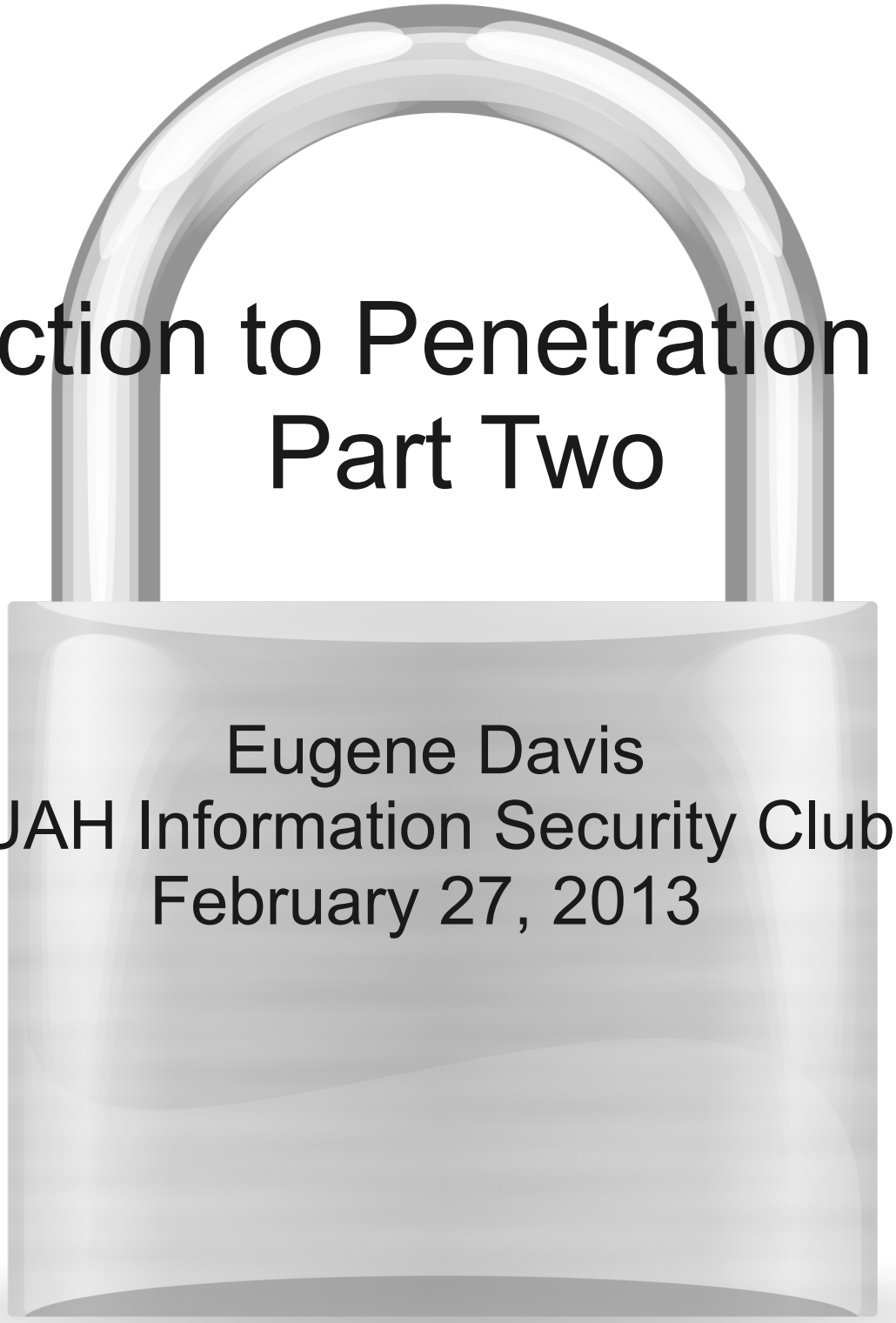


Introduction to Penetration Testing: Part Two

Eugene Davis
UAH Information Security Club
February 27, 2013

Ethical Considerations: Pen Testing

- Ethics of penetration testing center on integrity
- (ISC)² Code Of Ethics Cannons
  - Protect society, the common good, necessary public trust and confidence, and the infrastructure
  - Act honorably, honestly, justly, responsibly, and legally
  - Provide diligent and competent service to principals
  - Advance and protect the profession
- Maintain privacy and confidentiality
- Acquire written permission from appropriate authorities before performing pen testing on systems

