Attacchi ai sistemi
I malware
Attacchi alle password (brute force, dizionario, rainbow table)
I buffer overflow

Con Malware → «Malicious Software», ovvero qualsiasi software che viene utilizzato con l'intento di procurare danno su un sistema operativo, come:

- Causare un «Denial of Service»
- Spiare l'attività degli utenti di un sistema
- Ottenere controllo non autorizzato a dati e sistemi
- Altre attività malevole

La classificazione dei malware si basa sul loro comportamento, non tanto sulle funzionalità di attacco che forniscono.
Tra i tipi di malware distinguiamo: Virus Trojan Rootkit Bootkit Backdoors Adware Worm Spyware Dialer Key-logger Botnet Ransomware

- **Worm**: un worm è formato generalmente da codice molto compatto che si diffonde passando da computer in computer, senza azione diretta o autorizzazione da parte dei sistemi infetti. Si copiano in sezioni particolari all'interno dei file system e i più sofisticati cercano di nascondersi dalle analisi dei vari sistemi di sicurezza (come antivirus, anti malware).

- **Virus**: similare al Worm, con la differenza sostanziale che è necessaria l'interazione umana. L'utente deve avviare fisicamente il file che contiene il virus (in modo tale da concedere al virus le autorizzazioni dell'utente).

- **Trojan horse**: un trojan, è un malware che si nasconde all'interno di un file apparentemente innocuo, come potrebbe essere un eseguibile oppure un documento di office, oppure un PDF. Il malware si attiva quando la vittima apre il file. Tra i trojan più comunemente utilizzati nei penetration testing troviamo le backdoor, che sono generalmente utilizzate per fornire agli attaccanti delle shell sui sistemi infetti.

- **Backdoor**: le backdoor sono dei programmi con due componenti: un server ed un client. Il server gira sulla macchina della vittima mettendosi in ascolto sulla rete e accettando connessioni. Il client gira sulla macchina dell'attaccante e viene utilizzato per collegarsi alla backdoor per controllarla.

- **Rootkit:** è un malware progettato per nascondersi dagli utenti e dalle soluzioni antivirus per prendere il controllo completo del sistema operativo. Un rootkit permette di mantenere privilegi elevati su una macchina senza essere notati

- **Bootkit**: sono dei rootkit che aggirano le protezioni del sistema operativo in quanto entrano in funzione prima dell'avvio completo del sistema operativo, in particolar modo prima dell'attivazione dei moduli di sicurezza di un sistema operativo.

- **Adware:** gli adware sono dei programmi fastidiosi che mostrano pubblicità agli utenti di un personal computer

- **Spyware**: sono dei programmi che si usano per raccogliere informazioni sulle attività degli utenti di un sistema, ad esempio: il tipo di sistema operativo installato sulla macchina, i siti visitati, le password. Queste informazioni vengono inviate successivamente ad un server sotto il controllo dell'attaccante.

- **Dialer**: è (o meglio era) un programma che cerca di chiamare numeri telefonici a pagamento per guadagnare soldi. Non è più utilizzabile su contratti flat.

- **Keylogger**: è un programma che registra ogni tasto premuto sulla macchina della vittima. I keylogger registrano: i tasti premuti sulla tastiera, il nome delle finestre aperte dall'utente. Salvano poi queste info in un file di log che spediscono ad un server controllato dall'attaccante.

ALCUNI POSSONO ESSERE USATI DA NOI, NOSTRI ALLEATI.

Attacchi alle password (brute force, dizionario, rainbow table)

Le password sono la **prima linea difensiva dei sistemi informatici**, ed in molti casi, dove ancora non sono attivi metodi di autenticazione multi fattore, anche l'unica.

Le password vengono salvate utilizzando degli algoritmi di **cifratura a senso unico**, ossia non c'è modo di ricostruire la password partendo dalla sua **forma crittografica → HASH**

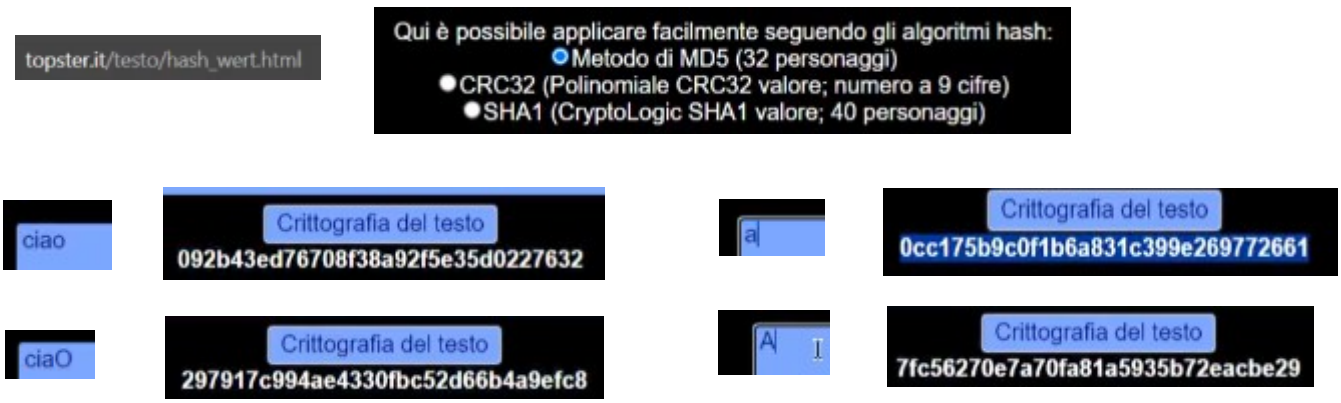
Quando effettuate la login sul vostro PC, il sistema operativo prende la password che avete inserito, ne calcola l'hash e confronta il risultato con l'hash salvato nel DB delle password.

Se i due valori corrispondono, la login va a buon fine.
Il sistema operativo non ha bisogno di conoscere il valore in chiaro della password!

