Network Security

Author: A.J.Gillette
Date: December 4, 2012
Revision: 1.1

Table of Contents

Introduction

This document describes some of the most common ways that information security can be compromised and proposes simple guidelines to protect that information. Some of these guidelines are easy to ignore, with no significant consequences, if the company is small. Network security guidelines tend to be independent of company size and are harder to ignore.

Many of the recommendations that follow have been extracted and or summarized from the book “Counter Hack Reloaded” by Ed Skoudis and Tom Liston.

Low Tech Guidelines

Just like bank robbers, effective attackers do their homework before launching a single attack packet. One of the most effective low-tech ways to get key information is through “Social Engineering”.

Social Engineering

The following quote sums up the social engineering approach. “The best social engineers improvise, acting their way through a telephone call using techniques that might earn them an Academy Award if they were in the movie business.”

This is less effective in a small company where everyone knows everyone else, but in a larger company the following has been very effective.

- A new employee calls a help desk trying to figure out how to perform a task on a computer.
- An angry manager calls a lower level employee because a password has suddenly stopped working.
- A system administrator calls an employee to fix an account on a system that requires a password.
- An employee has lost some important information and calls another employee to get remote access credentials

The most effective defense against Social Engineering is awareness.

Physical Break in

Physical break-in does not necessarily mean dressing in black, waiting until after hours and picking a lock. External attackers can pose as employees that have forgotten a badge or as customers coming for a meeting. In some cases attackers may be legitimate visitors with laptops containing software that can perform inside out attacks.

Suggestions for protection against physical attack:

- Badge access for all employees.
- Sign in sheet for employees that forget their badge.

- Locks on all computer room doors.
- Screen savers on all employee computers that engage automatically after 5 minutes.
- Travelling workers with laptops, should have encryption software to protect file systems in case their laptops are stolen.

Dumpster Diving

Discarding paper notes and media with passwords or other sensitive information can be intercepted by simply going through the dumpster behind the building. The best defense against this is to use shredding services for media and paper. Things to watch for include significant information rich trash events like office moves. One other policy that is strongly recommended is that discarded computers must have hard drives removed. If the drives cannot be re-used, they should be destroyed.

Web site Reconnaissance

Web sites can be searched for all kinds of information including files that may not be visible using a browser. In many cases web site creators will use spreadsheets and other documents to create web site content. Search engines like Google use bots to build searchable databases that contain file information. Applications provided by Google like Google Bots, The Google Index, The Google Cache and the Google API, can all be used to prepare for a cyber attack.

To protect a company from unintentionally leaking key information, any server with a web page exposed to the Internet should contain no sensitive content. All files on web sites should be “html” or “php” files with no generated content. Non-visible files should be removed and web sites should be hosted on external cloud based servers that are not connected to the corporate infrastructure.

The use of news groups and mailing lists from within the company’s network is not recommended. If employees have to use news groups or mailing lists, care should be taken to not expose the list to the public.

A Google index audit should be done once a year to make sure that Google has not cached information that is inaccurate or sensitive. It is possible to manually remove pages from the Google cache if need be.

DNS Servers

DNS servers are responsible for making things easy to find on the Internet. They change human readable names into addresses that machines use to index web based information. Unfortunately, this can be used to provide less desirable parties with lists of potential attack targets.

It is not possible or desirable to complicate DNS lookups, but there are some things that the company can do to make DNS abuse a bit more difficult. One of the most useful reports is a zone transfer. Zone transfers are typically used between primary and secondary DNS servers. Zone transfer responses should be restricted or disabled.

DNS registrations should be reviewed to make sure that things like company and server names and references to the OS running on the server are not part of the name.

HINFO and TXT records should be eliminated because they tend to advertise things like the OS type.

All internal DNS queries should be directed to an internal DNS server, which communicates directly with a trusted external DNS server. Queries from internal systems to external DNS servers should be blocked.

Technology Guidelines

Wireless Access Points

Wireless access points are targets for the geeky pastime call “War Driving”. A War Driver’s goal is to locate WLANs and determine their ESSID. People will drive around business parks with direction antennas and laptops scanning for WLANs. The original encryption scheme used to protect wireless access points called “WEP” has flaws that make it easy to determine WEP Key. Once a WAR Driver has the WEP key they have access to your internal network. With the appropriate software a War Driver can crack a WEP key with 100-800MB of data transfers.

In the past, dial up modems represented another point that could be used to gain network access. Fortunately they are for the most part, no longer used.

To defend against War Driving the first thing that should be done is to make sure that the SSID does not bring unwanted attention to your network. SSID names should

not contain the companies name and ideally should be coded to make them obvious to employees, while meaning nothing to War Drivers.