Summary of Part One

- Pen testing - looking for weaknesses in systems before the hackers find them
- **Phase 1: Reconnaissance –** passively learning about target's network
  - Searching the Web and dumpster diving
  - Social engineering
- **Phase 2: Scanning –** looking for technological openings in the network
  - Looking for access points
  - Looking for open ports
- **Phase 3: Exploits (OS) –** gaining control over individual machines
  - Buffer overflows
  - Automation (e.g. Metasploit)

Network Attacks

- Sniffing reveals sensitive information
  - Made worse by ARP/DHCP poisoning
- Session Hijacking allows an attacker to take over a communications channel
- DoS attacks take down services
- A DoS attack also can be used as a stepping stone to something nastier

Intruder Access to Network

- Given access to a network, there are many possibilities for attacking systems on it
- Perform session high-jacking
- Spoof ARP, DHCP or DNS
- Spoof legitimate network services
- Perform sniffing to obtain information

Network Sniffing

- Sniffing is bypassing the networking stack to read packets directly
- Can be performed by a compromised machine or attacker's machine
- Bypasses most protections on the network
- Allows attacker to read any plaintext or weakly encrypted communications
- By using ARP poisoning an attacker can reduce the security of a switch

ARP Poisoning

- Switches optimize network communications by sending packets on the port for the recipient
- This prevents an attacker from sniffing traffic on other Ethernet lines
- ARP poisoning rewrites the mapping of ports
  - Attacker indicates that they own the IP address that they wish to spy on
  - Can resend the packets to the original recipient to perform a Man-in-the-Middle (MitM) attack
- Only affects local networks

Session Highjacking

- Session Highjacking is when you take over an existing session between two machines
- Easiest to do on unencrypted channels
  - Wait for session to be established
  - Use ARP poisoning or a DoS attack to knock one party off-line
  - Pretend to be that party you have removed from the connection
- Also by DoS'ing an encrypted session and seizing the IP of one of them, a user may connect to you

Defending from Sniffing & Highjacking

- Use strong authentication protocols
- Encrypt sessions for both confidentiality and integrity
- Manually map MACs to ports on switches
- Keep attackers off the network
  - IDS, Firewalls
  - User education

Denial of Service (DoS)

- Denial of Service is the act of preventing legitimate access to a service
  - Can be performed locally, but this requires the attacker to have access to the target
  - Many network versions exist
- Common in the news are DDoS attacks which coordinate large numbers of machines
  - Often these just fill up the bandwidth

DoS Variants

- DDoS – use many machines (often botnets) to hammer a single target
- TCP Flood – send request packets to target, but never complete the handshake
- Teardrop – uses non-standard fragmenting on the packets that the target cannot handle
- ReDoS – uses malicious regular expressions to consume processing power

Defending from DoS

- A DDoS attack requires coordination with an ISP
- TCP Flooding means dropping open connections or not storing them in memory
- ReDoS is prevented by ensuring the service will not attempt to evaluate malicious expressions

Phase 4: Maintaining Access

- Once a system is taken over, the attacker wishes to maintain control
- Back doors allow an attacker to control the system remotely
  - Botnets create distributed networks with back doors
- Rootkits hide the control software used in maintaining control
- Covered in more depth in the Malware talk