Si parla dei sistemi offline (computer, telefoni, applicazionmi,..) sia log in online(Fb, amazon,..).

>In che senso non ha bisogno della psw in chiaro? "Appena viene digitata la psw in chiaro, prima di mandarla in database per il confronto, viene immediatamente convertita in HASH. Sia sistema offline sia online."
>Hash cambia ogni volta che digitiamo la psw? "NO. Imutabile e confondibile."

Forzare la scoperta della psw in chiaro o usare l'hash per fare log-in.



Il password cracking → è il processo per recuperare le password partendo dalla loro forma crittografica.

Si tratta di un processo di deduzione: l'attaccante che cerca di indovinare la password, ne calcola l'hash e poi lo confronta con quello contenuto del database delle password.
Per automatizzare il processo di password cracking, ci sono due strategie che abbiamo già avuto modo di vedere nelle lezioni scorse:

- Attacchi di forza bruta (brute force)
- Attacchi a dizionario

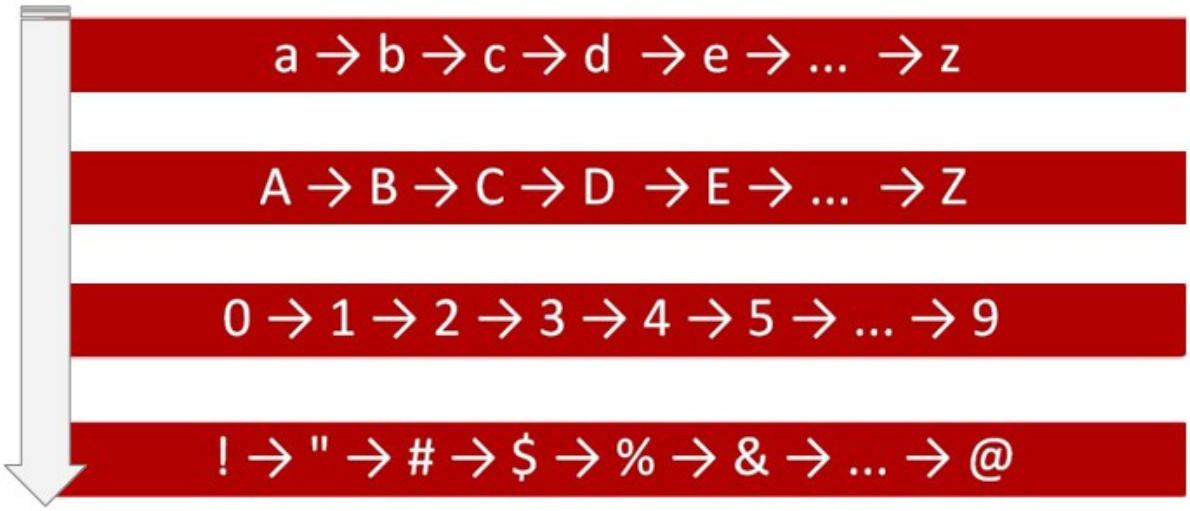
Gli **attacchi di forza bruta** generano e testano tutti i possibili valori di una password.
Questa strategia è l'unica che garantisce per certo di trovare la password di un utente, anche se in maniera poco efficiente. È infatti stimato che per password di una determinata lunghezza e complessità, un supercomputer impiegherebbe il tempo necessario al mondo per terminare.

Vediamo un esempio di pseudo-codice:

```
password_found = false
password_length = 1

while password_found == false
do
    while can_create_password_of_length(password_length)
    do
        password = create_password_of_length(password_length)
        if (hash(password) match attacked_hash)
        then
            password_found = true
    done
```

Per trovare una password di lunghezza sconosciuta, un algoritmo deve ciclicamente testare ogni carattere minuscolo, maiuscolo, numeri, simboli



Se la password non viene trovata, si aumenta la lunghezza.
Si stabilisce un valore fisso per il primo carattere e si ripercorrono gli step visti prima.

Se la password non viene ancora trovata si cambia il valore del primo carattere e si ripercorre il processo. Questa operazione si ripete per ogni valore possibile del primo carattere.

Si continua così, ciclicamente, fino a che non si trova la password valida.

Una cosa importante da sapere è che avendo a disposizione abbastanza tempo,
un attacco di forza bruta ha sempre successo.

Gli attacchi alle password con forza bruta, tuttavia, sono particolarmente efficaci se la password scelta dall'utente è debole (ovvero breve e composta di sole lettere, senza numeri o simboli).
Diversamente, come abbiamo detto, di fronte ad una password molto complessa potrebbero servire molti anni anche al miglior super computer per trovare la combinazione corretta

Provare manualmente le combinazioni di password non è una strategia efficace.
Ci sono dei tool che automatizzano le richieste come **John the Ripper**.
John the Ripper → tool di password cracking piuttosto popolare scritto per i sistemi operativi basati su Unix.

Fa uso della parallelizzazione dei task per ridurre i tempi di cracking durante una sessione brute force, ed è altamente configurabile (si possono specificare gli insiemi / sottoinsiemi di caratteri da utilizzare come lettere o numeri).

Per fare password cracking John the Ripper ha bisogno che il file delle password (contenente gli utenti) e il file con gli hash delle password siano in un unico file. Sui sistemi Linux questi due file sono:

- /etc/passwd

```
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
_apt:x:102:65534::/nonexistent:/usr/sbin/nologin
mysql:x:103:110:MySQL Server,,,:/nonexistent:/bin/false
tss:x:104:111:TPM software stack,,,:/var/lib/tpm:/bin/false
strongswan:x:105:65534::/var/lib/strongswan:/usr/sbin/nologin
systemd-timesync:x:106:112:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
redsocks:x:107:113::/var/run/redsocks:/usr/sbin/nologin
rwhod:x:108:65534::/var/spool/rwho:/usr/sbin/nologin
iodine:x:109:65534::/run/iodine:/usr/sbin/nologin
messagebus:x:110:114::/nonexistent:/usr/sbin/nologin
miredo:x:111:65534::/var/run/miredo:/usr/sbin/nologin
_rpc:x:112:65534::/run/rpcbind:/usr/sbin/nologin
usbmux:x:113:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
tcpdump:x:114:120::/nonexistent:/usr/sbin/nologin
rtkit:x:115:121:RealtimeKit,,,:/proc:/usr/sbin/nologin
sshd:x:116:65534::/run/sshd:/usr/sbin/nologin
dnsmasq:x:117:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
statd:x:118:65534::/var/lib/nfs:/usr/sbin/nologin
avahi:x:119:125:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
nm-openvpn:x:120:126:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
stunnel4:x:121:127::/var/run/stunnel4:/usr/sbin/nologin
nm-openconnect:x:122:128:NetworkManager OpenConnect plugin,,,:/var/lib/NetworkManager:/usr/sbin/nologin
Debian-snmp:x:123:129::/var/lib/snmp:/bin/false
```

/etc/shadow

Per unire questi due file è possibile utilizzare l'utility unshadow, parte di John the Ripper. Il comando per unire il file e redirigere l'output verso un terzo file chiamato hashes è il seguente:

```
unshadow password_file shadow_file > hashes.txt
```

Dove «password_file» e «shadow_file» sono i due file che abbiamo visto nella slide precedente che contengono rispettivamente gli utenti attivi sul sistema e gli hash delle password.

Generalmente, il file delle password contiene informazioni su più utenti.
Nel caso dovesse servire recuperare la password di un determinato utente,
si può specificare con il parametro –users.
Per lanciare un attacco di brute force, si usa la sintassi di seguito:

```
john –incremental --users= «nome_utente» «file_unito»
```

Dove:
«nome utente», va sostituito con l'utente per il quale si vuole recuperare la password
«file_unito», va sostituito con il file di output del comando unshadow che abbiamo visto prima

Al termine di una sessione di brute force con John the Ripper (spesso abbreviato con JtR) si possono controllare le password recuperate con il comando --show, come di seguito:

```
john --show <file_unito>
john --show --format=Raw-MD5<file_unito>
```

Dove <file_unito> è nuovamente il file di output del comando **unshadow**

Nota: la maggior parte delle sessioni di JtR può durare ore o giorni. Durante una sessione si può premere un tasto qualsiasi per controllare la percentuale di avanzamento di JtR.
Per password molto lunghe o particolarmente complesse JtR non è consigliato, in quanto i tempi di cracking potrebbero essere troppo dilatati.

Gli attacchi a dizionario non generano tutte le password possibili, ma piuttosto **utilizzano una lista** (dizionario) di password comuni, testando ogni voce che essa contiene.

Gli attacchi a dizionario sono certamente più veloci di quelli di forza bruta, anche perché un dizionario è sicuramente più piccolo del numero di possibili password per un dato account.

Esempio: il numero di password comuni di lunghezza 8 caratteri, frequentemente utilizzati dagli utenti di sistemi ed applicazioni è poco superiore a 3 milioni di unità, mentre il numero di password possibili che contengono solo lettere minuscole è circa 208 miliardi. Questo ci fa capire la differenza tra gli ordini di grandezza delle due tecniche.

Per eseguire un attacco a dizionario servono:

- Un file delle password da craccare
- Un dizionario, o una wordlist, di password
- Uno strumento che testi ogni parola del dizionario sul file delle password

Si possono lanciare attacchi a dizionario con John the Ripper, utilizzando l'opzione da riga di comando --wordlist per specificare il file delle password da utilizzare. Il comando per lanciare un attacco a dizionario è come in figura sotto:

```
john --wordlist=/usr/share/john/wordlist.txt file_unito
```

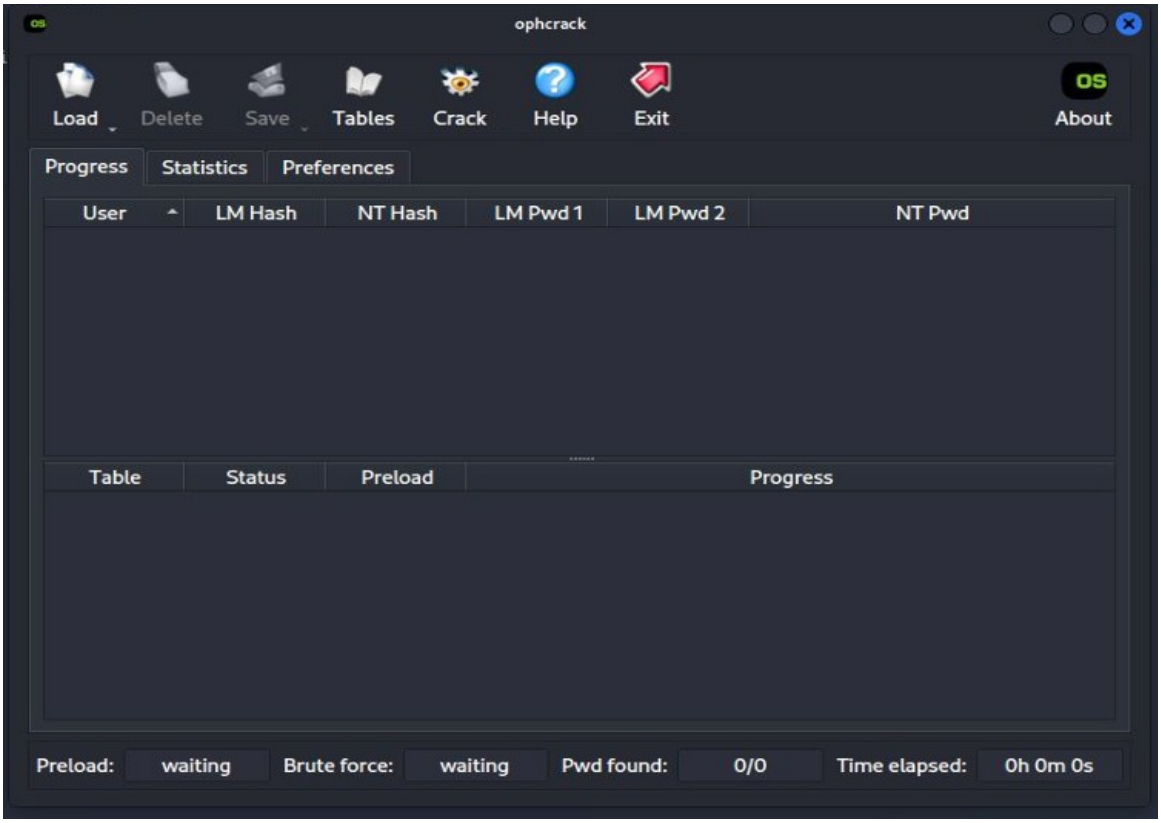
rainbow table → è una tabella che contiene dei collegamenti tra i risultati dell'esecuzione di diversi hash.

