

Introduzione:

Nessus è uno **scanner di vulnerabilità**, ampiamente usato nel campo della cybersecurity, per la sua efficienza e funzionalità. Tutto ciò che fa è analizzare una rete per scovarne le vulnerabilità, successivamente tramite degli **exploit** va a testare le stesse per dargli un grado di importanza in base a quanto potrebbero essere vulnerabili da un attacco (**High, Mixed, Info** etc...)

Una volta effettuato lo scan e testato gli exploit, compilerà un report, che potrà essere o **riassuntivo** (10-20 pagine), o **dettagliato** (anche 200 pagine).

Nel mondo CS troviamo anche **OpenVas** che è **simile** a Nessus, appartenente alla stessa famiglia ed offre gli stessi servizi, OpenVas però è molto **più lento** rispetto a Nessus, e la sua interfaccia non è **userfriendly** come per OpenVas.

Studio della scansione:

La scansione è stata fatta sulla macchina vittima **metasploitable2**, di cui l'IP è **192.168.0.102**,

```
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:e1:ed:f1
          inet addr:192.168.0.102  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fdd7:21:9d01:8782:a00:27ff:fee1:edf1/64 Scope:Global
          inet6 addr: 2a0e:419:3357:0:a00:27ff:fee1:edf1/64 Scope:Global
          inet6 addr: fe80::a00:27ff:fee1:edf1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:736 (736.0 B)  TX bytes:4585 (4.4 KB)
          Base address:0xd020 Memory:f0200000-f0220000

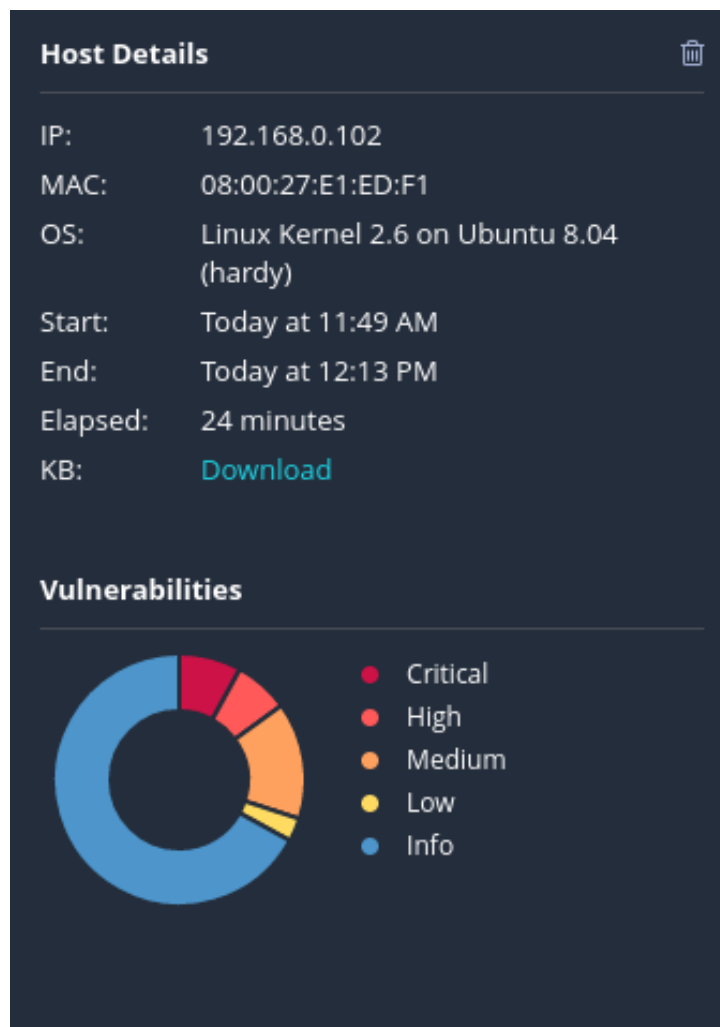
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:92 errors:0 dropped:0 overruns:0 frame:0
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)

msfadmin@metasploitable:~$
```

La scansione ci riporta tutte le **vulnerabilità trovate**, categorizzandole in base alla loro importanza nel venire **mitigare** o **risolte**. Ad esempio le vulnerabilità categorizzate come **Critical** (sono quelle che hanno la precedenza su tutte le altre), scendendo poi da **High**, **Mixed**, **Medium**, **Low** ed **Info**.