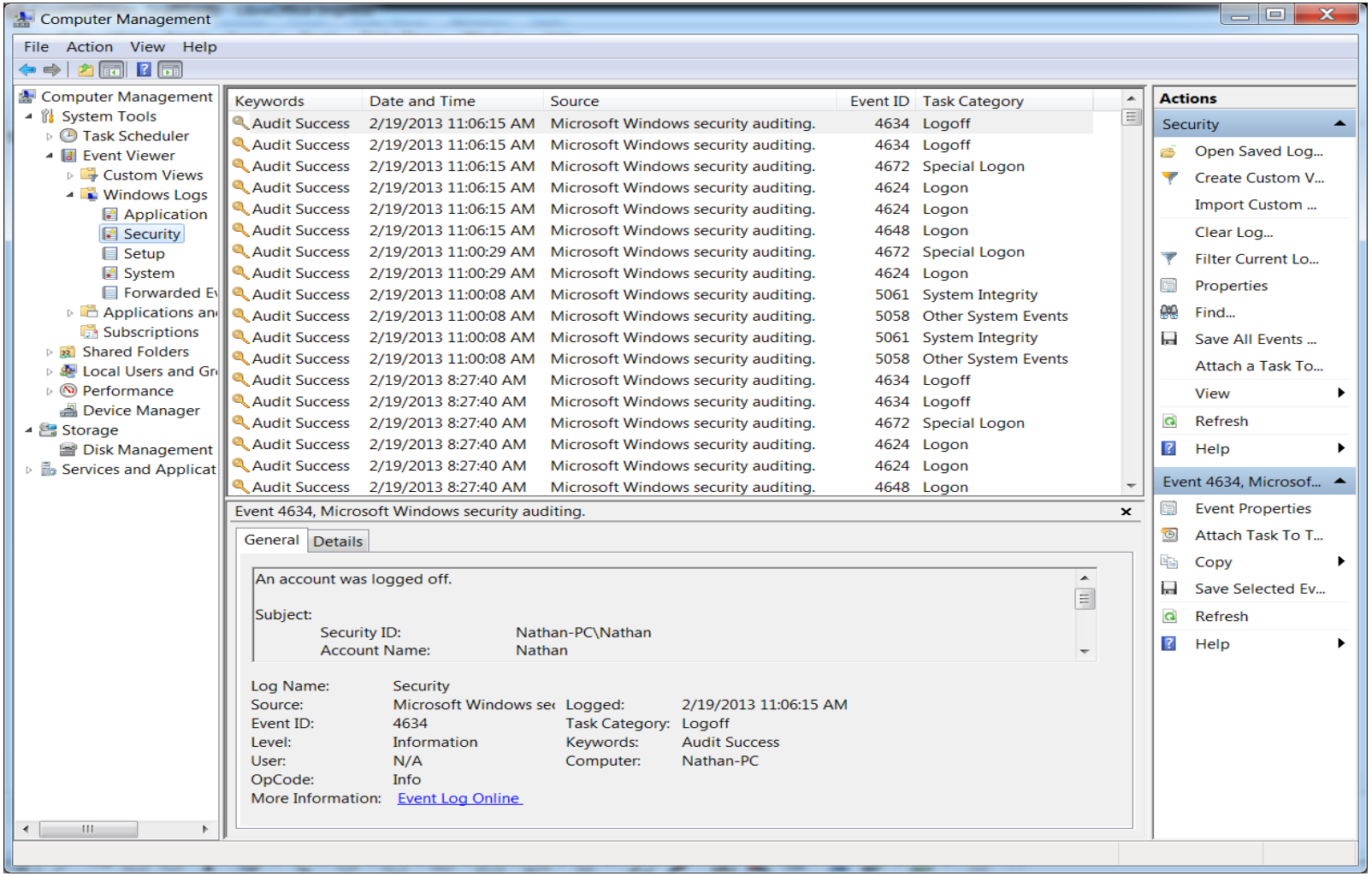
Phase 5: Removing Evidence

- Having seized control the final stage of an attack is cleaning up
- An obvious approach is the removal of log files
- Covert channels can use protocols in unexpected ways to send information out
- Using relays to hide attacker's location

Modification to Computer Log Files

- Log files are generated by system processes and applications to record system activity
- Log files are beneficial when troubleshooting a system or looking for inappropriate activity
- An attacker will attempt to modify system logs to remove any traces of their activity

Computer Log Files



Security Event Log File from Windows

Defenses

- Use write only media
  - Data can be written, but never modified
- Access Control List (ACL)
  - Assign specific permissions for access to objects
- Remote log server
  - Servers configured to accept logs from remote hosts
- Consistent backup of system logs to separate media

