Global Information Assurance Certification Paper

SANS Practical Assignment Version 2.0
GCIH Certification in
Advanced Incident Handling & Hacker Exploits

Wireless Vulnerability: ARP Poisoning

Shawn P. Duffy, CISSP
February 13, 2002

Table Of Contents

INTRODUCTION

ARP poisoning is not new to the hacker's cookbook of vulnerabilities. It became rather ineffective because it was confined to the local area network (LAN). Now, with the onset of wireless access, ARP has a renewed interest as we will see. But, is wireless a secure media? We already know that one security attempt with wireless, called WEP – Wired Equivalent Protocol, is not secure and that more and more hackers are able to crack it in a very short span of time. What you may not be aware of is how a once internal only protocol can now be exploited externally from your network. These ARP vulnerabilities will not affect only those currently using the wireless LAN, but also those connected to the local wired LAN.

It is possible for ARP poisoning techniques to go unnoticed. Even more disturbing, there are free and readily available applications that can be downloaded which allow anyone to attempt these vulnerabilities and maintain their anonymity.

Unless precautions are put in place, everyone with a wireless connection will be at risk. The information gathered here will address the preceding topics as well as how the ARP poisoning exploit is executed in a wireless environment and the type of damage that it could cause. Finally, it will address how an incident of this nature was handled and the lessons learned.

THE EXPLOIT

NAME

ARP poisoning in a wireless environment can produce either Denial of Service (DoS) or Session Hijacking. The MITRE hosted Common Vulnerabilities and Exposures (CVE) list has not submitted a specific CANDIDATE yet for this vulnerability, but there is a similar candidate under review: CAN-1999-0667. The ARP protocol can allow any host to spoof ARP replies and poison the ARP cache.

For a better understanding of the ARP protocol we must first look at the OSI model and the different layers it contains. The discussion of how the ARP protocol works becomes clearer with an understanding of how it relates to the rest of the TCP/IP operations.

The Open Standards Interconnection (OSI) reference model has seven layers. Each layer has independent functions that use the layers below and are necessary for the layers above in order to communicate. The International Standards Organization (ISO) created this framework for defining the communication process from the user through the network to the user on the

other side. It has become the standard from which all other models in network communication are derived.

The first layer, or the Physical layer, is the most basic of all levels. It specifies the electrical and physical connections between devices. This layer specifies rules for how the electrical current gets to the other side of two communicating devices. Using a binary example, it is how the ones and zeroes are differentiated.

The Data Link Layer (layer 2) is the next lowest layer, which defines several properties. The most important is the actual device address or unique identifier of each host. Without having a way to specify this particular piece of information, every host would have to determine, individually, if any set of data sent on the network was for them or not. This address scheme is called the Media Access Control (MAC) address in an Ethernet environment. This layer forms the first set of frames for sending data in a structured format on the physical medium. Because the bits now have a structured form, synchronization of the communication media is also required and fulfilled at this layer. Synchronization allows the starting and ending points of each frame to be defined on both sides.

The Network layer (layer 3) is where the logical connections are formed. This allows traffic to flow between devices over different available data paths. This layer is associated with the passing of information across networks. Some key features associated with this layer are logical addressing (IP addresses in TCP/IP), routing, switching, sequencing, and flow control procedures. Hosts that are not directly connected to one another can now communicate over many different paths.

The Transport layer (layer 4) is used to help make sure of the delivery of information across the network. This layer is where the TCP part of TCP/IP is located. The Transport Control Protocol (TCP) is a connection-oriented protocol whose job is to guarantee that each packet makes it to the other side and that each side reassembles each packet in the right order. Some protocols at this layer are not connection-oriented, adding speed to the connection as an offset to the guaranteed delivery. Without the guarantee, the packets may have to be transmitted more than once - the price for faster service.

The Session layer (layer 5) is responsible for organizing communication between devices in a network. Message flow control, dialogue control and end-to-end data control are many of the services rendered at this layer. The session layer and above, with regards to the TCP/IP stack, are incorporated into a single layer called the Application layer. Therefore, there are no specific protocol examples in the OSI model for TCP/IP above this layer.

The Presentation layer (layer 6) is responsible for the way the data looks to the end user. The Applications layer is dependant on this layer to provide proper

formatting and syntax to the each receiving application. For example, if we send data using ASCII formatting we want to make sure the data on the other side is converted back to ASCII and not HTML. This way the information looks the way it is supposed to for each application to use. Data encryption may also occur at this level.

The Application layer (layer 7) is the top layer that is responsible for the interaction between the user's applications and the lower layers. For example, file transfers, telnet, and mail transport such as POP3 and SMTP can be system calls created by applications such as web browsers or mail programs. This way each host can use similar programs to achieve the same result regardless of the specific user's application.

From IEEE's Request for Comment (RFC) 826, we know that the ARP protocol resides at the Data Link layer of the OSI model. The ARP frame is a datagram composed of a special header, which contains a TYPE field. The ARP protocol is used to convert logical address (IP addresses in TCP/IP) to physical addresses (MAC addresses on Ethernet) and back again. This is necessary because the hosts may not be on the same physical media (requiring routing). When the hosts are on the same physical media, the physical addresses are required. Logical addressing is used to route data from one network to another until it reaches its final network destination.

There are three different TYPE values that an ARP frame can have. TYPE 1 is a request for a physical address. The IP address is known at this point and we want to know the exact location to send the data. TYPE 2 is a reply to the TYPE 1. TYPE 3 is a reverse ARP request where we know the MAC and want to gain the IP address for routing purposes.

From a local network perspective, what happens is a TYPE 1 ARP is sent requesting the MAC address of a known IP host address. This packet is sent to everyone in the broadcast domain and is analyzed by each host to see if they are assigned the specific logical (network layer) address. In this way, the host with the matching network layer address can reply by sending its MAC address via a TYPE 2 reply. RFC 826 can provide more information on the process of switching logical addresses to unique identifiers (physical addresses).

Since making a request every time a host wants to send traffic is a huge waste of resources, the RFC includes a concept of caching each reply for a given amount of time. This ARP cache table resides on every host that makes a request. The information stored in the table consists of the sender protocol (IP) address, hardware (MAC) address, and protocol type: which is either static (manually entered) or dynamic (learned from a TYPE 2 reply).

Because not every host needs to know about every other host on the network, it is much easier for each machine to be concerned with *only* the host(s) it is

communicating with. Therefore, the tables are generated based solely on the TYPE 2 replies received from the network. Because the ARP protocol was designed on the assumption that every communication is bi-directional, it assumes a request was sent out whenever it receives a reply. This reply is coined the gratuitous reply when a request did not go out for it. Because anyone can render any type of ARP reply, it may also be considered a flaw in the protocol. This flaw allows exploits such as ARP spoofing, flooding, and generalized redirection of traffic through bad (termed "poisoned") ARP information.

OPERATING SYSTEM

This exploit falls under network traffic and is not OS specific. The vulnerability is in the physical address management scheme of the TCP/IP stack. This scheme is found in the Data Link layer of the OSI model. More precisely, the exploit is in the way ARP gratuitous acceptance works. Bridges, switches, and any other layer 2 host or device that hold a cached ARP table are vulnerable without proper configuration.

BRIEF DESCRIPTION OF THE VULNERABILITY

Returning to the wireless environment, the attacker starts by monitoring network traffic with any wireless sniffer. In our exploit we will use WildPacket's AiroPeek™. The attacker gathers all the ARP replies and requests that are sent out in addition to the communication between each host. This is accomplished through simple passive scanning techniques based on IP or MAC address matching.

The attacker then decides on a victim to pursue and then alters the ARP cache. He will need to poison both sides to ensure he gathers traffic in both directions. The victim's machine is obviously one side, but the other side is not necessarily the opposing host. In most cases, it will be a switch the victim is communicating with which is then forwarding the data on to the other host. Another way to explain it is the attacker makes the victim think he is still communicating to the same entity, but in reality the victim is sending the traffic to the attacker's machine. The attacker is now in the middle of the communication and now has the ability to intercept both incoming and outgoing traffic. This intercepting of the traffic from both sides is referred to as the Man-in-the-Middle (MiM) attack. From this strategic monitoring position, the attacker now has the ability to alter the data between the two entities, kill the connection in progress, or hijack the session altogether.

VARIANTS

Wired Media

The ARP poisoning exploit originated in a wired media environment. However, the one limitation is that it was only possible from within the local environment.

