SySto Technology Consulting

43 Bancroft Rd
Stoughton, Ma 02072
(617) 610 1527
info@systo.com

Attacks from Within

A White Paper About Security in Corporate Networks

Christian Donner, 1/7/2002

Introduction

For most users, corporate networks have become the primary way to access the Internet. Security is rarely a concern. Users feel safe, because the network is behind a firewall. If a transaction requires privacy, the provider of the service generally offers SSL - secure connections (Secure Socket Layer) to encrypt data sent from and to the user's browser.

However, FBI research has shown that 85% of all attacks are launched from within the organization, and that in most cases the attacker is authorized to work on the system [1].

The network protocol that the Internet is build on, TCP/IP, offers loopholes for security attacks. Everybody who accesses the Internet through a LAN, should be aware of the risks that are involved, of the techniques that are available to potential attackers and of the defensive measures that can be taken.

Today's LANs

Most local area networks today use the TCP protocol over Ethernet. Network clients either have a static or dynamically assigned address. Every data packet sent over the network carries the destination address with it. Routers and gateways serve as gates to the outside world. LAN clients send their outbound packets to the local gateway, if the destination address is outside of the local address space. The gateway forwards those packets to the Internet.

Usually, Ethernet network adapters ignore data packets that are not addressed to them. However, they can be programmatically switched into 'promiscuous mode'. In this mode, normally used for diagnostic purposes, the NIC accepts all packets that are sent over the network. Software can read those packets and extract information. *Tcpdump* [2] is a so-called network sniffer. It displays data packets and even offers filtering functions.

Password Sniffing

Network traffic is usually unencrypted. Passwords for network login, email servers and other applications are transmitted in clear text. In theory, anyone on the network can receive this data and read the passwords, using a sniffer.

In practice, it is a very difficult task to read passwords, though, because of the sheer amount of data that travels across the wires, even in small networks. But there is a simpler method:

Password sniffers are tools that know the login protocols for certain applications. They run in the background and quietly log user ids and passwords. *Dsniff* [3], part of a set of utilities written by Dug Song, knows over 50 login protocols, among others for POP, IMAP, Napster and ICQ.

HTTP Traffic

Web servers and browsers communicate via the HTTP protocoll. HTTP packets contain the address of the page, or URL, that is being viewed. Using *webspy* [3], another one of Dug Song's utilities, other users can capture HTTP packets, extract the URL and direct their own browser to follow the clicks of any person on the network.

Similarly, tools like *msgsnarf* [3] and *mailsnarf* [3] can capture AOL Instant Messenger traffic or regular emails containing certain words, and even save them in mail format to be viewed comfortably with the local email reader.

A tool called *Juggernaut* [4] takes attacks to the next level. *Juggernaut* is a network sniffer that can also be used to hijack TCP sessions.

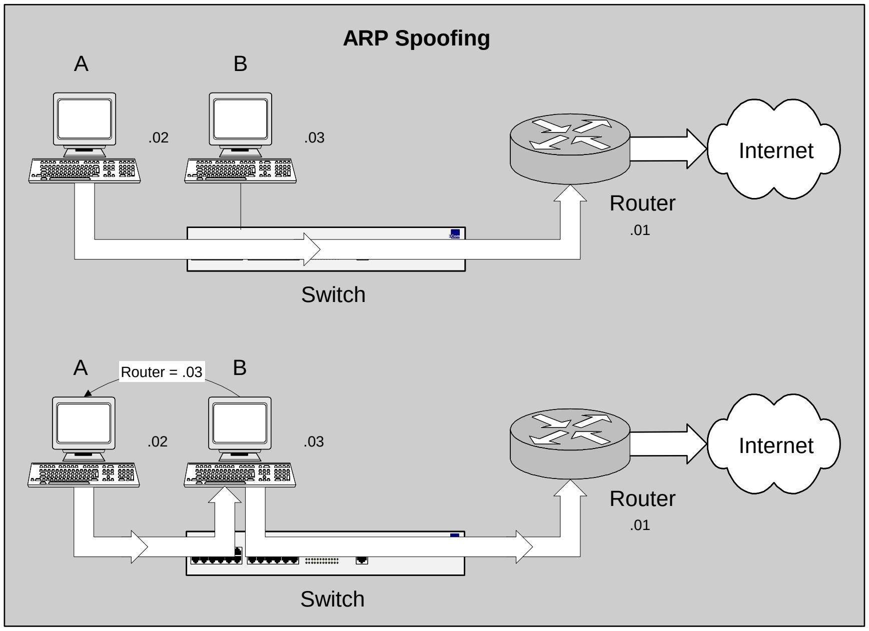
Juggernaut can be activated to watch all network traffic on the local network, or can be set to listen for a special "token". For example, *Juggernaut* can be configured to wait for the login prompt, and then record the network traffic that follows (usually capturing the password). By doing so, this tool can be used to historically capture certain types of traffic by simply leaving the tool running for a few days, and then the attacker just has to pick up the log file that contains the recorded traffic. This is different than regular network sniffers that record all network traffic making the log files extremely huge (and thus easy to detect).
But the main feature of this program is its ability to maintain a connection database. This means an attacker can watch all the TCP based connections made on the local network, and possibly "hijack" the session. After the connection is made, the attacker can watch the entire session (for a telnet session, this means the attacker sees the "playback" of the entire session. This is like actually seeing the telnet window).
When an active session is watched, the attacker can perform some actions on that connection, besides passively watching it. *Juggernaut* is capable of resetting the connection (which basically means terminating it), and also hijacking the connection - allowing the attacker to insert commands into the session or even to completely take the session into his/her hands (resetting connection on the legitimate client).

