**Scan “Pen” Test Report with Nessus**

The following report is a scan “pen test” analysis I develop to identify potential hidden vulnerabilities from a small home network designed for testing. Nessus is the security scanning tool I utilize over the course of this project. This is the most popular vulnerability scanner with advanced capabilities, within which logins are the scripts that perform the tests and is the attack scripting language that can be used to write my own plugins.

I tested two critical work stations that are the pillars of this network:

For security purposes, each machine’s IP address is code named: Guardian Square and Cutting-Edge.

* I could find critical vulnerability information when trying to scan Guardian Square with Nessus:

 Two medium alerts and 33 port scanners:

1st medium alert

* SMB Signing Disabled

This Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

**Solution**

Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

2nd medium alert

* SSL Certificate Cannot Be Trusted

The server's X.509 certificate does not have a signature from a known public certificate authority. This situation can occur in three different ways, each of which results in a break in the chain below which certificates cannot be trusted.

First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

Third, the certificate chain may contain a signature that either didn't match the certificate's information, or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.

If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and identity of the web server. This could make it easier to carry out man-in-the-middle attacks against the remote host.

**Solution**

Purchase or generate a proper certificate for this service.

33 port scanners

* 33 port scanners without major risk factor.

Netstat portscanner (SSH)

**Description**

This plugin runs 'netstat' on the remote machine to enumerate open ports.
See the section 'plugins options' about configuring this plugin.

* A series of medium, low and information of vulnerability aspects were revealed when trying to scan Cutting-Edge with Nessus:

 Four medium alerts, two low ones and one vulnerability categorized as “Info.” Only the most

 important ones will be included in this report.

1st medium alert

SSL Certificate Cannot Be Trusted

**Description**

The server's X.509 certificate does not have a signature from a known public certificate authority. This situation can occur in three different ways, each of which results in a break in the chain below which certificates cannot be trusted.

First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

Third, the certificate chain may contain a signature that either didn't match the certificate's information, or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.

If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and identity of the web server. This could make it easier to carry out man-in-the-middle attacks against the remote host.

**Solution**

 Purchase or generate a proper certificate for this service.

2nd medium alert

SSL Certificate Expiry

**Description**

This plugin checks expiry dates of certificates associated with SSL- enabled services on the target and reports whether any have already expired.

**Solution**

 Purchase or generate a new SSL certificate to replace the existing one.

3rd medium alert

SSL Self-Signed Certificate

**Description**

The X.509 certificate chain for this service is not signed by a recognized certificate authority. If the remote host is a public host in production, this nullifies the use of SSL as anyone could establish a man-in-the-middle attack against the remote host.

Note that this plugin does not check for certificate chains that end in a certificate that is not self-signed, but is signed by an unrecognized certificate authority.

**Solution**

 Purchase or generate a proper certificate for this service.

2nd low alert

SSL Certificate Cannot Be Trusted

**Description**

The remote host supports the use of RC4 in one or more cipher suites.
The RC4 cipher is flawed in its generation of a pseudo-random stream of bytes so that a wide variety of small biases are introduced into the stream, decreasing its randomness.

If plaintext is repeatedly encrypted (e.g., HTTP cookies), and an attacker can obtain many (i.e., tens of millions) ciphertexts, the attacker may be able to derive the plaintext.

**Solution**

 Reconfigure the affected application, if possible, to avoid use of RC4 ciphers. Consider

 using TLS 1.2 with AES-GCM suites subject to browser and web server support.

All in all, the purpose of this scan test with Nessus was to identify potential security flaws of my small network. A lot had been uncovered in that nature. Therefore, necessary solutions and fixes will be implemented as prescribed above.