Vulnerabilities 61									
Filter Search Vulnerabilities 61 Vulnerabilities									
Sev	CVSS	VPR	EPSS	Name	Family	Count			
<input type="checkbox"/> CRITICAL	10.0 *			VNC Server 'password' Password	Gain a shell remotely	1			
<input type="checkbox"/> CRITICAL	9.8	9.0	0.9728	Apache Tomcat AJP Connector Request Injection (Ghostcat)	Web Servers	1			
<input type="checkbox"/> CRITICAL	9.8			SSL Version 2 and 3 Protocol Detection	Service detection	2			
<input type="checkbox"/> CRITICAL	9.8			Bind Shell Backdoor Detection	Backdoors	1			
<input type="checkbox"/> CRITICAL	SSL (Multiple Issues)	Gain a shell remotely	3			
<input type="checkbox"/> HIGH	7.5	5.9	0.0358	Samba Badlock Vulnerability	General	1			
<input type="checkbox"/> HIGH	7.5			NFS Shares World Readable	RPC	1			
<input type="checkbox"/> MIXED	SSL (Multiple Issues)	General	28			
<input type="checkbox"/> MIXED	ISC Bind (Multiple Issues)	DNS	5			
<input type="checkbox"/> MEDIUM	6.5			TLS Version 1.0 Protocol Detection	Service detection	2			
<input type="checkbox"/> MEDIUM	5.9	4.4	0.9524	SSL DROWN Attack Vulnerability (Decrypting RSA with Obsolete and Weakened eNcryption)	Misc.	1			
<input type="checkbox"/> MEDIUM	5.9	4.4	0.0031	SSL Anonymous Cipher Suites Supported	Service detection	1			

Abbiamo anche un **grafico a torta** che ci fa capire quanto la mia rete sia **sicura** o **vulnerabile**. In questo caso abbiamo **8 vulnerabilità Critical** (8%), **High** (7%), **Medium** (15%), **Low** (3%), **Info** (67%).



Mitigazione e soluzione vulnerabilità:

Critical 1)

La prima criticità è quella che metasploitable utilizza una **password debole**, infatti Nessus è stato in grado di accedervi tranquillamente, la **soluzione** è quella di impostare una **password complessa**, magari usando caratteri speciali, ed usando lettere e numeri insieme;

Vulnerabilities 61

CRITICAL VNC Server 'password' Password

Description

The VNC server running on the remote host is secured with a weak password. Nessus was able to login using VNC authentication and a password of 'password'. A remote, unauthenticated attacker could exploit this to take control of the system.

Solution

Secure the VNC service with a strong password.

Critical 2)

La seconda criticità ci informa che un attaccante tramite il **Connettore AJP**, potrebbe leggere le richieste web fatte dalla macchina attaccata;

CRITICAL VNC Server 'password' Password**Description**

The VNC server running on the remote host is secured with a weak password. Nessus was able to login using VNC authentication and a password of 'password'. A remote, unauthenticated attacker could exploit this to take control of the system.

Solution

Secure the VNC service with a strong password.

E qui abbiamo la soluzione, possiamo o aggiornare la configurazione dell'AJP, o anche consultare i vari forum;

Solution

Update the AJP configuration to require authorization and/or upgrade the Tomcat server to 7.0.100, 8.5.51, 9.0.31 or later.

See Also

<http://www.nessus.org/u?8ebe6246>

<http://www.nessus.org/u?4e287adb>

<http://www.nessus.org/u?cbc3d54e>

<https://access.redhat.com/security/cve/CVE-2020-1745>

<https://access.redhat.com/solutions/4851251>

<http://www.nessus.org/u?dd218234>

<http://www.nessus.org/u?dd772531>

<http://www.nessus.org/u?2a01d6bf>

<http://www.nessus.org/u?3b5af27e>

<http://www.nessus.org/u?9dab109f>

<http://www.nessus.org/u?5eafcf70>

Critical 3)

Il messaggio descrive una vulnerabilità di sicurezza legata all'uso dei protocolli SSL 2.0 e SSL 3.0 per cifrare le comunicazioni di rete. Questi protocolli sono considerati insicuri e obsoleti a causa di diverse vulnerabilità crittografiche.

CRITICAL SSL Version 2 and 3 Protocol Detection**Description**

The remote service accepts connections encrypted using SSL 2.0 and/or SSL 3.0. These versions of SSL are affected by several cryptographic flaws, including:

- An insecure padding scheme with CBC ciphers.
- Insecure session renegotiation and resumption schemes.

An attacker can exploit these flaws to conduct man-in-the-middle attacks or to decrypt communications between the affected service and clients.

Although SSL/TLS has a secure means for choosing the highest supported version of the protocol (so that these versions will be used only if the client or server support nothing better), many web browsers implement this in an unsafe way that allows an attacker to downgrade a connection (such as in POODLE). Therefore, it is recommended that these protocols be disabled entirely.

NIST has determined that SSL 3.0 is no longer acceptable for secure communications. As of the date of enforcement found in PCI DSS v3.1, any version of SSL will not meet the PCI SSC's definition of 'strong cryptography'.

Per risolvere questa vulnerabilità, è consigliabile disabilitare SSL 2.0 e SSL 3.0 e configurare il sistema per utilizzare TLS 1.2 o versioni superiori, come TLS 1.3. O come sempre consultando i vari forum.

Solution

Consult the application's documentation to disable SSL 2.0 and 3.0. Use TLS 1.2 (with approved cipher suites) or higher instead.

See Also

<https://www.schneier.com/academic/paperfiles/paper-ssl.pdf>

<http://www.nessus.org/u?b06c7e95>

<http://www.nessus.org/u?247c4540>

<https://www.openssl.org/~bodo/ssl-poodle.pdf>

<http://www.nessus.org/u?5d15ba70>

<https://www.imperialviolet.org/2014/10/14/poodle.html>

<https://tools.ietf.org/html/rfc7507>

<https://tools.ietf.org/html/rfc7568>