The second step is to make sure that access points are configured to ignore probe requests that don't include the SSID. Access points should also not include the SSID in beacon packets.

The third step is to make sure that access points are not configured to use WEP for key encryption. All access points should use WPA or some stronger version of key encryption.

The forth step is to make sure that all access points are not configured to use IKE for key exchange.

Although not very likely, dial up modems should not be connected to any device that has access to the internal company network.

For the best defense, Wireless access should require the use of VPN's. There should be no unencrypted wireless access.

Network Mapping

Once an attacker has access to your internal network the next step is to map the internal network topology. The mapping process will give the attacker a list of key hosts/servers, routers and firewall/gateways. The first step in the mapping process is sweeping. Sweeping means that attackers use icmp(ping), tcp connections to port 80 or UDP packets to determine what IP addresses are in use. Once attackers know what IP addresses are alive they use tools like "traceroute" to build a location map or topology. Traceroute relies on the TTL field in the IP header to get a response from each node in a link.

To defend against network mapping, you should block the messages that the network mapping tools use. The primary two message types are ICMP echo requests (ping) and trace route responses. For external systems, firewall rules should be adjusted to provide ping responses to ISPs only.

Port Scanning

Once an attacker has a list of live nodes in a network and some idea of the network topology, the next step is to determine the purpose of each system. One way to get information about a system is by trying to connect to Ethernet ports. Port scanners like nmap, systematically try to connect to all of the 65,536 ports using UDP and TCP to determine which ports are in use. Nmap will use the discovered information produce a report that attackers can use to focus their attacks.

To defend against port scanning you can check periodically which TCP and UDP ports are open.

On windows you would do this with the following command:

C:\> netstat –na | find “LISTENING”

On the Mac or Linux you would use the command:

netstat –na | grep LISTENING

Once you have a list of ports you must evaluate whether each network service is required on the target box. If the service is not needed you disable or kill the process using the process id. Disabling the wrong service can make the entire system unstable so this should be done carefully with testing to ensure that nothing has been broken. Another useful command on MAC and Linux machines is “lsof”.

lsof –i

Will list all TCP and UDP ports in use.

lsof –p [pid]

Will then give specific detail about the app using those ports. This will make it easier to track down apps. In Unix based systems, services are started in the file “/etc/initd.conf”. Services can be disabled by commenting out lines in that file. To get a list of services and their status us the command:

chkconfig –list

To disable a service you can type:

chkconfig [svc_name] off

The next step in a security eval would be to list and examine the firewall rules. The following is a list of the rules for a stock HDPassport.

```
Chain INPUT (policy DROP 177K packets, 26M bytes)
 pkts bytes target     prot opt in     out     source       destination
    0     0 DROP       udp  --  eth0   *       0.0.0.0/0    0.0.0.0/0     udp dpt:5060 STRING match "REGISTER sip:" ALGO name bm FROM 28
TO 80 ICASE
```

```
    0     0 DROP       tcp  --  eth0   *       0.0.0.0/0    0.0.0.0/0     tcp dpt:5060 STRING match "REGISTER sip:" ALGO name bm FROM 28
TO 80 ICASE
    1   333 ACCEPT     udp  --  eth0   *       0.0.0.0/0    0.0.0.0/0     udp dpt:5060 limit: up to 50/sec burst 100 mode srcip
    0     0 DROP       udp  --  eth0   *       0.0.0.0/0    0.0.0.0/0     udp dpt:5060
95922   21M ACCEPT     all  --  *      *       0.0.0.0/0    0.0.0.0/0     ctstate RELATED,ESTABLISHED
   26  1508 DROP       all  --  *      *       0.0.0.0/0    0.0.0.0/0     ctstate INVALID
    0     0 ACCEPT     udp  --  eth1   *       0.0.0.0/0    0.0.0.0/0     udp spt:67 dpt:68
    0     0 ACCEPT     icmp --  *      *       0.0.0.0/0    0.0.0.0/0     icmp type 3
    0     0 ACCEPT     icmp --  *      *       0.0.0.0/0    0.0.0.0/0     icmp type 4
    0     0 ACCEPT     icmp --  *      *       0.0.0.0/0    0.0.0.0/0     icmp type 11
    0     0 ACCEPT     icmp --  *      *       0.0.0.0/0    0.0.0.0/0     icmp type 12
 4515  379K ACCEPT     icmp --  *      *       0.0.0.0/0    0.0.0.0/0     icmp type 8
 264K   12M ACCEPT     all  --  lo     *       0.0.0.0/0    0.0.0.0/0
   11  1746 ACCEPT     all  --  eth1   *       0.0.0.0/0    0.0.0.0/0
   71  3496 ACCEPT     tcp  --  eth0   *       0.0.0.0/0    0.0.0.0/0     tcp dpt:5060 ctstate NEW limit: up to 10/sec burst 10 mode
srcip
    0     0 DROP       tcp  --  eth0   *       0.0.0.0/0    0.0.0.0/0     tcp dpt:5060 ctstate NEW

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source       destination

Chain OUTPUT (policy ACCEPT 362K packets, 20M bytes)
 pkts bytes target     prot opt in     out     source       destination

Chain fail2ban-apache (0 references)
 pkts bytes target     prot opt in     out     source       destination

Chain fail2ban-apache-nohome (0 references)
 pkts bytes target     prot opt in     out     source       destination

Chain fail2ban-apache-noscript (0 references)
 pkts bytes target     prot opt in     out     source       destination

Chain fail2ban-apache-overflows (0 references)
 pkts bytes target     prot opt in     out     source       destination

Chain fail2ban-ssh (0 references)
 pkts bytes target     prot opt in     out     source       destination

Chain fail2ban-ssh-ddos (0 references)
 pkts bytes target     prot opt in     out     source       destination
```