Layer 2 devices can bridge networks that belong to, but separate from, the same system. Layer 3 devices, on the other hand, are used when corporations want to conglomerate many local networks into a dispersed single LAN configuration. Routers work with layer 3 addresses. Bridges, and other layer 2 devices, do not use these addresses and view this information as a part of the data portion of the frame packet. Therefore, physical (layer 2) addresses are not necessary outside of the local area network. The routing (layer 3) addresses are built through static or learned entries and *only* have to do with what devices the data actually travels through.

It should be clear now that it is no longer necessary to bridge information to the other side of an enterprise network. This was great news since the lower media layers do not intercept or modify routes. The physical addresses are now contained inside the router domain and therefore, layer 2 functionality is not exposed outside your local network. Simply put, this means the ARP exploits are no longer an external threat.

However, with the introduction of wireless Access Points (APs), this protected level becomes altered. Wireless APs are hubs, which are simple repeaters of data. By allowing this lower layer device to exist as an access point to your network, you introduce the attributes and exploits of this layer to the external side of your network. These protocols are no longer contained within the network perimeter and are now used to dictate external functions.

ARP protocol

We have already described the variant in the ARP protocol itself. Gratuitous acceptance simply means a host will accept an ARP reply without actually requesting one. This happens when the switch sends out updates to connected hosts or when some host device makes changes they wanted everyone else to know about. An example would be when a Windows® systems is coming online and wants to make sure no one else is using the IP address assigned to it.

However, another variant in of an ARP exploit is that it can cause flooding or broadcast storms that lock up or revert switches to hub-like functioning. The attacker could also provide a twist to a Denial of Service (DoS) attack. What if every host sends their information to the wrong destinations? Traffic would be sent everywhere but where it was intended.

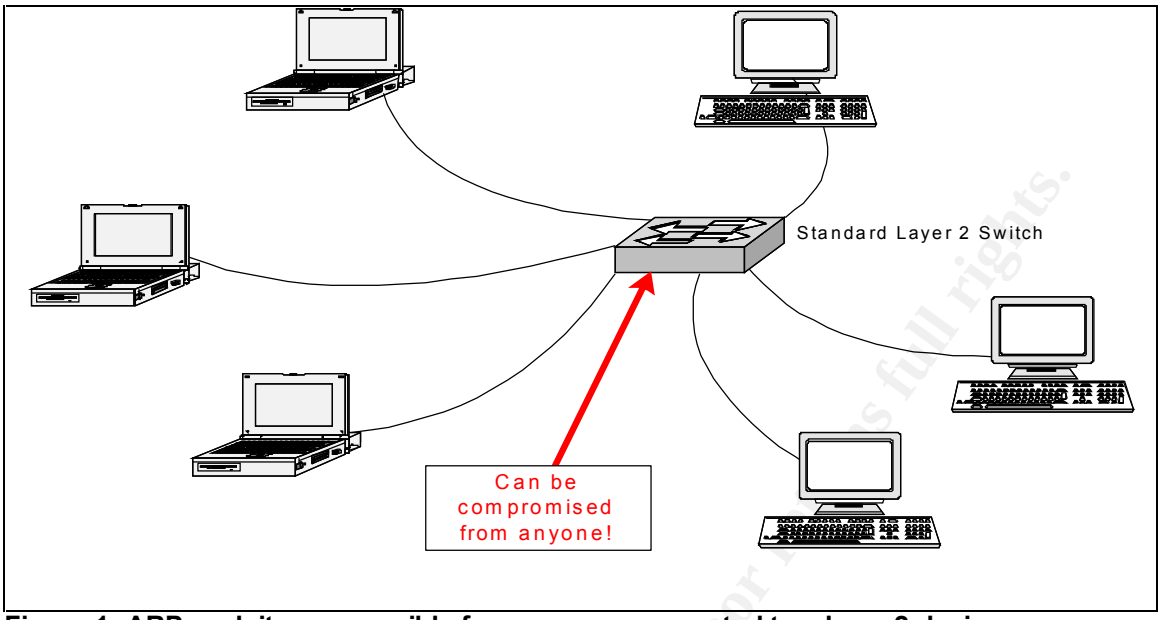

Figure 1: ARP exploits are possible from anyone connected to a layer 2 device

ARP Flooding: DoS

Besides the Man-in-the-Middle (MiM) attack, ARP poisoning can cause other problems. The switch can be altered with ARP flooding (refer to **Figure 1**). This technique involves sending a massive amount of random address entries to the switch. The switch will either crash or become no more productive than a hub. If the switch is stable enough to survive the flooding attack, the attacker may now have the ability to sniff all the ports on the switch. Before the attack, the only port the attacker could see was the one they were attached to. In a hub-like state, not only does the switch give access to other portions of your network to the attacker, it also defeats the primary function of the switch to separate collision domains. Worst of all, if your switch is unable to handle a flood attack, it may simply crash. The Denial of Service (DoS) is effective regardless of which way your switch handles ARP flooding.

Wireless Access

Variants in the wireless access may include WEP encryption, Wireless Transport Layer Security (WTLS), or Wireless Authentication Protocol (WAP) to slow the attacker's success rate. Each of these security measures could also deter the success rate of any attack, not just ARP poisoning techniques.

Different wireless protocols could also be a preventative measure. Some other protocols to consider are IEEE's 802.11a (57Mbs) and the personal area network (PAN) technology of Bluetooth (721Kbs). However, there may be different concerns in these alternate choices. Regardless, these protocols are outside the scope of this paper. The official websites for more information are IEEE's at http://www.ieee.org and Bluetooth's at www.bluetooth.com. Be mindful that these also function at layer 2.

SOFTWARE CODE

The following three URLs will link you with freeware tools to accomplish or deter ARP poisoning in either DoS or Session Hijacking scenarios:

http://www.monkey.org/~dugsong/dsniff/
Dsniff is a sniffer tool by default, but it will allow you to ARP spoof and provides other tools to help explain the vulnerabilities of ARP. It also has methods for hijacking TCP connections.

http://ettercap.sourceforge.net/
The ettercap tool allows a multitude of sniffing methods, injecting of characters in a data stream, OS Fingerprinting, as well as the MiM attack. In addition to the above, there are new features within ettercap that allow hijacking of SSL (Secure Socket Layer) and SSH (Secure Shell) connections. This is the software utilized in the attack for this paper.

http://letanou.linuxfr.org/arpwatch/arpwatch.html
ARPWatch provides a means for verifying changes in the ARP table by confirming the standard ARP table with a cached one. This is a French page, so you may need to translate.

Please refer to the REFERENCE pages at the end of this paper for a variety of links that will give you additional ARP and wireless information.