Hidden Files & Covert Channels

- Hidden files are ordinarily out of sight or hard to find
- Can be used for storage of attack tools
- As well as other system information or sniffed passwords

Hidden Files & Covert Channels

- Covert channels are paths of communication that exist in a way developers didn't intend
  - Hidden from access controls
  - Can not be detected or controlled by hardware security
  - Can tunnel through secure operating systems
- Not to be confused with legitimate channel exploitation

Hidden Files & Covert Channels

- Covert channels exist when:
  - Sender and receiver have a shared resource
  - Communication can be synchronized between sender and receiver
  - When system administrators can't detect
- Not used for quick data retrieval
- Allows the attacker to continue to receive data and up-to-date information without detection

Covert Channel Examples

- Internet Control Message Protocol (ICMP) tunneling
  - Bypasses firewall
  - Loki provides shell access over ICMP
- Reverse WWW Shell
  - Allows internal network access from outside
  - Requires installation of a trojan program on the network

Defenses

- Hidden Files
  - File integrity check
  - Internal Intrusion Detection System (IDS)
- Covert Channels
  - Anti-virus; IDS; Up-to-date and patched systems
  - No effective defense once an attacker has access
  - Manually removed if found

Use of Relays to Conceal Source Location

- An attack from A to B can be routed through outside machines called relays
- Indirect attack is harder to trace than direct attack
- Additional complications arise when path relays are placed within different jurisdictions

Netcat Relay

- Uses the network utility network to relay traffic
- On relay machine, runs the command:
  mknod buffer p
  - This command creates a file to act as a buffer
- Next, on the same machine, run:
  nc -l -p [recv_port] 0<buffer | nc [dest_ip] [send_port] 1>buffer
  - Which receives traffic on the first ports and forwards it to the destination IP

Conclusions

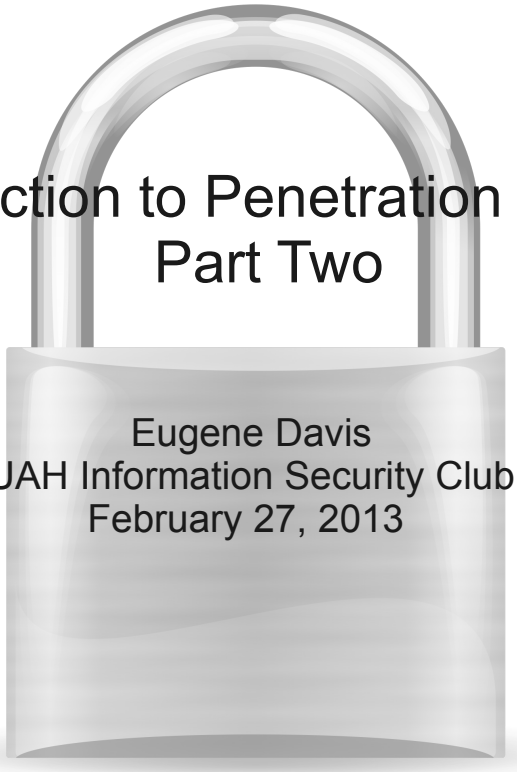
- Pen testing requires adherence a code of ethics
- Pen testing can help find holes in your system before attackers can exploit them
- Pen testing is divided into five phases:
  1. Reconnaissance
  2. Scanning the Target
  3. Exploits and Compromises
  4. Maintaining Control
  5. Removing Evidence

References and Links

- CounterHack Reloaded by Ed Skoudis
- Nmap - http://nmap.org/
- Firewalk - http://packetstormsecurity.com/UNIX/audit/firewalk/
- Nessus Vulnerability Scanner - http://www.tenable.com/products/nessus
- (ISC)² Code Of Ethics - https://www.isc2.org/ethics/default.aspx
- EC-Council - http://www.eccouncil.org
- Open Web Application Security Project (OWASP) - https://www.owasp.org

License

Introduction to Penetration Testing: Part Two



Eugene Davis
UAH Information Security Club
February 27, 2013

Ethical Considerations: Pen Testing

- Ethics of penetration testing center on integrity
- (ISC)² Code Of Ethics Cannons
  - Protect society, the common good, necessary public trust and confidence, and the infrastructure
  - Act honorably, honestly, justly, responsibly, and legally
  - Provide diligent and competent service to principals
  - Advance and protect the profession
- Maintain privacy and confidentiality
- Acquire written permission from appropriate authorities before performing pen testing on systems

Ethics in pen testing is a very broad topic that is only covered in very general terms. This is partly due to the fact that the legal system has yet to catch up with technology and new laws regarding information security are being written every day.