The first step in defending against port scans is to make sure that there is nothing open to scan. The best tool for the job is NESSUS manufactured by Tenable Network Security. The tool is free for home use and is $1200.00 US per year for commercial use.

An active defense strategy

Once an attacker has access to the companies network, he/she will use port scanners and other noisy tools to try to determine the network topology. Intrusion Detection Systems (IDSs) can be deployed to detect the network traffic and Intrusion Prevention Systems (IPSs) can be deployed to detect attack signatures and prevent/minimize damage. An IDS will match network traffic to a normal use signature and if an abnormality is detected, will match the abnormality against known attack signatures. The IDS will email the system administrator with the results. An IPS would start dropping packets from the attacker and trigger port closures to block an attack before the attacker reaches his/her target.

Attackers can avoid detection by fragmenting packets in unexpected ways and in large quantities.

IDSs and IPSs are available as commercial and open source tools. Setting up and maintaining these tools can be a time intensive process.

Application and Operating System attacks

Buffer Overflow Exploits

Buffer overflow attacks started with the publishing of a “how to” paper in 1996. Since that time, applications and operating system exploits have been posted daily. Buffer overflow exploits are based on the attacker sending more data than the programmer expected and are only effective if the programmer did not place bounds on the input data. Programming languages like C and C++ allow data to overwriting adjacent variables with executable machine code, which when executed allows the attacker to take over the target at the most basic level. Once in control, the attacker can create new user accounts, remotely control the GUI or alter the systems configuration. Attackers will use source code from known programs like Apache to determine what data will cause the program to execute their code instead.

Care should be taken to make sure that any system connected to the Internet does not expose programs known to have exploits. By using operating systems like Linux and keeping them up to date via public repositories, you can prevent the most common attacks. By hosting your web site on cloud-based servers and limiting the number of open ports, you can greatly decrease the probability of this type of attack. “IT” should perform regular audits on all publically connected systems to make sure they are not venerable.

Password Attacks

Passwords are perhaps the most commonly used security tool. Unfortunately the use of passwords is largely unregulated and tends toward simplicity in order to aid the human memory. In many cases system default passwords are never changed and are publically available via a Google search. The Phenoelit hacking group based in Germany maintains a large database of default passwords. Policies should be put in place to insure that passwords are changed and as strong as possible. It is also good practice to force password changes on a regular basis.

Even with a good password policy, there are many programs that can be deployed to guess and crack your security. Once of the most effective password guessing programs is called Brutus. It runs on windows platforms and has a very easy to use GUI interface. THC Hydra is a similar tool for windows.

Account lockout tools are perhaps the best defense against password guessing.

Password cracking differs from password guessing in that the attacker has an encrypted version of the password and will apply decryption software to recover the original. Packages like Cain, “John the ripper”, Pandora and LC5 have been used for years to find and decrypt passwords.

The best way to protect and organization against password cracking is to insure strong passwords are in use, rotate passwords on key resources, conduct regular password cracking tests, and use tools other than passwords for access to key assets.

Security FOBs with rotating access keys and VPN tunnels with certificates and keys can be used both internally and externally to protect key systems.

Web Application Attacks

As more companies rely on the web for e-commerce and portals with information not available to the general public, web application attacks have grown in popularity. In many cases information about the underlying data is available in the URL that is displayed by an error page. Sometimes a web application will differentiate between a bad user id and or a password for example. This tells the attacker that the user id is valid and that he/she only has to concentrate on the password cracking.

It is common practice for web applications to be session based. Session IDs can be embedded in the underlying HTML or included in the URL response. Cookies are perhaps the most common form of maintaining session information. Many web-based applications have problems with the generation and tracking of session information making it possible for attackers to highjack or clone a session.