THE ATTACK

DESCRIPTION AND DIAGRAM OF THE ENVIRONMENT

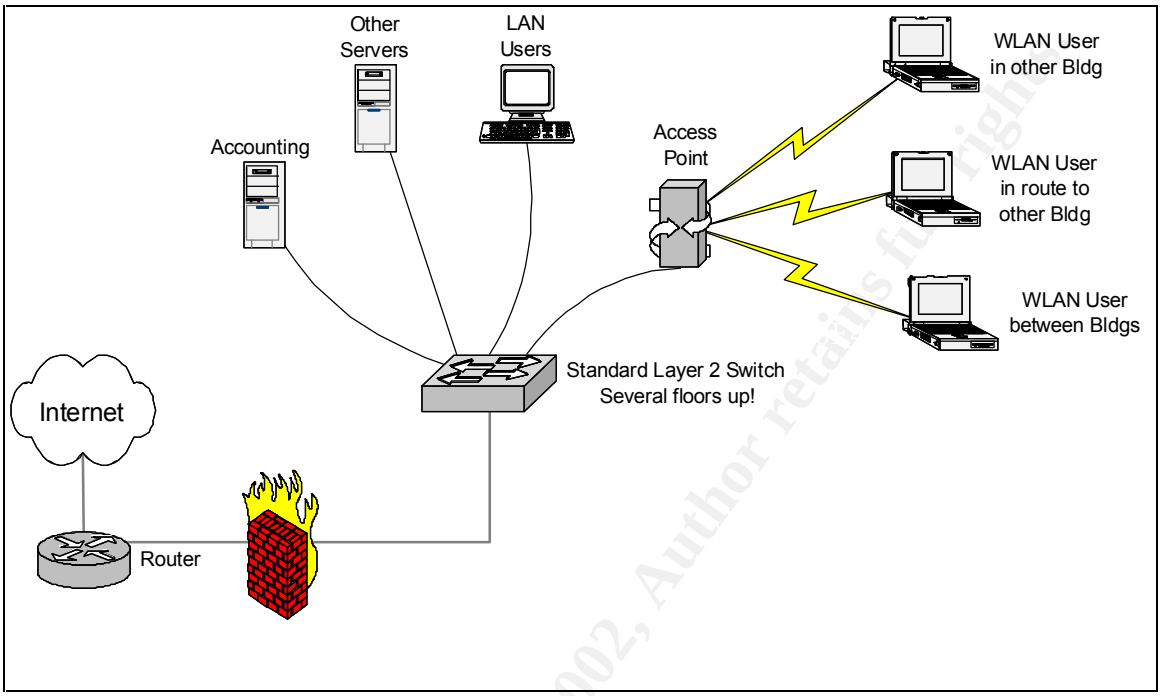

Figure 2: The Environment

Here we have, in **Figure 2**, the typical small network infrastructure. We have a Cisco 2600 router running IOS 12.0 without any access control lists (ACLs). A StoneSoft StoneGate 1.5 firewall is protecting the organization from the Internet. A demilitarized zone (DMZ - sometimes also referred to Data Management Zones) could have been utilized for their wireless access point, but because this wireless solution is between two building and the firewall is in the basement, the systems administrator decided to place the D-Link wireless access point (AP) close to a window and high enough off the ground to get a good transmission to the other building. The other side consists of strictly remote users with Intel and D-Link wireless cards in their laptops and computers. The computers used the same wireless cards by installing a special PCI cards containing a slot designed to hold the wireless PCMCIA card.

The wireless access point is connected to the network with an old 3-Com layer 2 switch. The other devices consist of a few Windows NT machines running software such as MS Exchange and some old accounting software, and Windows 98 or 2000 clients. No security auditing is enabled on the servers or clients and it appears that no updates above Service Pak 4 were installed on the NT machines. Service Pak 1 was installed on some 2000 clients while others remained in the original installed configuration.

NetBIOS is running over TCP/IP without filtering from either the router or servers. Everyone logs into the domain having the same name as the company. All the shares are created on an as-needed basis. Though most people do not use secure email, some use PGP for encryption of emails while others use standard X.509 certificates. PGP is free at www.pgpi.org and standard X.509 certificates can be obtained free of charge from www.thawte.com.

The employees can use the network from either the main location as well as the other building. They can even bring their laptops to lunch and work from a Deli in a completely different building. And, of course, they can stay connected when traveling between the buildings.

With security in mind, the administrator tried using WEP encryption. But because they could not get the WEP configurations to work correctly, it was decided to get the access point (AP) running first and figure out the encryption part later when there was more time. Any idea how long it takes to make more time? We can therefore observe the wireless users accessing the AP and entering the network without any security.

Behind the firewall is a switch to separate collision domains and allow for a more fluid flow of traffic in and out of the local network. This also separates the AP from other devices not communicating with the wireless users. Finally, we have the servers and users whom are accessing the system and the Internet.

HOW THE EXPLOIT WORKS

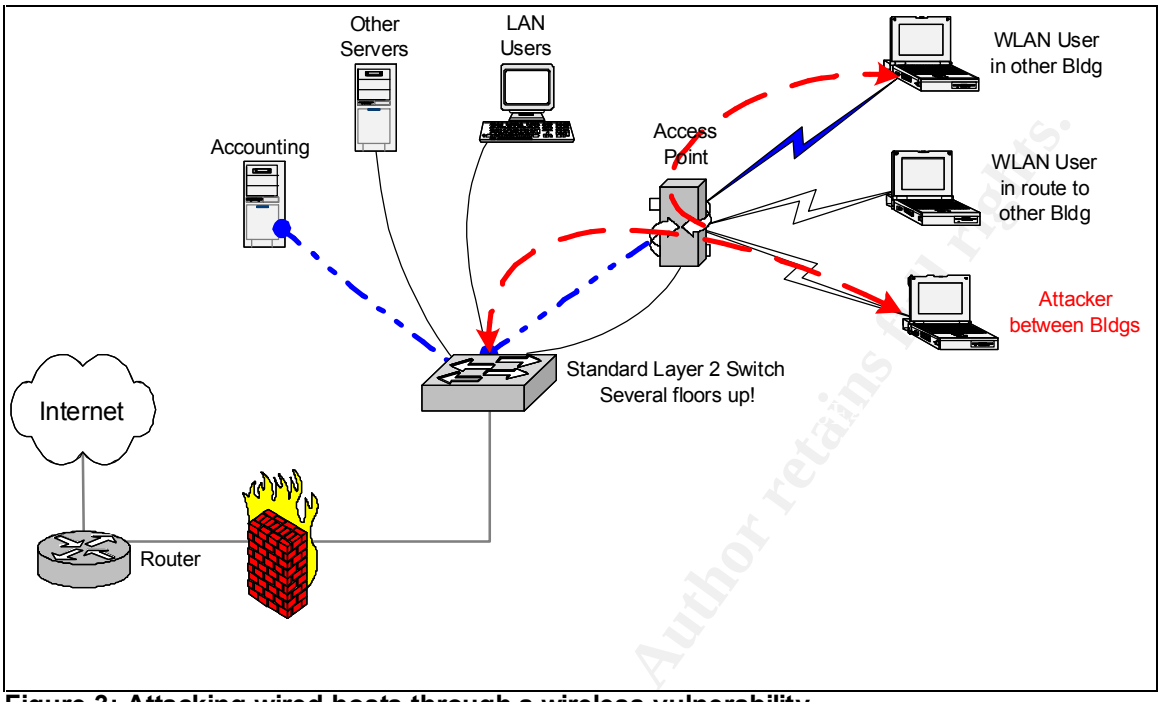

Figure 3: Attacking wired hosts through a wireless vulnerability.

By looking at **Figure 3** above, we see in the blue line that a wireless LAN (WLAN) user is accessing the Accounting server. Because the AP is hub-like in nature, all the traffic is broadcasted to everyone on the wireless media. It is easy for the WLAN attacker to see what is going on, especially without any encrypted traffic. The WEP encryption that was chosen earlier, but not implemented, would have complicated the monitoring of the traffic, but only slightly. We have already briefly covered how weak WEP encryption is and how fast it can be broken. You can refer to http://rr.sans.org/wireless/equiv.php for further information on WEP vulnerabilities.

The red lines indicate that the WLAN attacker has poisoned the ARP cache of the WLAN user and that of the switch behind the wireless AP. It may be necessary to flood the victim's host with garbage or bogus requests to keep it occupied while the actual MiM attack takes place. Note, there are no timeout concerns as long as there is frequent data from the connection believed to be correct. The victim's poisoned cache, as with all ARP listings, will eventually timeout without activity. Until a timeout occurs, the ARP cache will continue to use the last one it received. After the timeout, the cache removes the entry and will request a new one when it needs to communicate again. If there is no device or methodology in place to alert for ARP changes, the attacker may go undiscovered indefinitely.

Once the attacker has each side believing the altered ARP entries are correct, both sides will forward traffic to the WLAN attacker. Remember that the IP addresses do not matter on a local area network only the MAC address.

From here, the attacker can simply monitor the connection for username/password combinations, inject his own traffic (the MiM attack), or just kill the session and move on to others.

One stipulation to a successful attack is that the switch does not contain a list of static MAC entries in its port configuration. This is unlikely, but possible, that all host MAC addresses are entered statically in the port table of the switch. Static port entries would not typically be implemented for individual users due to the maintenance required to continually monitor static configurations. This is a good security solution but can be a nightmare to administer.

By using a freeware program called ettercap, we can establish a connection relatively easily with limited knowledge of the ARP protocol.

The set ettercap commands used for the attack are as follows:

```
# ettercap –NCsz                    [captures username/password combinations- highlighted below]

ettercap 0.6.3.1 © 2001 AloR & NaGA

Your IP: 192.168.0.70 with MAC: 00:03:FF:BE:F0:52: eth0
Loading plugins… Done.

Resolving 1 hostnames…

Press ‘h’ for help…
 Sniffing (IP based): ANY:0 < -- > ANY:0

 TCP packets only…  (Default)

Collecting passwords…

00:22:10 192.168.0.70:1107  < -- > 192.168.0.42:80      www

USER: root
PASS: hamhocks4#age

http://mail.victim.com/root.asp          [the site where username and password was entered]
```

And

```
# ettercap –Nf 192.168.0.70          [will display the operating system of the victim]

ettercap 0.6.3.1 © 2001 AloR & NaGA
```

```
Your IP: 192.168.0.70 with MAC: 00:03:FF:BE:F0:52 eth0
Resolving 1 hostnames…

Fingerprinting . . .

Operating System:

Linux 2.2.16-22

Network Adapter:

Intel Corporation

#_
```

Or

If you want a GUI / ncurses interface just type:

***ettercap*** **[**no switches will run the GUI interface**]**

After you run this command, you will be given a screen from which to choose your target(s). From there, it is simply a matter of deciding what to do and who to do it to (see **Figure 4a**). Ettercap also was nice enough to include a "–h" (*help*) option for every screen. The help screen is shown in **Figure 4b**.