Lo spazio necessario a salvare una rainbow table dipende dalla lunghezza massima delle password e da che tipo di caratteri si possono utilizzare.
Generalmente, le tabelle variano da centinaia di MB fino a centinaia di GB.
Il grosso limite delle rainbow table è lo spazio necessario per garantire delle sessioni di cracking di successo.

Per esempio: una rainbow table che garantisce una percentuale di successo di circa il 95% di craccare una password di lunghezza da 1 a 9 caratteri il cui hash è stato ottenuto con MD5, occupa circa 900 GB.

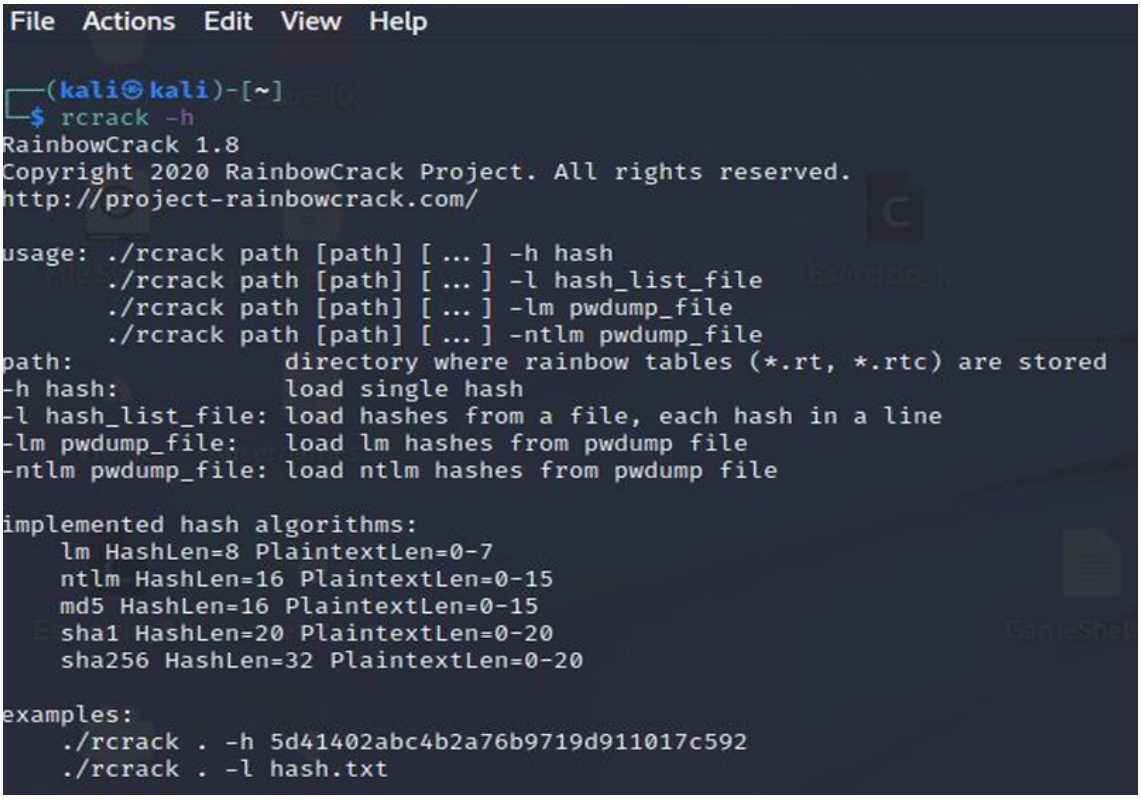
Visto così potremmo dire che le rainbow table sono ottimi strumenti per craccare password sia semplici che complesse di una lunghezza moderata, ad esempio da 5 ad 8 caratteri.

Un ottimo tool piuttosto utilizzato per fare cracking delle password di Windows tramite rainbow table è ophcrack. Sul sito ufficiale si possono trovare le rainbow table gratuite la cui dimensione varia da poche centinaia di MB fino a diversi TB. N.B. ophcrack è limitato, può solamente craccare le password di Windows



Cliccando sul tab «Tables» si possono installare delle tabelle gratuite o a pagamento. Non è fondamentale installarle tutte, concentriamoci su quelle gratuite. Il file delle password si carica nel tool tramite il tasto «Load». Una volta scelto il file delle password, ed installate le tabelle, non ci resta che cliccare su «Crack» per avviare il processo.

Un altro tool per craccare le password via rainbow table senza limitazioni di sistema operativo è «rainbow crack» (rcrack). Potete scaricarlo sulla vostra Kali Linux per vederlo all'opera. Cyber Security & Ethical Hacking Gli attacchi alle password La sintassi base del tool è piuttosto semplice, potete notare alcuni esempi dell'utilizzo del tool qui.



I buffer overflow & lo stack

La vulnerabilità buffer overflow vengono sfruttate da vari attacchi che ottengono il controllo sul flusso di esecuzione di un programma o una routine del sistema operativo. Controllare l'esecuzione di un programma significa essere in grado di fargli fare qualcosa di differente rispetto alla logica stabilita dal programmatore.

