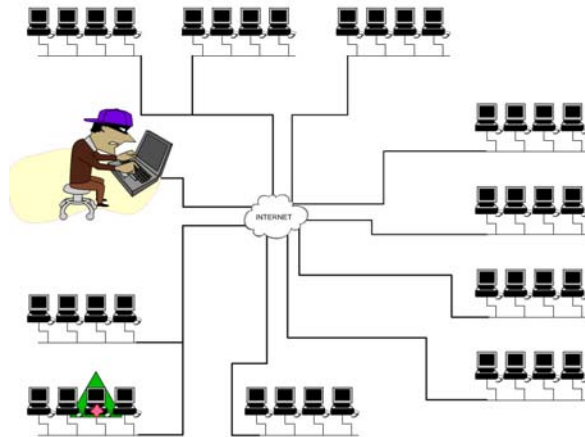
Distributed denial-of-service (DDoS) attacks extend the methods of DoS attacks and use large numbers of distributed systems and networks to affect attacks against systems and networks over a (typically) large, if not global, region across multiple networks and/or the Internet. Some instances of DDoS include:

- An attacker who has previously compromise tens, hundreds, or thousands of computer systems to act as agents which generate large amounts of network traffic bound for a selected target(s)
- An attacker who redirects network traffic through technological manipulation to direct large amounts of network traffic toward a selected target(s)
- An attacker who coordinates a large number of people (such as activists or hackers) to direct large amounts of network traffic toward a selected target(s). For example, in retaliation for the inadvertent bombing of the Chinese Embassy in Yugoslavia, thousands of Chinese students crippled the Whitehouse.gov Web site using Web browsers to generate more HTTP requests to the server than could be responded.



Typical Distributed DoS Attack -2

Step 1: Intruder to Handler



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 42

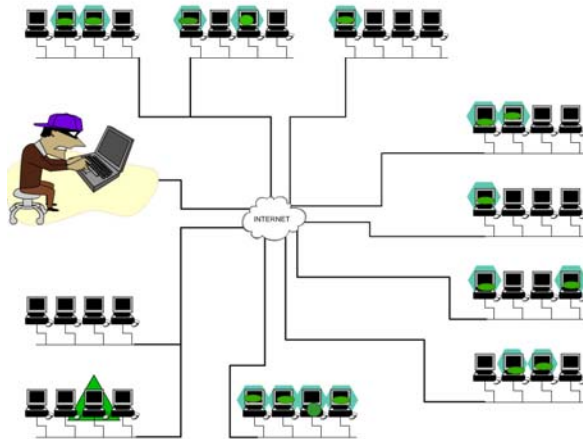
In a typical DDoS attack scenario an intruder performs the following:

1. The intruder researches and selects a target for the attack
2. The intruder compromises tens, hundreds, or thousands of computer systems to act as agents – which launch and sustain the network-based attacks through packet generation – through manual or automated (worms, DDoS tool suites, etc) means



Typical Distributed DoS Attack -3

Step 2: Handler to Agents



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 43

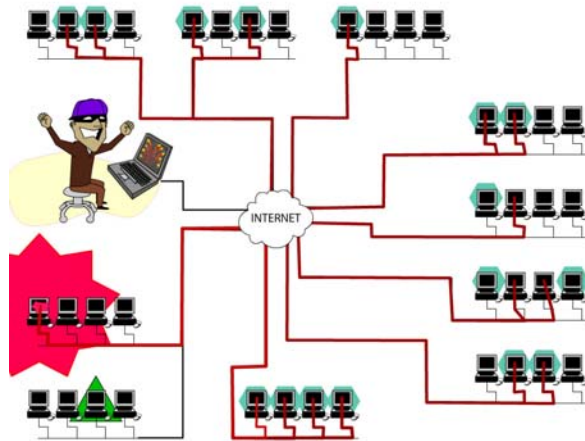
Typical DDoS Attack Scenario (Continued)...

3. The intruder sets up one of the compromised systems to act as a master controller for the attack
4. The intruder sends the master controller a predetermined set of variables that describe: the attack type (method of denial-of-service), the location and number of agents (IP addresses, ports, hostnames), how to communicate/authenticate with the agents, duration of the attack, whether systems should falsify their network identities (through IP spoofing), the time at which to begin the attack, the targeted network or systems, etc.



Typical Distributed DoS Attack -4

Step 3: Agents to Victim



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 44

Typical DDoS Attack Scenario (Continued)...

5. The master controller relays this attack definition to each DDoS agent computer
6. The agent computers then wait until the predetermined attack time and begin their DDoS attack against the targeted systems or networks
7. The attack commences...

Acquisition and Attack

- Select the target
- Gather information
- Scan and probe
- Research vulnerabilities
- Sequence and align vulnerabilities
- Gather tools and exploits
- Attack the target



© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 41

Let's take a quick look at a fictitious intrusion scenario and relate some of the principles and objectives from this module. Below we will examine some plausible steps in an intrusion starting with after target selection.

Step 1: The intruder, having selected a target for attack, starts by using public sources of information on the Internet to learn more about the target, its location, its defenses, and/or its security management practices. One first step might be a “whois” lookup to determine facts about the organization, such as:

- Name of the organizations Internet Service Provider (ISP)
- Physical location of the organization
- Names of the billing and technical points-of-contact
- Names and size of the IP block allotted to the organization

Some other useful sources of information for the attacker in this first step may include:

- The organization's Web site
- Web forum or e-mail lists to which the organization's systems and security professionals publish or request information
- Socially engineered information from the organization's employees, contractors, vendors, service agencies, etc
- Articles or other publications that document information about the organization

Step 2: The intruder may begin network probing and scanning on the IP address range of the organization in order to enumerate services (which may be identified later as vulnerable to some form of attack; or the intruder may opt out of this step by having determined this information from previously obtained information (network maps from dumpster diving, IP addresses and account names through social engineering, etc).