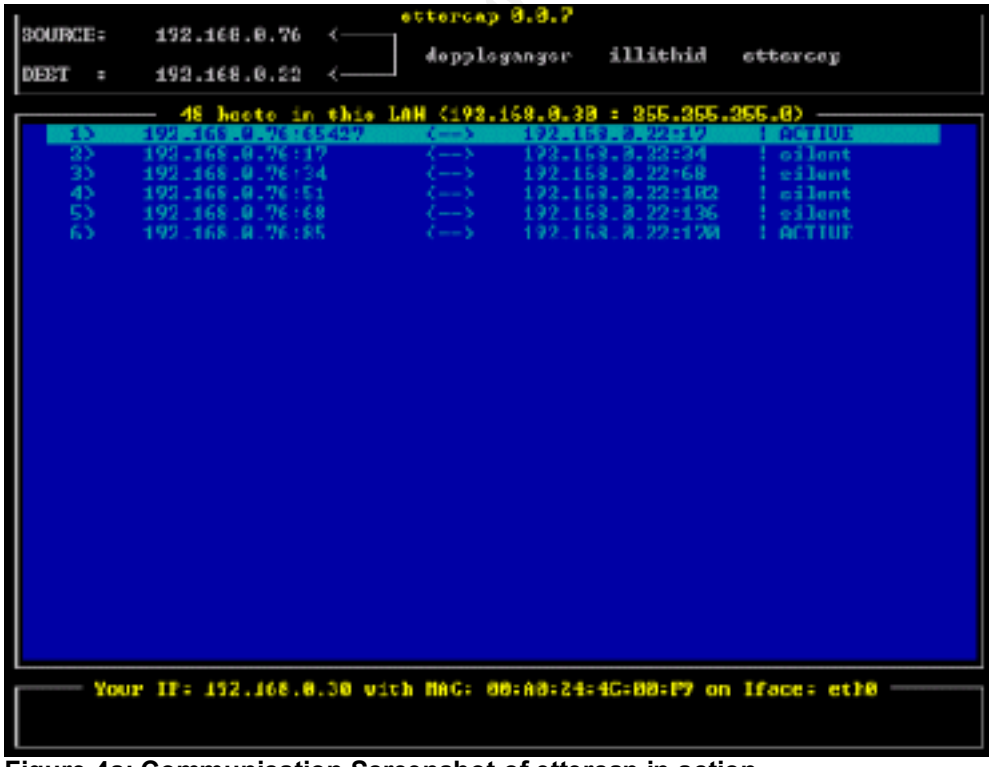

Figure 4a: Communication Screenshot of ettercap in action.

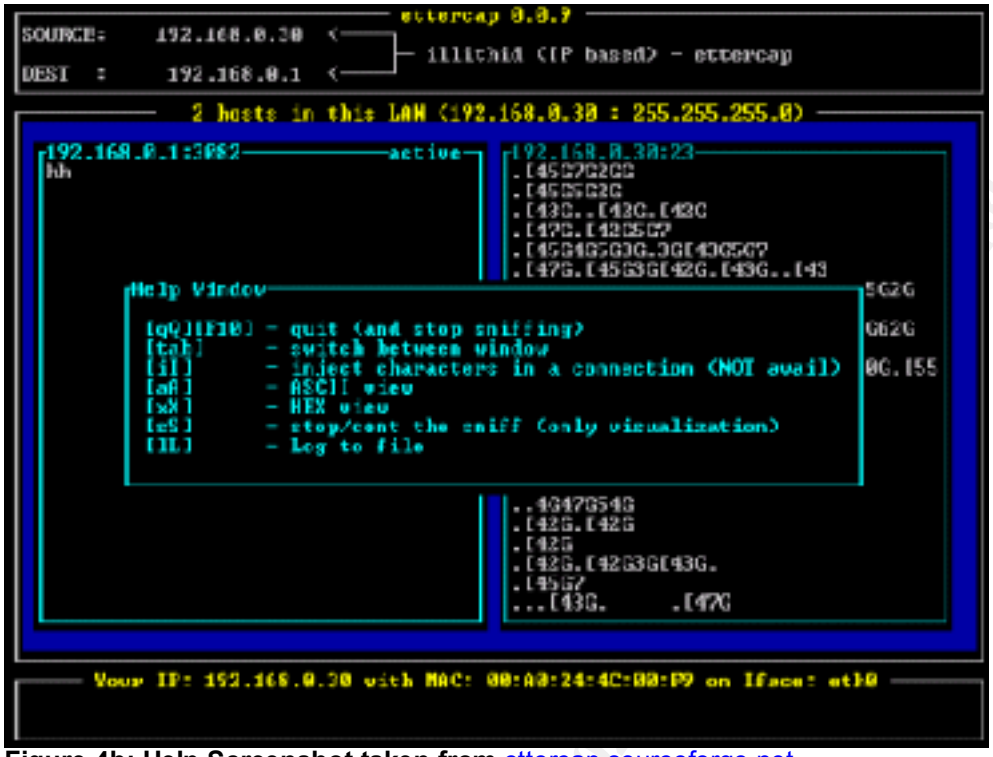

Figure 4b: Help Screenshot taken from ettercap.sourceforge.net**.**

Once the user has access to someone's account (e.g. the root account from the first ettercap command above) the next stage of the attack can begin. The attacker now has the ability to scan the network from the inside. It is much easier to find more valuable targets to exploit from this vantage point.

For example, here the attacker has hijacked a connection between the wireless user and the server the user was talking to. Now, it just so happens that the Accounting server is a prime target on this network and worth a thorough look through. And, because the attacker is now working from the inside, they have a much better chance at compromising many more systems than simply the one used to get in. This is why it is so important to contain an incident as soon as possible. Knowing which ones to investigate will save hours of work.

The attacker will most likely load valuable tools from the first machine compromised. This host will act as the launch pad for additional explorations and exploitations. The attacker may hide his software by creating directories that are hidden, or using ones deep in the file structure, or in unsuspecting areas such as \TEMP\ or \WINNT\SYSTEM32\RECYCLE\.

Once the attacker has the tools needed to continue, the next phase is scanning internal network. The attacker may look for Intrusion Detection Systems (IDS), or traffic trends, or other means by which to understand you networking habits and users routines. This information reveals valuable information to the attacker in

regards to when and how to maneuver through the network. Eluding the network administrator this way may foster neglect on the administrator's limited resources during peak working hours. Capabilities used in daily activities may miss suspected or tip-off signatures across the infrastructure.

If the attacker does go unnoticed, they may find other devices besides servers and workstations in which to do their dirty work. They may find your PBX and start making long distant calls to their buddies. Perhaps, they come across a user's brand new Ultra-Mega-Unix station with all the latest greatest tools and gadgets. This particular user may have decided not to have a strong password on this station because he knows no one in the office knows how to use it. Even more frightening, what if the hacker finds the server that holds the company's database of client credit cards or address records? The legalities of the type of damage this could cause would be endless.

SIGNATURE OF THE ATTACK

Two things that are pretty good tips that something is wrong are invalid SSL certificates and dropped connections.

An invalid SSL certificate will identify itself to the user through a browser error. But, unless the user is aware that this is not normal, they may just click *accept* and continue to use the invalid certificate. The certificate comes from a MiM attack when the victim has requested an SSL connection. The attacker needs to supply a certificate back for the victim's request. If the attacker used the certificate of the real web site, then the encryption created between them would not allow the attacker to see the conversation. So, the invalid certificate is supplied by the attacker in hopes the victim will just accept and move on.

Dropped connections are simply viewed as a timeout on the user's part and many times (possibly always) the user simply tries the connection again. If they are successful on the next attempt, they generally forget about the first connection drop. If it happens often enough to frustrate the user, then there is a good chance you are going to hear something from them.