Un attacco che sfrutta un buffer overflow può:

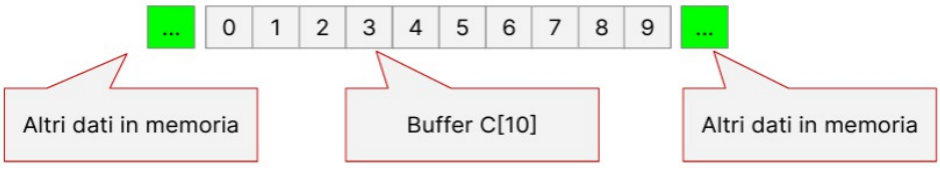
- Causare il crash di un programma o dell'intero sistema operativo
- Scatenare un secondo attacco di tipo privilege escalation
- Ottenere la possibilità di eseguire codice malevolo direttamente sulla macchina vittima
- Aggirare le funzionalità di sicurezza di un sistema operativo

Per capire cos'è una vulnerabilità di tipo buffer overflow, dobbiamo capire cos'è un buffer.

Un buffer è un'area di memoria che risiede in RAM riservata per contenere dei dati temporanei, come:

- Un input utente
- Una parte di un file video
- Il banner dei server ricevuti da una web app
- Altro.

I buffer hanno una dimensione finita, ossia possono contenere un certo quantitativo di dati. Per farvi un esempio pratico, vi ricordate la dichiarazione di un vettore in C o C++? Ai vettori viene associata una grandezza tra parentesi quadre che indica la dimensione del vettore. Ad esempio: INT C[10], definisce un vettore di 10 interi.

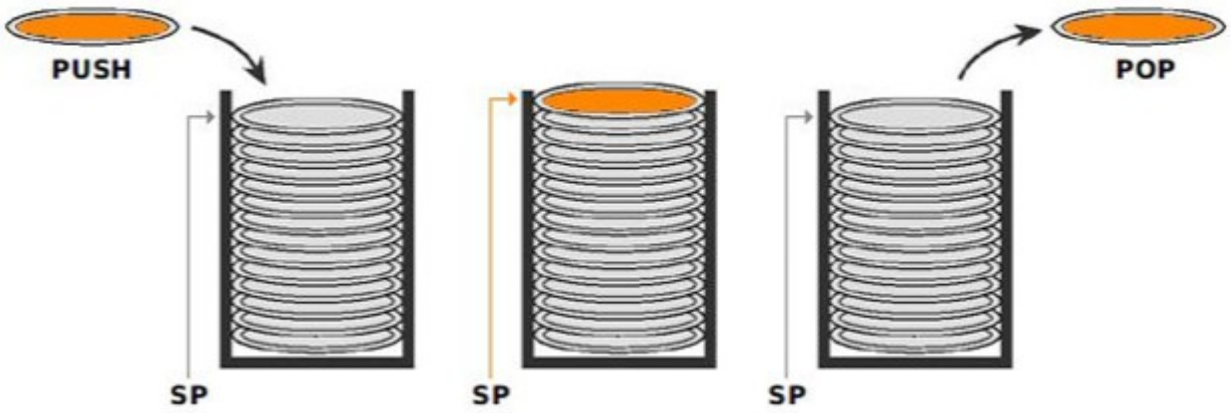


Se lo sviluppatore di un'applicazione non impone strettamente dei limiti ai buffer, un attaccante potrebbe trovare un modo per scrivere dei dati oltre questi limiti, scrivendo codice arbitrario nella memoria del computer. **Con un codice arbitrario particolare un attaccante potrebbe arrivare a controllare il flusso di esecuzione del programma.**

una vulnerabilità buffer overflow sfrutta proprio la mancanza di controllo dell'input in una determinata porzione di memoria.

Di conseguenza, un potenziale attaccante potrebbe inserire 100 interi all'interno di un buffer di dimensione 10, di fatto andando a sovrascrivere i dati contigui al buffer in memoria.

Durante l'esecuzione di un programma i dati e le variabili, così come i buffer, vengono salvati in una struttura chiamata **stack**.
Lo stack può essere immaginato come una pila di piatti, dove si può solo aggiungere e rimuovere un piatto alla volta dalla cima. L'azione per aggiungere un piatto sulla cima della pila, viene chiamata «PUSH» mentre l'azione che rimuove un piatto dalla cima viene detta «POP». Questo approccio prende il nome di LIFO (last in first out), ovvero l'ultimo piatto inserito sulla cima è il primo ad essere rimosso.



Se un dato programma deve salvare un array di interi, può allocare direttamente tre posizioni sullo stack. Il codice in C sarebbe :

Int C [5] = {1,2,3,4,5};

C[0]	1
C[1]	2
C[2]	3
C[3]	4
C[4]	5

Se lo stesso programma volesse poi allocare un ulteriore array, chiamiamolo B

Int B [3] = {11,22,33}

Ci accorgiamo come lo stack cresce dal basso verso l'alto (approccio bottom-up).

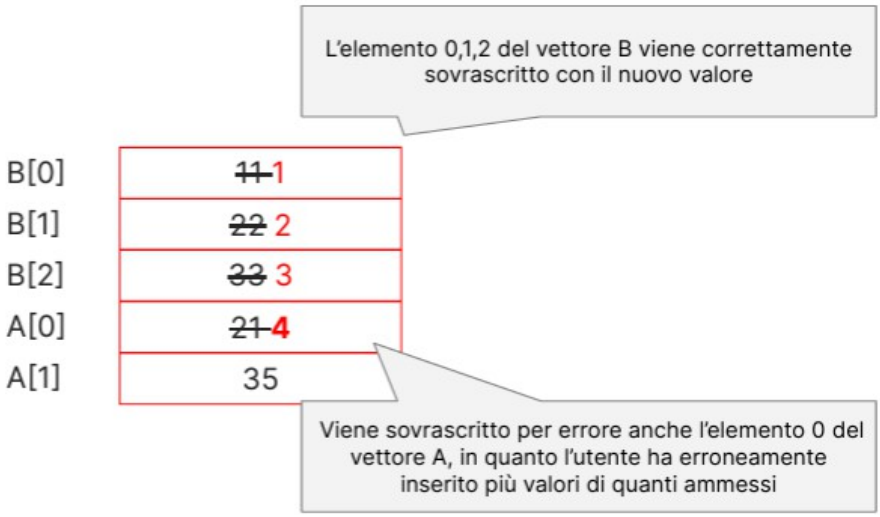
>Cosa succederebbe se un attaccante scrivesse 5 numeri nel vettore B, che ha spazio solo per 3 valori?