There are several certifications out there for pen test engineers and ethical hackers that require an adherence to a code of ethics. Examples: (1) The International Information Systems Security Certification Consortium ((ISC)²) and their Certified Information

Summary of Part One

- Pen testing - looking for weaknesses in systems before the hackers find them
- **Phase 1: Reconnaissance** – passively learning about target's network
  - Searching the Web and dumpster diving
  - Social engineering
- **Phase 2: Scanning** – looking for technological openings in the network
  - Looking for access points
  - Looking for open ports
- **Phase 3: Exploits (OS)** – gaining control over individual machines
  - Buffer overflows
  - Automation (e.g. Metasploit)

Network Attacks

- Sniffing reveals sensitive information
  - Made worse by ARP/DHCP poisoning
- Session Hijacking allows an attacker to take over a communications channel
- DoS attacks take down services
- A DoS attack also can be used as a stepping stone to something nastier

Intruder Access to Network

- Given access to a network, there are many possibilities for attacking systems on it
- Perform session high-jacking
- Spoof ARP, DHCP or DNS
- Spoof legitimate network services
- Perform sniffing to obtain information

Network Sniffing

- Sniffing is bypassing the networking stack to read packets directly
- Can be performed by a compromised machine or attacker's machine
- Bypasses most protections on the network
- Allows attacker to read any plaintext or weakly encrypted communications
- By using ARP poisoning an attacker can reduce the security of a switch

Network stack is what handles the interpretation of packets

Attacker may be able to gain access to physical connectors, but even more common is the discovery of open wireless access points

One of the ways that access may be gained is by social engineering your way into a networked location of the target building

Most protections of a network tend to operate above the level of IP (thus MAC address) and can't help out much. Even if they do filter MACs, the attacker may

ARP Poisoning

- Switches optimize network communications by sending packets on the port for the recipient
- This prevents an attacker from sniffing traffic on other Ethernet lines
- ARP poisoning rewrites the mapping of ports
  - Attacker indicates that they own the IP address that they wish to spy on
  - Can resend the packets to the original recipient to perform a Man-in-the-Middle (MitM) attack
- Only affects local networks

ARP normally sends out a request “hey, who has this IP address”, to which the owner says “Me, here's my MAC address”, but ARP poisoning uses ARP broadcasts - “Hey, I'm IP address and my MAC is...” to gain control of the target's IP

Could be fixed by updating the Address Resolution Protocol

This is almost as good as a hub - where all traffic is just rebroadcasted

Session Highjacking

- Session Highjacking is when you take over an existing session between two machines
- Easiest to do on unencrypted channels
  - Wait for session to be established
  - Use ARP poisoning or a DoS attack to knock one party off-line
  - Pretend to be that party you have removed from the connection
- Also by DoS'ing an encrypted session and seizing the IP of one of them, a user may connect to you

Defending from Sniffing & Highjacking

- Use strong authentication protocols
- Encrypt sessions for both confidentiality and integrity
- Manually map MACs to ports on switches
- Keep attackers off the network
  - IDS, Firewalls
  - User education

Strong authentication is only as good as the user's let it be – good passwords must be used

To keep attackers off the network, users cannot do stupid things that open it up for access

Denial of Service (DoS)

- Denial of Service is the act of preventing legitimate access to a service
  - Can be performed locally, but this requires the attacker to have access to the target
  - Many network versions exist
- Common in the news are DDoS attacks which coordinate large numbers of machines
  - Often these just fill up the bandwidth

DoS Variants

- DDoS – use many machines (often botnets) to hammer a single target
- TCP Flood – send request packets to target, but never complete the handshake
- Teardrop – uses non-standard fragmenting on the packets that the target cannot handle
- ReDoS – uses malicious regular expressions to consume processing power

TCP floods consume the memory handling open TCP connections

ReDoS – because regular expressions require processing power, you might figure out how to cause an infinite loop. For instance, many systems on the web process XML, you might be able to find a way to make them get stuck...