However, since neither of these are definite indications, it is highly unlikely you will be notified of such an incident. If you demand your users notify you every time something happens, you will most likely be inundated with false incidents. However, all is not lost. We will look into what administrators can do to help with proper reporting habits.

Today's intrusion detection systems (IDS) are more in tune with changes that occurred with the ARP protocol. IDS systems can be found in personal firewall software as well as network appliances or as software for servers.

Some typical results from an IDS log regarding ARP might be a duplicate IP address alert, a LanD attack, IP Spoofing, or, if you are really lucky, an ARP attack itself!

Duplicate IP address alerts result from the ARP table showing one MAC address associated with a host and then receiving a reply from the network with a different address. Every host that transmits to another device maintains an ARP listing of the receiving device. This makes transmitting data to the same person much faster. For clarification, if you already know the MAC address of a local host then having the IP address of the same host is really unnecessary (you already know where it is).

So why not maintain a listing of everyone? The answer is you do not need to know where everyone is. You are more than likely only talking to a few of them anyway. This physical address listing supplies the specific locations of only the ones you are communicating with the most. Having this stored (cached) listing also prevents having to "look up" where each host is for every transmission sent. Remember that the logical address (layer 3) only acts as a domain location. The physical MAC address is required for every host-to-host communication.

IP Spoofing is when a host appears to be another host by using an IP address that does not belong to him. The reason someone would want to do this is simple. They do not want to get caught doing something they are not supposed to be doing. It works because the source IP address of any spoofed alert is bogus. On a side note, an ARP spoof attack will look like an IP spoof attack to an IDS system who is only monitoring changes at the layer 3.

LanD (LAN Daemon) attack is when someone has sends you bogus or forged packet using your own address! This is a clever Denial of Service (DoS) attack where the attacker may be on the same subnet, wants to remain anonymous, and simply wants to slow your system down. The trick in the attack is to make your system think it must respond to itself on the same port it sent the request. What happens in a Windows® environment is the system slows down while it repeatedly tries to reply to itself. An ARP poison attack may be configured to use LanD attacks in order to block the original host. If the victim were running a personal firewall at the time, it would show a LanD alert.

HOW TO PROTECT AGAINST WIRELESS ARP POISONING

The first thing to recognize is that we are no longer dealing with layer 3 at the perimeter. When wireless media was introduced to the perimeter environment, so were new protocols that required attention from the integrator. In addition, many older protocols were also made available again. This could obviously have been missed in the design of many security implementations.

Wireless Placement with Regard to the Firewall

You should place the wireless device outside the firewall for very good reasons. The primary reason is the security holes produced from a wireless "hub" being a perimeter access point inside your network.

> *"An 802.11 wireless access point should be placed outside the firewall in a network design. In this way, the normal and ordinary protection afforded by the firewall is extended to wireless access as well as wired communication. Placing an access point inside the network firewall defeats all the protection offered by the firewall and, in essence, opens the front door of the company or organization to anyone who has an interest in using the resources of the network from the street or parking lot. Wireless networks are not inherently a security loophole. They do, however, demand careful network design and appropriate security considerations to avoid creating a point of security exposure."*
> **-Joe Bardwell, Vice President of Professional Services, WildPackets, Inc.**

WildPackets is the maker of a wireless sniffer called AiroPeek™. According to the WildPackets' white paper, Guide to Wireless LAN Analysis, "Packet analyzers cannot detect eavesdroppers". Therefore, it is imperative that you proactively check you network for new APs. For any existing wireless device, you should also check the signal strength and diameter or footprint of the existing signal availability to limit access. WildPacket's AiroPeek is one of just a few Wireless Sniffers available today that can accomplish this task.

Encryption

Also, using WEP may seem a mute point, but the fact is that *any* encryption is better than none. It has also been recommended to use Virtual Private Networking (VPN) tunneling technologies on your wireless access points. IP Security (IPSec) VPNs encrypt and decrypt at the network layer and therefore are not vulnerable at the lower layers where they appear as encrypted data. However, there exists some layer 2 VPNs that can cause problems in your implementation. Point-to-Point Tunneling Protocol (PPTP) and Layer 2 Tunneling Protocol (L2TP) are two examples of layer 2 VPNs. The problem with layer 2 VPNs is they get in the way of a unified transmission medium. PPTP is proprietary and revolves around only using Microsoft services. L2TP is a combined effort of Cisco's Layer 2 Folding (L2F) VPN and Microsoft's PPTP. IPSec operates at layer 3, is not proprietary, and is the de facto standard for VPNs.

Detection

Look at getting a good network IDS system. Many of them will detect ARP spoofing and poisoning. A host based personal firewall will certainly detect something amiss if your ARP table is suddenly altered without requests. BlackICE from ISS is a good example of one that works. (See **Figure 5**)

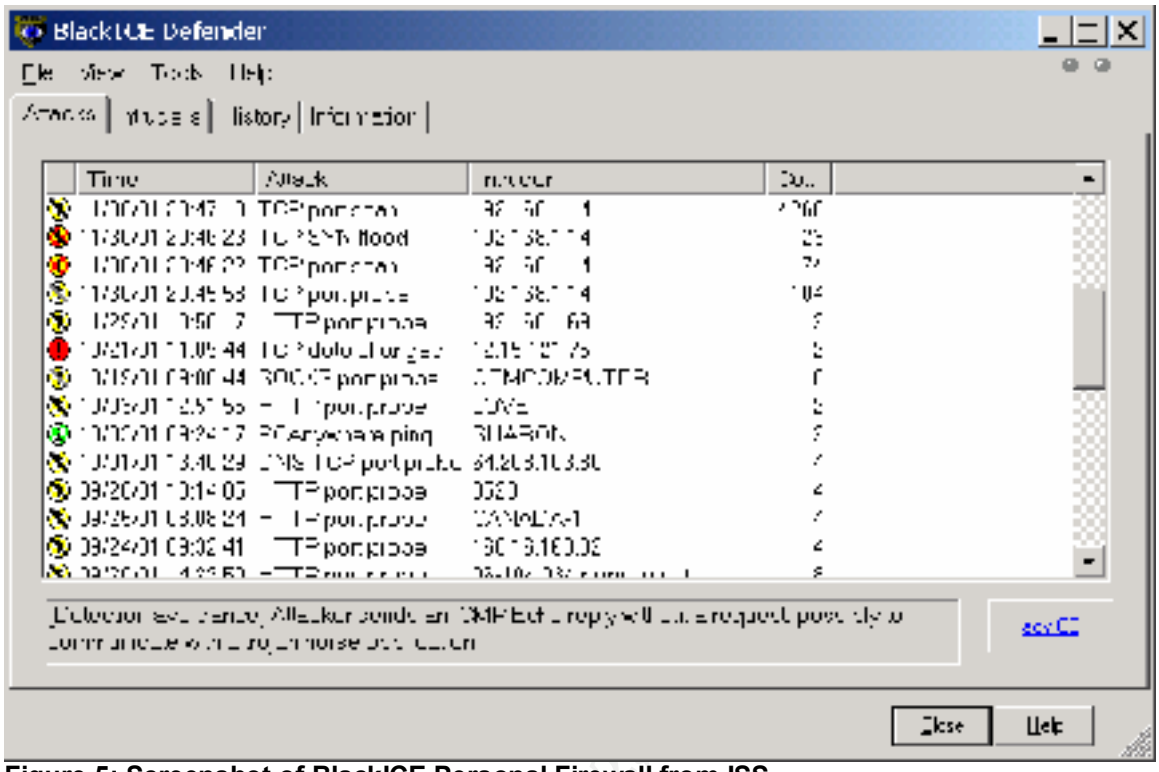

Figure 5: Screenshot of BlackICE Personal Firewall from ISS

According to Sean Whalen of GMX, "the only defense available is detection." He suggests ARPWatch as a solution because it correlates the table it builds with the cache ARP table of the host to check for inconsistencies. It also automatically alerts the administrator or other security personnel.

Finding the Right Wireless Product

Another defense mechanism is in carefully choosing a wireless product. Look for one whose manufacturer has spent time working on the security issues and limitations. Cisco's Aironet products have a technology called LEAP (Lightweight Extensible Authentication Protocol) that allows for mutual authentication and the encryption keys to be periodically rotated. This will provide for a more secure link with less data available per broken set of keys. Mutual authentication means both sides authenticate the connection. This prevents someone from pretending to be you because they cannot authenticate as you. According to Gail Meredith of Cisco's Packet Magazine, "Mutual authentication is the only way to prevent man-in-the-middle attacks." Rotating keys allows a smaller window of opportunity to gain information from a successful cracking of the encryption codes. So you can see, Cisco is taking an active role in deterring the vulnerabilities of wireless networking.