B[0]	11
B[1]	22
B[2]	33
C[0]	1
C[1]	2
C[2]	3
C[3]	4
C[4]	5

Finirebbe per sovrascrivere 2 elementi del vettore C, che è contiguo in memoria. Finirebbe quindi per modificare il vettore C

B[0]	11
B[1]	22
B[2]	33
C[0]	**1**
C[1]	**2**
C[2]	3
C[3]	4
C[4]	5

```
int A [2] = {4,35}
int B [3] = {1,2,3}
```



Un utente lecito che vorrà leggere il valore A, troverà un valore diverso da quello reale, in quanto A[0] è stato modificato a cause dell'overflow di B.

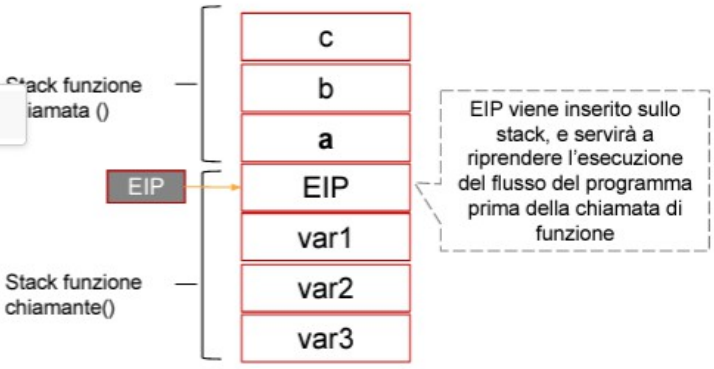
Lo stack utilizzato dalle applicazioni e dai sistemi operativi, a differenza di quello appena visto, non contiene solo variabili e dati, ma anche delle informazioni circa il flusso di esecuzione dei programmi.

Nella fattispecie,
lo stack contiene l'indirizzo di memoria al quale ritornare per proseguire con l'esecuzione del chiamante
Per ogni funzione chiamata all'interno di un programma viene creato uno stack, dove sono salvate localmente le variabili che verranno utilizzate da quella particolare funzione.

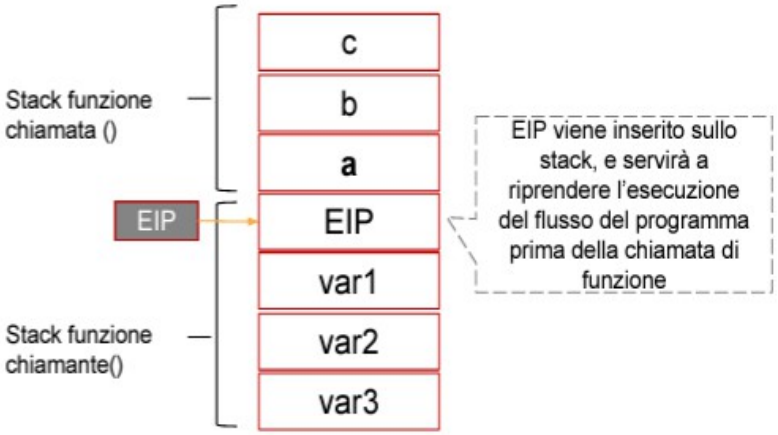
Quando una funzione ne chiama una seconda per utilizzare una particolare funzionalità, **blocca la sua esecuzione e salva sullo stack all'istruzione corrente (EIP, Extended Instruction Pointer)**, prima di creare un nuovo stack per la funzione chiamata.

Copy text

A cosa serve l'EIP? **L'EIP serve per tornare al punto dove la funzione chiamante ha lasciato l'esecuzione prima di chiamare la seconda funzione**



Perché è così importante l'EIP?
Il puntatore all'istruzione corrente salvato sullo stack al momento della chiamata di funzione, contiene l'indirizzo della prossima istruzione da eseguire una volta che la funzione chiamata ha terminato il suo lavoro. Ciò significa che se un attaccante riesce a modificare l'EIP con un indirizzo a suo piacimento può modificare il flusso del programma, ovvero può fare in modo che il programma esegua del codice magari scritto dall'attaccante.



W14D1 - Pratica (1) → Password Cracking

Traccia:

Abbiamo visto come sfruttare un attacco SQL injection per recuperare le password degli utenti di un determinato sistema.

Se guardiamo meglio alle password trovate, non hanno l'aspetto di password in chiaro, ma sembrano più hash di password MD5.
Recuperate le password dal DB come visto, e provate ad eseguire delle sessioni di cracking sulla password per recuperare la loro versione in chiaro.

Sentitevi liberi di utilizzare qualsiasi dei tool visti nella lezione teorica.