Defending from DoS

- A DDoS attack requires coordination with an ISP
- TCP Flooding means dropping open connections or not storing them in memory
- ReDoS is prevented by ensuring the service will not attempt to evaluate malicious expressions

Ddos attacks need either huge amounts of bandwidth (improbable) or filtering at the ISP level (potentially this isn't even enough). Either way, most organizations cannot survive them

Some TCP versions use stateless connections, which send a hash that uses a secret key that only they have. Once that is sent, the server forgets the TCP connection until a client sends back the cookie. However, this can consume a lot of CPU, so is only pushing the problem around

Phase 4: Maintaining Access

- Once a system is taken over, the attacker wishes to maintain control
- Back doors allow an attacker to control the system remotely
  - Botnets create distributed networks with back doors
- Rootkits hide the control software used in maintaining control
- Covered in more depth in the Malware talk

Basically, everything here has already been covered in the context of the Malware talk

Phase 5: Removing Evidence

- Having seized control the final stage of an attack is cleaning up
- An obvious approach is the removal of log files
- Covert channels can use protocols in unexpected ways to send information out
- Using relays to hide attacker's location

Removing evidence is part of what a pen tester should do, since part of defending a system is being able to determine what an attacker has done, and where possible send law enforcement their way

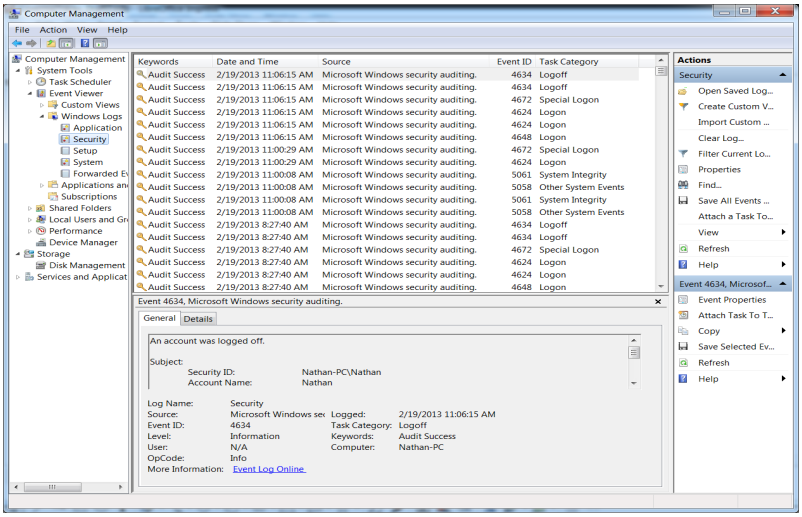
Modification to Computer Log Files

- Log files are generated by system processes and applications to record system activity
- Log files are beneficial when troubleshooting a system or looking for inappropriate activity
- An attacker will attempt to modify system logs to remove any traces of their activity

Log files track user account activities, security audits and application processes. It flags warnings and failures for the things previously mentioned which make them a great tool for troubleshooting and looking for inappropriate activity.

UNIX files are usually in plain text and are easily edited with the correct permissions

Computer Log Files



Security Event Log File from Windows

Screen shot of a security log file from Windows

Defenses

- Use write only media
  - Data can be written, but never modified
- Access Control List (ACL)
  - Assign specific permissions for access to objects
- Remote log server
  - Servers configured to accept logs from remote hosts
- Consistent backup of system logs to separate media

Log files track user account activities, security audits and application processes. It flags warnings and failures for the things previously mentioned which make them a great tool for troubleshooting and looking for inappropriate activity.