Static MAC entries

One final defense against ARP poisoning is static routes or the binding of MAC addresses directly on your switch or, if possible, the wireless AP. This is very

time consuming to both implement and maintain. It may also be impractical given the number of users and their mobility requirements. This concept is mentioned as a thought-provoking method to help defend you network.

THE INCIDENT HANDLING PROCESS

The above-described incident did not actually take place. In accordance with version 2.0 guidelines for a GIAC practical, the following are the steps that would be used to address the attack above if it had actually happened.

This organization is new to the security world and has little in place to stop attackers. They have a systems administrator and many managers and supervisors that consider security an overhead cost rather than a functional necessity for corporate business.

PHASE 1: PREPARATION

It was believed that this organization's chance of becoming a victim was very slim. There was little effort put in place to handle an incident until the Nimda worm was released in late September of last year.

When this organization became another casualty of Nimda, it took several days for them to contain the worm and return to normal working conditions. It was then decided that some type of security measure would be necessity to handle any attack in the future. The entire implemented solution consisted of:

- Regular backups for a quick restore
- The patch that fixed the related vulnerabilities in IIS
- Modification of a few firewall rules and the security policy
- Create and fill the new position of Information Security Officer (ISO)

There obviously should have been a little more preparation work. The intent was there, but unfortunately no action took place. Of course, this sets the company up as a prime candidate for a new attack, which happened to be ARP poisoning through their new wireless environment.

It would be up to the new ISO person along with the help of the systems administrator to know what to do in the future to make the company more secure.

The security policy that was already in place only covered computer and Internet use. The policy also dealt with virus and file downloads, but nothing that dealt with handling an incident or what the computer access rights were.

The policy's purpose was to establish the management direction along with procedural requirements to ensure the appropriate protection of automated

information. Regrettably, there was no statement on how to enforce this policy or who was responsible for its changes and implementation. This turned a would-be dynamic security policy into a shelved space-taking dust collector. To make matters worse, no one was even shown the policy prior to initial hiring nor did they receive any security awareness training to promote a good security practice.

The policy lacked some basic principles that could have help to prevent or to prosecute the attacker if they were caught. Some suggestions that could have had an impact might have been system login security banners, personal purpose Internet use, prohibit the testing of system controls, what information to include in logs and that those logs must support audit handling.

Login security banners offer a legally posed warning that you are not allowing anyone without legitimate reasons to access the network. There are stories out there that describe an individual getting away with hacking a network because there was nothing that told him he could not do the damage. The banner should consist of at least four items, which are:

1. Access is only for authorized use
2. Any access may be monitored
3. Unauthorized access is prohibited. (Very important)
4. Unauthorized access may be subject to both criminal and civil prosecution

Having a response team in place would have also been helpful. They were without primarily because management did not know that one was needed. Therefore, finding any available response team tools was limited and dispersed at best. For example, it can be rather hard to find a small Ethernet hub when you had not planned for one to be available.

Some of the more unsuspecting tools that should be on the list include a set of procedures of what to do when an incident occurs, a list of the people who should be contacted, documentation and reference materials on what each system has installed, and where to find each of the installation media.

A team lead should have been selected to run a response. Without a lead there is a good change that information will become dispersed and not all the information gets dispersed to everyone. Therefore, important information might be overlooked in a decision that affects everyone. The team lead should also have talked with the local law officials to get an understanding for when to get them involved. There should be a policy for how to conduct investigations before contacting officials and under what circumstances to contact them.

Not every case it prosecutable by criminal law, but civil law is certainly a viable alternative for punishing an attacker in court.

In our incident, the team consisted of the Information Security Officer, the facilities officer, the CIO's assistant, and the systems administrator. The systems administrator is normally always on the team because of his involvement with the functions of the network. His actions may be limited to the needs of important functions such as locating current backups, recovering logs, verifying network layout, etc. Because the organization did not already have a response team in place, scrambling to find additional people who could play the role of an actual handler was difficult given the nature of the problem and number of people who knew enough to be helpful.

PHASE 2: IDENTIFICATION

Identification of the wireless ARP attack was quite frankly, sheer dumb luck. The user whose session was being tampered with happened to mention one afternoon that he was having connection problems in one of the buildings. He stated he would often get disconnected. He would simply try to re-login and usually everything would be all right. He also stated that the connection was sometimes slow or intermittent, when it was working. It was later discovered that it only happened when he was accessing particular hosts within the system.

A few days after the user reported the issue, it was discovered that the servers being accessed showed multiple logins by the same users from different locations. This was a clear indication that something was wrong.

They were without logging and therefore had nothing to substantiate the incident occurring. The systems administrator was rebooting a box when he noticed the issue with multiple logins of similar accounts. The incident was not identified as a wireless problem until later.

The incident was never determined to have come from someone who was not in the building until the tip-off occurred. The tip-off happened one morning when someone who appeared to be already logged in had not even turned on his computer yet.

Remove from Network

Unfortunately, the last thing anyone wanted to do was to remove the wireless connection from service until they were absolutely sure that was the problem. It sounded, at first, as though this particular user was having a hardware problem or perhaps a software configuration problem. But now there was a serious indication from the tip-off that it was not even his system but an impersonator.

By now, management was also using this link so you can image the type of issues that stemmed from forcing discontinuity from between the two buildings. Fortunately, it turned out to be quite easy to convince them once it was understood the servers were being exposed to an intruder.

Eventually, after confirmation this was a wireless vulnerability, they decided to test the line by using the WildPackets' AiroPeek™ wireless sniffer.

Examine log files

The logs from the sniffer did not confirm their suspicions that the wireless connection was being compromised. However, it did show that the user's account had been breached. After observing the logs for login attempts they found one user trying to log in at the same time they were already logged in from a different MAC addresses. It was apparent that someone was online pretending to be someone they were not.

The team also observed login attempts to other services and hosts that the legitimate user did not have authorized access. Since no scans were detected, they started to contemplate the possibility that this could be an insider attack. If it was not, it could only be from an attacker with an abundant amount of learned knowledge from "listening" to network traffic.

Evidence Collection

Evidence collection was started right away. Since there was no policy as to what to do, the team decided to leave the sniffer running and prevented the legitimate user from accessing the wireless connection. From there, they could watch for the attacker attempt a connection and see where they went. It was necessary to maintain the link at this point to verify how far the incident had gone and what other compromises may have taken place. If the attacker had actually logged into a box, they would have killed the wireless link and prevented any additional mayhem. The team was unsuccessful in finding any attempts to regain access from the intruder. They believed that this meant someone or something had tipped-off the attacker and he went into hiding, obviously, to avoid detection.

The team also started a notepad of activities to keep track of what they were doing in order to have a record of all the details of this incident. They got the systems administrator to collaborate any logs that he may have already obtained during this time. However, it can be difficult to keep current records during an response situation, so someone had be to elected as a "notes taker" in order to make sure that something was written down periodically.

PHASE 3: CONTAINMENT

Removing the Access

When they were sure they were not going to get anything else regarding the incident, the team saw no reason to maintain the wireless link. The link was shut down between the buildings until they found a solution to fix the exploits. They went through all systems that they believed the attacker had attempted access looking for system abuse and changed passwords. Because they had no logging

information to refer to, they basically began looking at servers where the legitimate user could remember he had logged into. They had no audit trails to search and inspect changes from, so the user's recollection was all they had to work with.

Assess the Damage

During this phase the team turned to the auditing features in the servers by making them functional where possible. Some servers did not have enough resources to allocate to logging. In the others, the team configured a network IDS system and forward logging information to a separate syslog server.

Since this was a very specific access point it was easy to disable the connection. However, the compromised systems could be any one that was running at any time the attacker had access. The problem now was to figure which ones they were.

Backup a Pristine System

Sadly, there were no recent backups. Of those systems that where backed up, some were never tested and some had their most recent backups completed months ago.

Keeping the current system pristine only lasted during the new backup sessions. The rest of the time it was devoted to messing around with the configurations with trial and error on finding compromised systems, and trying to integrate logging and auditing tools.

Some of the configurations that were used to examine the servers and switch were audit tools such as Foundstone's NTLast, Windows 2000® log properties, Event Viewer settings, and perms.exe to see who has access to what and when.

The NTLast command has an –r switch (C:\NTLast.exe –r) that will show all the remote (-r) login attempts, but only if auditing has been activated and is configured to log this type of event. However, the problem with our fictitious organization was that auditing had not been activated and therefore, NTLast was useless from the start.