L'obiettivo dell'esercizio di oggi è craccare tutte le password trovate precedentemente.
Consegna:

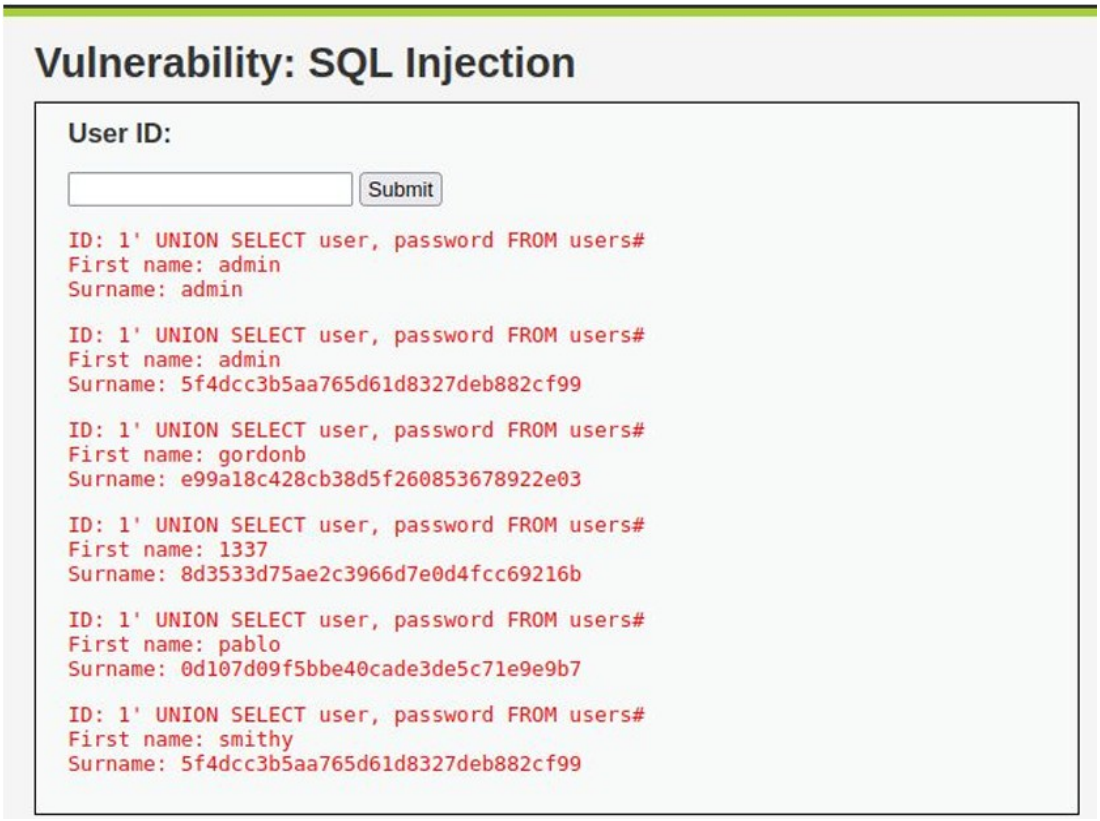
1. Screenshot dell'SQL injection già effettuata
2. Due righe di spiegazione di cos'è questo cracking (quale tipologia / quale meccanismo sfrutta)
3. Screenshot dell'esecuzione del cracking e del risultato

Soluzione

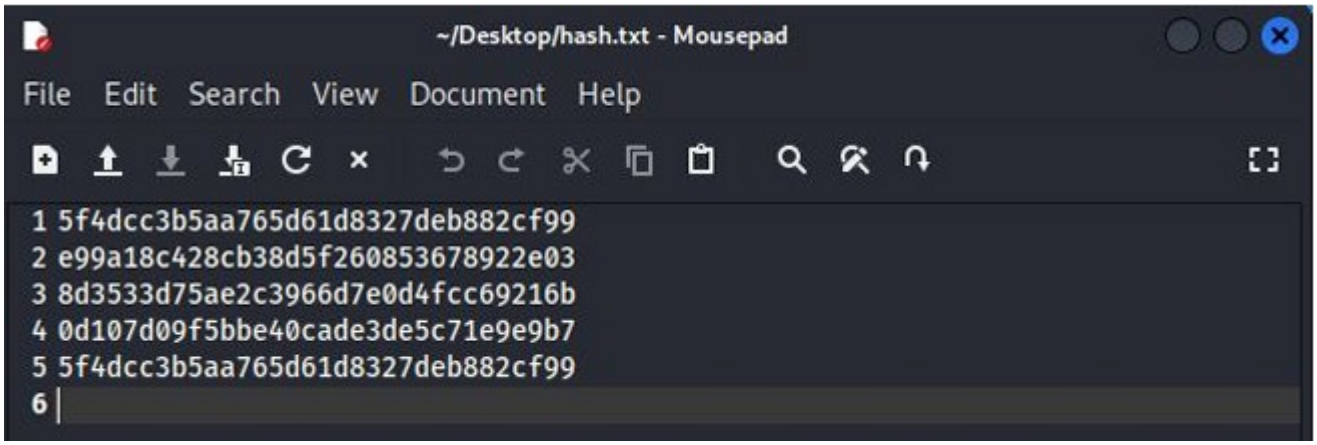
Le password da craccare sono le seguenti:

- 5f4dcc3b5aa765d61d8327deb882cf99
- e99a18c428cb38d5f260853678922e03
- 8d3533d75ae2c3966d7e0d4fcc69216b
- 0d107d09f5bbe40cade3de5c71e9e9b7
- 5f4dcc3b5aa765d61d8327deb882cf99

Vediamo che la prima stringa è uguale all'ultima, quindi ci aspettiamo la stessa password per la prima riga e la quinta riga



Per craccare gli hash delle password MD5 utilizziamo JtR (John the Ripper). La prima cosa da fare è creare un file hash.txt dove andiamo ad inserire tutti gli hash delle password.