ARP Spoofing

After reading the above paragraph, administrators of larger corporate networks will feel save now. They may think not be affected, because they use *switches* in their network, as opposed to less expensive *hubs*. Switches establish point-to-point connections between clients, whereas hubs broadcast the data to all the connected clients. Obviously, switches use the available bandwidth more effectively. A network sniffer, attached to a regular port of a *switch*, will normally only display data that is addressed to this very client.

Unfortunately, hackers have found ways around this restriction. The most simple is called ARP (address resolution protocol) spoofing.

In a LAN, IP packets are transmitted in an Ethernet frame that contains the address of the network interface card of the destination computer (also called the MAC address). The sender normally gets this information by sending out an ARP request – "who has 192.168.1.3, tell 192.168.1.2" – for instance. The computer at 192.168.1.3 will respond with an ARP reply packet "192.168.1.3 is at 0:0:b4:53:3a:37", to continue with this example. The sender will save this information for a certain amount of time in its ARP cache so that it does not have to be repeated for every data packet that is sent out. The problem with ARP is that all reply packets end up in this cache, even it they were not requested. An attacker can make use of this by sending out large numbers of reply packets containing his own MAC address. Other clients in the network will then end up sending packets to the attacker instead of the default router. If IP forwarding is activated on the hacker's computer, it will forward the packets to the real gateway and the attack will remain unnoticed by the victim.



Another target for ARP-spoofing is the local DNS server. It resolves Internet domain names like www.systo.com and returns the corresponding IP address (for instance, 66.78.62.220). Using a tool like dnsspoof [3], an attacker could redirect DNS requests to his workstation and return the IP address of another server. If this other server runs a web site that looks similar to the real one, it is an easy task to collect passwords, credit card numbers and other confidential information.

This method can be adapted to free up the bandwidth which is used to load banner ads, by spoofing the DNS addresses of popular ad servers and routing requests to them to the local loopback address (127.0.0.1), hence neutralizing them.

SSL

Critical data, like online purchase orders containing credit card numbers or online banking transactions, is transmitted securely nowadays. This means, both instances, web server and browser, encrypt data before it is sent across the network. The recipient uses the same key to decrypt the information after it has been received. The data is protected from any party listening in between. The encryption itself is considered secure, especially if a 128-bit key is used.

Hackers soon realized that the only way to attack a secure connection is at the end points. Assuming that web servers at trusted service providers are secure, there are only two remaining possibilities: introducing a Trojan Horse to the victim's computer, or what is known as the man-in-the-middle attack [5].

The man-in-the-middle attack scenario involves three hosts: the attacker, the victim, and the target:

- Attacker is the system used by the attacker for the hijack
- Victim is the system used by the victim for Telnet client connections to the target system
- Target is the target system that the intruder wants to compromise

When a secure communication is established, both systems exchange their public keys over the Internet [6], which represents a potentially hostile network. A 'man-in-the-middle' can intercept this exchange and send his own keys to both the victim and the target. The attacker will then capture all following encrypted data packets, strip the sender's key, replace it with his own and forward the packet to the recipient, who will think he is communicating with the real sender.

Unfortunately, without major restructuring of the SSL protocol (and the SSH protocol, for that matter), there is no simple protection against this form of attacks. Educate your users about the risks. Attackers usually don't have access to signed certificates from trustworthy institutions. If configured properly, the browser will display a warning message, if it receives a certificate that was issued for a different site, or that is not signed properly. These messages should not be ignored.

IP Spoofing Detection

Using network-monitoring software such as netlog, packets on the external interface that have both its source and destination IP addresses in the local domain indicate that an attack is currently going on. Netlog is available by anonymous FTP from [7].

Another way to detect IP spoofing is to compare the process accounting logs between systems on the internal network. If the IP spoofing attack has succeeded on one of the systems, there may be a log entry on the victim machine showing a remote access; on the apparent source machine, there will be no corresponding entry for initiating that remote access.

Detecting a Hijacking tool

When the intruder attaches to an existing terminal or login connection, users may detect unusual activity, such as commands appearing on their terminal that they did not type or a blank window that will no longer respond to their commands. Encourage your users to inform you of any such activity. In addition, pay particular attention to connections that have been idle for a long time. Once the attack is completed, it is difficult to detect. However, the intruders may leave remnants of their tools. For example, you may find a kernel streams module designed to tap into existing TCP connections.

Preventing IP spoofing

The best method of preventing the IP spoofing problem is to install a filtering router that restricts the input to the external interface (known as an input filter) by not allowing a packet through if it has a source address from the internal network. In addition, you should filter outgoing packets that have a source address different from your internal network in order to prevent a source IP spoofing attack originating from your site.

Further Information

The End of SSL and SSH? (Kurt Seifried, SecurityPortal.com)

Spionage am Arbeitsplatz (Jürgen Schmidt, C’t 12/2001)

CERT® Advisory CA-1995-01 IP Spoofing Attacks and Hijacked Terminal Connections

CERT® Advisory CA-1996-21 TCP SYN Flooding and IP Spoofing Attacks

References

[1] The Network Security Crisis - A White Paper
http://www.attrition.org/~modify/texts/security/whtpap2.html

[2] http://www.tcpdump.org/

[3] http://www.monkey.org/~dugsong/dsniff/

[4] http://www.phrack.org/show.php?p=50

[5] http://www.incrypt.com/mitma.html
http://www.sans.org/infosecFAQ/threats/middle.htm

[6] RFC 2409: Internet Key Exchange
http://www.freesoft.org/CIE/RFC/bynum.cgi?2409

[7] ftp://net.tamu.edu/pub/security/TAMU/netlog-1.2.tar.gz
MD5 checksum: 1dd62e7e96192456e8c75047c38e994b