In learning that auditing was necessary but not implemented, the systems administrator began to correct this problem. Security logging in Windows 2000® is not enabled by default. So the first thing was to get it running by going to the *Local Security Settings* from *Control Panel/Administration Tools* and selecting *Local Policies* and then *Audit Policy*. From the list on the right choose to activate the particular audit template or import your own. Enable the auditing of each template you choose by right-clicking the template and choosing to audit successful and/or failures for each. You can tweak the parameters of how much

to log and other filters from the *Event Viewer* by selecting the properties of the *Security Log*.

Perms.exe is a file found in the Resource Kit of Windows NT® and 2000®. The file gives you the ability to check the user's access against multiple files. This enables us to see if our attacker has granted better access for our victim during the course of their intrusion. It was a simplistic way to also check for compromised systems should any of them show signs of non-privileged access rights.

The software used to back up the few systems that actually were is an old version of VERITAS' (previously from Arcadia) Backup Exec™ Program. The servers were backed up each night with a differential backup of changed files throughout the week and then a full backup every Monday night. This backup process will provide a more quickly recoverable system because there are only two backups required to recover. The two required are the last differential because it contains all the changes since the last full backup and the last full backup.

An incremental backup would back up only the changes that occurred since the last incremental backup. It is faster to complete each night, but to recover your system, you must use every one since the last full backup. This works because the archive bit is set on a differential backup and not cleared. The incremental backup resets the archive bit for every backup and so each one only backs up the files that changed since the previous incremental backup.

The problem with backup operations at this organization was simply a turnover of responsibility and a lack of dedication to the process. It must have seemed unnecessary or not high priority in the scheme of day-to-day business. The backup operation eventually became sporadic and then seemed to become important only when there was time to do them.

PHASE 4: ERADICATION

Determine Cause and Symptoms

Determining the cause of the incident became clear when it was understood what the wireless media was introducing to the network. Access became readily available to anyone around the area who knew about the wireless access. It was also very easy to understand why there were problems when there was no confidentiality on the data being transmitted.

Improve Defenses

Improving the defenses to prevent further incidents was clearly necessary. When the patches were installed on the web servers to prevent the last attack, no one took into account of other vulnerabilities that might exist. Simply

scanning your network for existing vulnerabilities could have help expose additional exploits. System scans, network scans and environmental scans including personnel and office layout, all could have helped discover weaknesses.

Removing the Cause

The AP was re-configured with 128-bit WEP encryption in a separate lab area and VPN was discussed and later implemented for both wired and wireless remote access.

Filters were added directly behind the AP to act as a first level firewall allowing only VPN tunnel traffic through. The filtering unit was also a layer 3 device, so it also prevented ARP poisoning of any wired hosts.

A Network Intrusion Detection System (N-IDS) was also implemented on a traffic management port on the switch to help ensure MAC address integrity throughout the network segment. It was also used for all the other evident reasons of an IDS system.

Vulnerability Analysis

Checking the system for further exploits was clearly a step that needed to be implemented. Time pressures and stress levels would have been needlessly spent if the system fell victim again days after it is put back into operations.

PHASE 5: RECOVERY

Restore from Backups

After it was clear there was no way to prove what changes had occurred, they were forced to complete full backups and hold each current image as evidence.

The only way to be absolutely sure that the system was clean was to go to each machine and format them for a clean install of a new operating system. Luckily there were not hundreds at this location and the few that were could be accomplished over a weekend.

They restored a fresh copy of the operating system (OS) using the existing backup software (VERITAS Backup Exec®) and restored only files that were part of the operation of each particular system. It was not pretty, but it was all the team could do at the time. Of course, this also meant going to the web or installation/upgrade media and locating the most current service packs and security patches for installation.

To be sure that they were actually talking to the correct website, the systems administrator used connections from a remote location where he was sure he had the correct site and was not being redirected elsewhere. These files were

then burned to a CD where they could be distributed between new systems on the network.

Along with the OS, patches, and applications install, auditing was also implemented and started before the new systems were put back in place. The audit policy would be implemented with a Host IDS (H-IDS) system as soon as an evaluation took place to determine the best product overall for each host in the infrastructure.

Validate the System

System validation was not implemented other than a check that the operations were working and that the processes did not fail during the restoring of the data files and operating system.

Monitoring of the system took place for the first day and a half looking for anything that was not working correctly.

PHASE 6: LESSONS LEARNED

There were a multitude of lessons to be learned. It would be a gallant effort for the systems administrator to gain respect from colleagues after losing trust in the security of the network. It was apparent that a number of things must change in order to be effective in handling future incidents.

Follow-up

The follow-up process has several areas to consider. First, there needs to be a report to show the people, places, time, and dates of all the events that took place during the life of the incident. The report should gather data from the moment an incident was determined. This will be key to helping your organization prosecute the offender. It will also help form suggestions and future details in the learning process of your next incident. There should be a consensus to what happened throughout the entire process. This consensus will be necessary if the incident goes to trial. The reason is that it is very easy for the defendant to show inconsistencies during the trial if the prosecution and your organization's representatives do not have their story straight.

Consensus needs to be legally binding. Therefore, if everyone does not agree, they should put their additional comments in the document and still agree to the final writing. Now everyone involved agrees with one document.

An executive summary, along with any recommended changes, should also be created and handed off to upper management for approval and for documentation of the systems that they are held accountable.

They should also look at each incident handling section again and evaluate them to develop a methodology of each process. This way, the reoccurrence of an incident will be handled in a much more organized way.

Preparation

In the first step, they learned they did not want to go through this the same way again. The team is developing an emergency action plan. The plan will enable them to have a method to follow and will allow them to deal with the next incident a little more effectively.

The plan will include actions to deal with network intrusions, theft, and business continuity. It will also be a dynamic plan because "things change", as we all know. They need a plan that will account for current information on people, checklists, and any updates in technology.

Second, their preparation during the last incident was not the best situation to be in. Perhaps you, as the reader, will take some advice; a little preparation goes a long way in dealing with an entity (like the attacker) that clearly has done their homework and is well prepared to work your system over.

Another suggestion would be to create a "Jump Pack" that includes things like a small hub, backup media and configuration software, camera and mini-disks for recording notes, and the documentation on equipment you are responsible for.

Having a good security policy will help you in the event you need to prosecute. A good security policy will also established due diligence methodologies to help your users/clients maintain secure computing practices in your organization.

Law enforcement policies and procedures should also be of concern. You need to know what the boundaries are and who is responsible for what. Law enforcement can be limited in their abilities to take on new cases. You need to become familiar with your state and local agencies and what their criteria are for getting involved. Your team lead should know what to have ready and who to contact.

Identification

Identification of the incident would have been much smoother had they had proper audits and logging methods in place. Audit controls and active monitoring processes would have spotted this incident much sooner. That would have undoubtedly deterred much of the consequential searching of which devices were misused and which were not.

Personal firewalls are additional help because they determine unauthorized or unwanted access to individual systems. Similar to personal firewalls, host-based IDS systems are formidable allies from a server's point of view.

Using file protection mechanisms like Tripwire (www.tripwire.com) can help alert you to changes in your file structure or contents. This is very helpful for concerns in web defacements. It also is good for detecting backdoors and other Trojans that alter existing files. A good example was described in the Incident Handling course at SANS when you type "*ls*" and Netscape® comes up.

Even better than alerts, the best line of defense is to actively monitor your network activities. Every day someone should be observing any ambiguities such as activities late at night or multiple failed login attempts or just anything out of the ordinary should be investigated for an answer. This does not mean everything is an incident, but it could save you time and heartache if you catch something sooner than later. StealthWatch™ is a good network IDS system that gathers ambiguities on your network. This network IDS system works through traffic profiling and normalizing your data flow. When something out of the ordinary is found passing over your infrastructure, the StealthWatch™ network IDS system alerts you. This can be configured for different levels of alerts as well as the way you are alerted (e.g. pager, email, email, etc.). What is nice about traffic profiling is that its basis is on your network habits not a signature of known attacks. This way, the StealthWatch™ appliance can determine new attacks even before a signature is created for them.

Containment

Keeping a system as pristine as possible is key to retrieving evidence as well as to gathering information to return the system back to a working state. You must look at what services have been accessed and what damage has been done. An assessment is much easier when you can see log entries on what has been accessed. Remember what you are doing is trying to contain the problem. If you do not know where to start, you are going to have to look at everything. For a small system, you may want to do that anyway. But for a large organization with many servers, it might be just impossible to go through all of them.