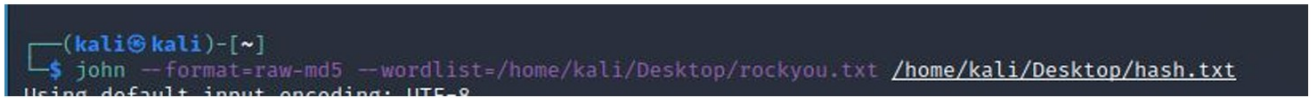
Soluzione

Il comando da eseguire utilizzando JtR è il seguente:

```
john --format=raw-md5 --wordlist=/home/kali/Desktop/rockyou.txt /home/kali/Desktop/hash.txt
```

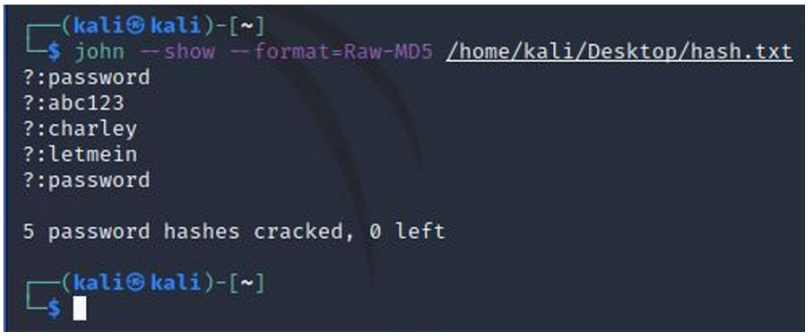
dove:

- Format, specifica il tipo di hash da craccare
- --wordlist, specifica il dizionario da utilizzare per craccare le password
- /home/kali/Desktop/hash.txt, specifica il file con gli hash delle password da craccare

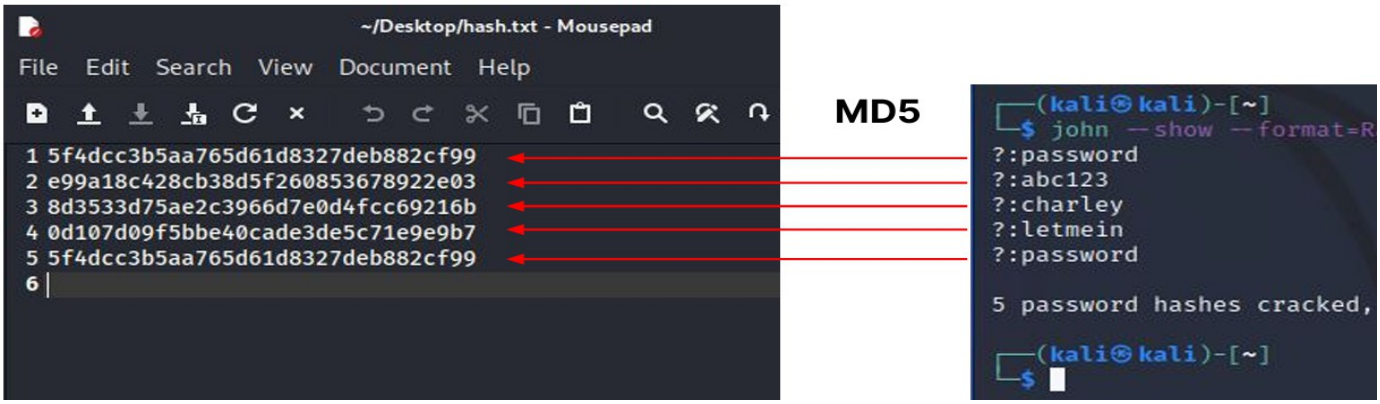


Una volta terminata la sessione di cracking, possiamo controllare le password craccate da JtR con il comando in figura

Potete anche vedere le password in chiaro craccate. Come ci aspettavamo il primo record è uguale all'ultimo



Se volete fare una controprova, provate ad utilizzare un generatore MD5 partendo dalla password in chiaro. Dovreste ricevere esattamente i risultati nel file hash.txt



W14D1 - Pratica (2) → Infezione malware

Traccia:
Hai appena scoperto che l'azienda che segui come consulente di sicurezza ha un computer con Windows 7 è stato infettato dal malware WannaCry. Cosa fai per mettere in sicurezza il tuo sistema?

Consegna:
• Per prima cosa occorre intervenire tempestivamente sul sistema infetto
• In seguito, preparare un elenco delle varie possibilità di messa in sicurezza del sistema
• Per ogni possibilità valutare i pro e i contro

Aggiorna il sistema operativo: L'attacco di WannaCry ha sfruttato una vulnerabilità presente in molte versioni di Windows, compresa Windows 7.
Per proteggere il tuo computer, devi assicurarti di avere tutti gli aggiornamenti di sicurezza più recenti installati.
Segui questi passaggi: V
ai al menu Start e seleziona "Control Panel"
Clicca su "System and Security" e poi su "Windows Update"
Clicca su "Check for updates" per verificare la disponibilità degli aggiornamenti
Clicca su "Install Updates" per installare tutti gli aggiornamenti di sicurezza necessari.

Disabilita il protocollo SMBv1: WannaCry ha sfruttato una vulnerabilità presente nella versione obsoleta del protocollo SMB, ovvero il SMBv1. Disabilitando questa funzionalità, è possibile ridurre il rischio di future infezioni.
Ecco come fare:
Vai al menu Start e cerca "Turn Windows features on or off"
Scopri la voce "SMB 1.0/CIFS File Sharing Support" e deselezionala.
Clicca su "OK" e riavvia il computer.

Scansiona il sistema con un software anti-malware: Anche se hai seguito i primi due passaggi, potrebbe esserci ancora del malware nel tuo sistema. Utilizzare un software anti-malware affidabile per eseguire una scansione completa.
Scopri i seguenti passaggi:
Scarica ed esegui un software anti-malware come Malwarebytes o Avast.
Clicca su "Scan Now" per avviare la scansione.
Segui le istruzioni visualizzate sullo schermo per rimuovere eventuali minacce trovate

Modifica le password: WannaCry ha sfruttato le credenziali di accesso deboli per diffondersi nelle reti aziendali. Per evitare la futura diffusione di malware tramite le tue credenziali, modifica le password dei tuoi account.
Vai alle impostazioni del tuo account e modifica la tua password.
Usa una password complessa e diversa per ogni account.
Esegui periodicamente la modifica della password.

In conclusione, la soluzione migliore è aggiornare il sistema operativo a Windows 10 o 11. Qualora non fosse possibile, seguire le altre azioni di rimedio. Ovviamente, il primo passaggio è mettere in sicurezza la rete isolando il computer (staccando il cavo di rete oppure disattivando il wifi) ed effettuare la scansione antimalware