Step 3: The intruder, having enumerated the target systems or networks for ports, services, and/or vulnerabilities, now conducts research to determine how best to launch an attack (i.e. they determine what chain of events is necessary to reach and achieve their goals – to disclose, destroy, modify, abuse, make unavailable, the resources of the target).

Student Workbook – Module 8: Threats, Vulnerabilities, and Attacks

Step 4: After the research is concluded (which can range from minutes to days to months), the intruder most likely aligns the vulnerabilities with the desired outcome necessary at each step of the intrusion. For example, if the attacker's goal is to leverage a DDoS attack against the target site, then the intruder would first compromise the rogue computers to act as controllable agents in the attack.

Step 5: When the sequence of vulnerabilities is determined, the attacker must gather an arsenal of tools, exploit scripts, and/or command-line inputs to affect the vulnerabilities in an attack.

Step 6: Finally, the attacker uses the gathered tools and their knowledge of the organization's networks and systems to launch an attack.

Step 7 and beyond: Step 6 rightly concludes with a successful attack by the intruder against a target system, but the intruder may not yet have achieved their overall mission objectives. After a successful compromise or intrusion, the attacker may decide to initiate any number of activities, such as:

- Use the access gained to launch further attacks against similar systems and networks, both inside and outside of the currently compromised organization
- Use the access gained to launch direct attacks against a local system's resources and assets in order to destroy, breach, or tamper them
- Use the access gained to set up a staging area for future attacks, or use the current system in a future denial-of-service attack

Review Questions

1. What is a threat actor and what are some of the named threat actors?
2. What are some of the different threats to integrity?
3. What are at least two vulnerabilities discussed in this module?
4. What are at least two forms of attacks used by intruders?
5. What are two examples of Denial-of-Service attack discussed in this module?

© 2002 Carnegie Mellon University

Module 8: Threats, Vulnerabilities, and Attacks - slide 42

Review Questions

1. What is a threat actor and what are some of the named threat actors?

A threat actor is any person who leverages, creates, or affects an undesirable disclosure, destruction, modification, or interruption of information, systems, or networks. Threat actors include anyone (with the means and opportunity) to cause the undesired outcome, independent of their intent or motivation – activists to hackers to reports to tiger teams.

2. What are some of the different threats to integrity

- a. Unauthorized or maliciously deliberate modification or destruction of an information asset, system, or network.
- b. Losing the ability to authenticate information or the ability to verify the integrity of information because of a destruction, modification, or interruption to the mechanisms affording these functions or safeguards

3. What are at least two vulnerabilities discussed in this module?

For this answer, please see examples and descriptions of vulnerabilities on pages 20, 23-25, and 28-30 in this module.

4. What are at least two forms of attacks used by intruders?

For this answer, please see examples and descriptions of attacks on pages on pages 28-66.

5. What are two examples of Denial-of-Service attack discussed in this module?

- a. Mail Bombs
- b. Ping Floods
- c. SYN Attacks
- d. UDP-Bounce attacks
- e. DDoS attacks



Summary

- Common terms
- Threats to information systems
- Threat actors
- Vulnerabilities
- Forms of attack
- Network intrusion scenario example

This module focused on the range of threats, threat actors, vulnerabilities, and attacks leveraged against information system. Common terminology was defined in order to present a consistent language when relating threats and vulnerabilities. This module also introduced the concept of an intruder – someone who has the means, motive and opportunity to affect attacks against a victim. Various types, classes, and strategies for intrusion and attack were also presented by discussing the numerous vulnerabilities and exploits that intruders leverage against information systems and networks. Finally, a scenario was presented at the end of this unit to help construct a robust example of the attack sequence and decisions used by an attacker, in essence the attack picture.

Bibliography:

1. Cohen, F. "All.Net – New Security Database." Fred Cohen & Associates, 1999. Available [online] at <http://all.net>.
2. CERT Coordination Center®, 1999. "CERT® Advisory CA-1999-04 Melissa Macro Virus." Available [online] at <http://www.cert.org/advisories/CA-1999-04.html>.
3. Anonymous. "Buffer Overflow." SearchSecurity.Com. Available [online] at <http://searchsecurity.techtarget.com/>.
4. Anonymous. "ARPOc connection interceptor." Available [online] at <http://www.phenoelit.de/arpoc/>.
5. CERT Coordination Center®. CERT/CC Vulnerability Notes By Metric. Available [online] at <http://www.kb.cert.org/vuls/bymetric>.
6. CERT Coordination Center®. "Email Bombing and Spamming." Available [online] at http://www.cert.org/tech_tips/email_bombing_spamming.html.
7. Cohen, F. "A short course on computer viruses." ASP Press, 1990. Page 11.
8. Anonymous. "WD: Frequently Asked Questions About Word Macro Viruses." Microsoft TechNet. Available at <http://support.microsoft.com/directory/article.asp?ID=KB;EN-US;q187243>.
9. Stalnaker, J. Concept Virus. Available [online] at:
<http://emt.doit.wisc.edu/wordvirusFAQ/wordvirus.FAQ.04.1.html>.
10. Anonymous. Virus Glossary – File Infector. McAfee, Network Associates Technology. Available [online] at <http://www.mcafeeb2b.com/naicommon/avert/avert-research-center/virus-glossary.asp>.
11. Syngress Publishing. "Hack Proofing Your Network: Internet Tradecraft." 2000. Pages 69-70.
12. CERT Coordination Center®. "CERT® Advisory CA-2001-26 Nimda Worm." September, 18, 2001. Available [online] at <http://www.cert.org/advisories/CA-2001-26.html>.