When an incident occurs, having a backup process will save days in the recovery phase. Backup systems are absolutely essential to a smooth recovery. Those of you that are not doing them, start! Those that are, please check them regularly. Without proper checkups on your system, you run the risk of having invalid or incomplete backups. It is also important to remember the backup software will show successful, even when the backup was incremental and you wanted complete.

In order to better contain the incident, there should be Service Level Agreements (SLAs) with your Internet Service Provider (ISP). You will want one that allows you to proactively protect your network with their help. This does not mean that they must drop everything and come to your rescue. It does mean, however, that they must be willing to take an active role in, for example, denying DoS attacks at an access point beyond your direct connect. SLAs could also help the ISP by

blocking a compromised host in your network from using your outbound link to perform nasty hacks on other networks.

Lastly, when you first arrive on location to where an incident has occurred, there is a good possibility that more changes have occurred between the time you heard about the incident and your arrival. Do not dismiss anything as untrue without checking into it first. Be careful to use a "need to know" profile to prevent tipping off the intruder. He might be an insider.

Eradication

In this phase this organization certainly learned a lot about why they need current backups. The ability of knowing which systems were corrupt and which were not was paramount when determining how long the system would remain off-line.

Starting from scratch is the best way to restore a host, but it is also the longest path to a working condition. Certainly, the team could not blindly trust anything existing on the host after contamination.

According to SANS Incident Handling training course it is quite possible that if your attacker is part of the hacking community, they are going to tell all their other hacker buddies about your discovered vulnerability. This means you could expect many additional attempts to take place at your facility. Simply put, if you decide not to scrap the old configuration and simply apply the patch on an already compromised system, there's a good chance you are going to be compromised again.

After your systems are back in a clean, working condition, it would be a good idea to run your own set of vulnerability checks. This is the same information that "newbies" in the hacker community (upon hearing about your site) will attempt on your system. This is a preventive measure that helps ensure your system is up to date with current patches and that you have not accidentally missed something.

Recovery

This was one area that was learned by doing. The primary accomplishment was to do another check for vulnerabilities after the system was returned to production. This is how you keep the returning attacker from compromising the standing vulnerability. It is also the way you shore up anything you have missed in the past or anything new to the system since.

Spotting a weakness early can also mean preventing your systems from becoming a "helper" host for other attackers. In other words, you want to prevent hackers from using your system to penetrate other systems, or worse, another host on your own network!

A very good recovery technique to bring a server back online was to use Tripwire in reverse. From the SANS Incident Handling course, if you load Tripwire on a brand new system, it gains all the file integrity information of that system. You can use that integrity structure to compare it against a possibly compromised system. This way you can check for changes to see if a system has been compromised or not. If it has, you now know what the changes are and can fix them.

The validation part of recovery is also in this phase. After the system is back in working order, it needs to be validated by someone other than the person who was doing the work. It is important this happens to make sure that the operations are correct and the system is functioning properly. The handlers are responsible for each individual piece, but someone else should handle the system as a whole.

Monitoring the system for a while at the end of the recovery process increases your chances of success and positive reinstallation. Once the system is functioning and signed off by someone as complete and in working order, it should go through a week of intense monitoring. This will set everyone at ease and the routine of daily security process should return to business as usual. According to SANS Incident Handling course, it is more likely the attacker will try to return soon after the incident was recognized than simply move on to another victim.

Lessons Learned

The one thing that was not done during the Nimda worm incident was to work on a lessons learned phase. From the wireless ARP poisoning example above, the organization can see the many things that could have been done to minimize stress levels, worries, and the number of days required to recover the system.

We need to be aware that there will continue to be a plethora of vulnerabilities available to the hacker. It is always a good idea to continue to monitor public alert notices and bulletins for new vulnerabilities as part of your daily security regime. We need to constantly interpret the changing environment of the Internet and hacking communities to be effective security professionals. Know what you have in your systems and how to recover from any catastrophe either natural or deliberate. A little preparation goes a long way.

REFERENCES

Andrew Conry-Murray. "*Swatting Persistent Security Pests*." Network Magazine December 2001:36-42.

Assessing Wireless Security with AiroPeek. Walnut Creek, CA: WildPackets, Inc. 2001.

Bardwell, Joe. "WildPackets' Wireless Seminar." Jefferson Hotel, Washington DC. December 7, 2001.

"*Cisco Comments on* Recent WLAN Security Paper *from* University of Maryland." San Jose, CA: Cisco Systems. Posted Nov 1 2001. http://www.cisco.com/warp/public/cc/pd/witc/ao350ap/prodlit/1327_pp.htm

Cigital, Inc.; "Wireless Access Points and ARP Poisoning: Wireless vulnerabilities that exposed the wired network", by Bob Fleck and Jordan Dimov; http://downloads.securityfocus.com/library/arppoison.pdf

Gail Meredith. "*Securing the Wireless LAN.*" Cisco's Packet Magazine Third Quarter 2001:74-77.

Held, Gilbert. Understanding Data Communications: From Fundamentals To Networking. West Susses, England: Wiley, 1991.

IEEE 802.11b Standard. IEEE 1999. http://a957.g.akamai.net/7/957/3680/v0001/standards.ieee.org/reading/ieee/std/lanman/802.11b-1999.pdf

Klaus, Chris. "WLAN Security FAQ" October 8, 2001. Online Posting. Pen-Test Mailing List, SecurityFocus.com.

Muller, Nathan J. Bluetooth Demystified. New York, NY: McGraw-Hill, 2001.

RFC 826. "An Ethernet Address Resolution Protocol" David C. Plummer. Nov. 1982. http://www.ietf.org/rfc/rfc0826.txt?number=826

Rhoades, David. "BoF: Wireless Discussion." SANS Cyber Defense Initiative. Grand Hyatt Hotel, Washington, DC. November 30, 2001.

Rik Farrow. "*Wireless Security: A Contradiction in Terms?* " Network Magazine December 2001:74-76.

"Track 4 – Advanced Incident Handling & Hacker Exploits." SANS Institute. Grand Hyatt Hotel, Washington DC. December 2001.

Whalen, Sean. “An Introduction to ARP Spoofing.” April 2001 rev.1
arpspoof@gmx.net

WildPackets’ Guide to Wireless LAN Analysis. Walnut Creek, CA: WildPackets, Inc. 2001.

Wireless LAN Security. San Jose, CA: Cisco Systems. 2001.
http://www.cisco.com/warp/public/cc/pd/witc/ao350ap/prodlit/a350w_ov.htm

Last Updated: July 24th, 2017

Upcoming Training

Security Awareness Summit & Training 2017	Nashville, TN	Jul 31, 2017 - Aug 09, 2017	Live Event
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Memphis SEC504	Memphis, TN	Aug 21, 2017 - Aug 26, 2017	Community SANS
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Mentor Session AW - SEC504	Milwaukee, WI	Aug 23, 2017 - Sep 29, 2017	Mentor
Mentor Session AW - SEC504	New York, NY	Aug 24, 2017 - Sep 08, 2017	Mentor
Mentor Session - SEC504	Denver, CO	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS vLive - SEC504: Hacker Tools, Techniques, Exploits and Incident Handling	SEC504 - 201709,	Sep 05, 2017 - Oct 12, 2017	vLive
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
SANS Dublin 2017	Dublin, Ireland	Sep 11, 2017 - Sep 16, 2017	Live Event
Mentor AW - SEC504	Santa Clara, CA	Sep 11, 2017 - Sep 22, 2017	Mentor
Mentor Session - SEC504	Arlington, VA	Sep 20, 2017 - Nov 01, 2017	Mentor
SANS SEC504 at Cyber Security Week 2017	The Hague, Netherlands	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Columbia SEC504	Columbia, MD	Sep 25, 2017 - Sep 30, 2017	Community SANS
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
Mentor Session - SEC504	Boston, MA	Sep 26, 2017 - Nov 07, 2017	Mentor
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Mentor Session AW - SEC504	Houston, TX	Oct 02, 2017 - Dec 11, 2017	Mentor
Mentor Session - SEC504	Columbia, SC	Oct 03, 2017 - Nov 14, 2017	Mentor
SANS October Singapore 2017	Singapore, Singapore	Oct 09, 2017 - Oct 28, 2017	Live Event
Community SANS Chicago SEC504	Chicago, IL	Oct 09, 2017 - Oct 14, 2017	Community SANS
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event