An attacker logs into a server with their own credentials and then changes the session number to clone the session of another valid user.

Web application manipulation proxies like Achilles, Paros Proxy and web Scarab can be used to clone a session, giving the attacker access to all available information.

Web applications that use databases like SQL often include SQL search commands as part of the HTML web pages. By carefully constructing an SQL query is part of a user input field, an attacker can gain access to other information in the underlying database. This is sometimes called SQL-Injection.

To defend against web application attacks, care should be taken to insure that the underlying software does not unnecessarily expose information to an attacker. Make sure that the web application software uses digital signatures and hashing software to prevent exposing information to attackers. To defend against SQL-Injection, web pages should include filters to control user-entered data.

Network Attacks

Network attacks happen if the attacker has physical access to your network infrastructure or has gained access to a machine on your network. Once the attacker has access, there are a number of tools that can be used to gather information directly from your network.

Network sniffers are a common and very effective way to acquire information about network topologies and the applications running over them. That said switched networks by definition make it harder for network sniffers to get a complete picture. Network switches us ARP to bind IP Addresses to MAC addresses which means that they will only forward traffic to a port of the device destination for that traffic is associated with that port.

In other words, if a sniffer is connected to a switch port it will only see broadcast traffic and/or traffic destined for the system the sniffer is running on.

There are several ways that attacker use to get around this problem. The first is to force the switch into HUB mode by flooding it with traffic. Many inexpensive and older switches will fail to HUB mode where all traffic is forwarded out all ports.

The second method is to poison the ARP table in the target system with false ARP responses. Using this approach, the attacker sends a message to the target system to tell it that the default gateway for the LAN segment is at a different MAC address. Once this is in place all network traffic destined for the default gateway will be sent to the attackers machine instead. The attackers machine then forwards the traffic to the real default gateway after it has inspected it.

For key network assets hard coding the binding of the MAC and IP Address can prevent this type of attack. If the binding is hard coded the false ARP responses are ignored. When this happens attackers have to be even more creative and use a technique called port stealing.

The target for a port stealing attack is the switch not an endpoint. The attacker floods the switch with bogus Ethernet packets that contain the attacker's MAC

address listed as the destination. This is hard to detect because the packets in question are originating and terminating on the same port. The key point here is that the origination address in the packets has been set to the machine the attacker wants to monitor. The table in the switch that keeps track of what machines are on what port gets confused because the bogus packets fool the switch into believing the machine is connected on a different port. If the machine in question is the networks default gateway, then all traffic on the LAN destined for the default gateway will be sent to the new port where it can be inspected.

Programs like Ettercap can run the port stealing attack, intercept traffic, run a corrective action and forward the packets to their original destination. They alternate between confusing and fixing the switch to make sure the attack goes undetected.

Another form of attack involves sniffing and redirecting DNS queries. DNS is used to assign easy to understand labels to cryptic network addresses like 192.168.217.22.

Programs like Dsniff can issue false DNS responses to tell network devices to send their traffic to the attacker's machine instead of the real destination. If the attacker knows the DNS in advance then the attack is easy. If the attacker must sniff the LAN to get the DNS, then one of the previous methods for messing with the switch will be required to gather information.

When a machine sends out a DNS query it will get a response from the DNS server. If the attacker immediately sends the false DNS response the real response is replaced/updated by the false response in the receiving machines cache. Fro that point forward all traffic to the remote system will be redirected to the false IP address.

The next attack that we will consider is on https, ssl and ssh connections. These connections are supposed to be secure but there are several ways to monitor data running over these connections.

To conduct a man-in-the-middle attack the attacker will typically start with a dns-spoof to fool the target machine into sending data to the attacker directly. The Dsnif tool comes with two additional programs called webmitm and sshmitm. Both of these programs were designed to perform man-in-the-middle actions that will insert the attackers machine into a secure stream.

When the victim's computer starts a secure connection, the attackers computer replies with a generated certificate. This will probably cause the victim's computer to display a cryptic warning saying that the certificate can not be verified. Most users will ignore this message and continue.

The attacker's computer then establishes a second connection to the destination computer and forwards all traffic between the two devices. As the data passes, it is decrypted and can be sniffed.

Both Webmitm and sshmitm displayed the decoded data which usually includes usernames and passwords.

Ettercap is another program that approaches the problem a slightly different way. Ettercap will use its own certificate when the user attempts to connect to a server but when the connection keys are exchanged it will grab the keys. This means that there will be a direct connection from the client to the server but with the keys, ettercap can decode the encrypted ssl/ssh traffic that it sniffs.

So we have established that attackers can be very creative and that even secure links are not that secure. The obvious question then is "How do we protect ourselves?"