Hidden Files & Covert Channels

- Hidden files are ordinarily out of sight or hard to find
- Can be used for storage of attack tools
- As well as other system information or sniffed passwords

For UNIX, pre-pend '.' to the file name
In Windows, select the hidden attribute in the properties tab of the file/folder

Hidden Files & Covert Channels

- Covert channels are paths of communication that exist in a way developers didn't intend
  - Hidden from access controls
  - Can not be detected or controlled by hardware security
  - Can tunnel through secure operating systems
- Not to be confused with legitimate channel exploitation

Legitimate channel exploitations – the use of schemes such as stenography to disguise prohibited objects within legitimate information objects – data hiding

Hidden Files & Covert Channels

- Covert channels exist when:
  - Sender and receiver have a shared resource
  - Communication can be synchronized between sender and receiver
  - When system administrators can't detect
- Not used for quick data retrieval
- Allows the attacker to continue to receive data and up-to-date information without detection

Covert channels suffer from a low rate of data transmission. The downside to this is that it cant be used for quick data retrieval, but unlike a brute force attack that would alert system admins, it allows for a continuous data stream where the attacker can receive more and more data over time and up to date information

Covert Channel Examples

- Internet Control Message Protocol (ICMP) tunneling
  - Bypasses firewall
  - Loki provides shell access over ICMP
- Reverse WWW Shell
  - Allows internal network access from outside
  - Requires installation of a trojan program on the network

The lowest level that a firewall can operate is level 3. This is the network layer in the OSI model and the IP layer in the TCP/IP model.

ICMP Tunneling: Establishes a connection through ping request and reply packets

Loki: Switches between UDP and ICMP on the fly; Supports blow-fish encryption and uses Diffie-Helman

Similar tools to Loki are CCTT (Covert Channel Tunneling Tool) and MSNShell

Reverse WWW Shell: covert channel using HTTP. Typically tries to contact the external master server every 60 seconds for command updates. Appears as normal internet traffic as if someone is browsing the internet

Defenses

- Hidden Files
  - File integrity check
  - Internal Intrusion Detection System (IDS)
- Covert Channels
  - Anti-virus; IDS; Up-to-date and patched systems
  - No effective defense once an attacker has access
  - Manually removed if found

Covert Channels: The best defense is to keep attackers out to begin with! Up-to-date Anti-virus software, IDS, keep systems patched...all that we have talked about previously. Once an attacker has access there is no real defense against covert channels.

In depth analysis with secure systems and established covert channel analysis strategies is the only way to locate covert channels and they have to manually be removed.

Use of Relays to Conceal Source Location

- An attack from A to B can be routed through outside machines called relays
- Indirect attack is harder to trace than direct attack
- Additional complications arise when path relays are placed within different jurisdictions

Netcat Relay

- Uses the network utility network to relay traffic
- On relay machine, runs the command:
  mknod buffer p
  - This command creates a file to act as a buffer
- Next, on the same machine, run:
  nc -l -p [recv_port] 0<buffer | nc [dest_ip] [send_port] 1>buffer
  - Which receives traffic on the first ports and forwards it to the destination IP

Simple approach to making a relay

Mknod buffer p

Creates a special file, the option “p” says that the file should be a queue (first in first out)

Nc -l -p [recv_port] 0<buffer

Receives data from a remote IP on the specified port, writes it to the special file buffer

Nc [dest_ip] [send_port] 1>buffer

Sends data from the buffer file to the specified IP on the specified port

Conclusions

- Pen testing requires adherence a code of ethics
- Pen testing can help find holes in your system before attackers can exploit them
- Pen testing is divided into five phases:
  1. Reconnaissance
  2. Scanning the Target
  3. Exploits and Compromises
  4. Maintaining Control
  5. Removing Evidence

References and Links

- CounterHack Reloaded by Ed Skoudis
- Nmap - http://nmap.org/
- Firewalk - http://packetstormsecurity.com/UNIX/audit/firewalk/
- Nessus Vulnerability Scanner - http://www.tenable.com/products/nessus
- (ISC)² Code Of Ethics - https://www.isc2.org/ethics/default.aspx
- EC-Council - http://www.eccouncil.org
- Open Web Application Security Project (OWASP) - https://www.owasp.org

License