Lab 02

PILL Box Vulnerability Assessment

Aayush Bulusu

JOT898

Prof. McCulley

March 8, 2024

Table of Contents

Executive Summary

According to TechTarget, vulnerability assessment is described as a process that involves identifying, defining, and prioritizing vulnerabilities in computer applications and networks as a way of gaining knowledge and awareness to properly react to threats. Vulnerability assessments are generally conducted using a variety of tools, interfaces, or environments to generate thorough insights into the present issues. Risk assessment or analysis, according to CrowdStrike, is a term used in business contexts that describes the processes and methodologies used to identify the likelihood of security events and the degree of impact on assets. Such assessments provide the company with a more accurate and improved understanding of the company's assets and applications, identify weaknesses across devices and ecosystems, and can be used with vulnerability assessments to provide optimization of company resources, reduce potential damage costs, ensure compliance with regulations, and improve availability and reduce downtime. This lab focused on vulnerability assessment and risk analysis using Kali Linux and required me to conduct assessments using Kali tools over the PILL network IP range 192.168.101.150/205. In this lab, I used two tools: Nmap and Sparta. I used Nmap to conduct a vulnerability analysis and saved it into a text file to easy retrieval, while Sparta generated a risk assessment on the range. The following report documents my process and thinking over the course of the lab and provides an analysis of the reports generated by the Nmap and Sparta tools.

Vulnerability Assessment

My first responsibility was to run a basic scan on the .150-.205 range of the Kali virtual machine. First, I initiated a basic ping command to check for connectivity in Figure 1a, where I pinged the IP address 192. 168.101.176 once as directed. The ping command’s output showed that I was on the correct network. Next, to check for vulnerabilities, I performed the Nmap command shown in figure 1b. Nmap is a commonly used tool in network security that creates a network map and is helpful for multiple reasons. Among the many reasons for nap being effective for vulnerability and risk assessment are that Nmap allows users to gain a much more in-depth understanding of the network by showing active devices connected to the network, identifies open and closed ports, and reports protocol or service versions. Moreover, Nmap is also an open-source tool that can be accessed through the standard Kali terminal, removing the need to download the service from websites.

In the same figure, the Nmap command scanned the IP range, returned the output, and saved it into a text file for easy retrieval. In the report, Nmap displayed all of the open ports along with vulnerability and risk status. For example, in figure 1c, Nmap reported a vulnerable state and high-risk factor for the IP 192.168.101.156, stating that the IP had three open ports: 135,139, and 445, and contained a remote code execution vulnerability regarding the SMBv1 servers. According to Microsoft, an attacker can take advantage of this vulnerability by sending messages to a Microsoft SMB 1.0 server and perform remote code execution. An easy fix to this issue would be simply updating the SMB protocol to the latest version to patch how SMB handles requests from hackers. Next, Nmap also returned a scan output for 192.168.101.192. In figure 1d, the Nmap scan analyzed the network and classified it as “likely vulnerable” due to CVE-2007-6750. According to NOST, this common vulnerability allows attackers to take advantage of

Apache's HTTP Server 1.x and 2.x and cause a denial of service, or daemon outage through partial HTTP requests as shown by Slowloris. In the figure, Nmap's output stated that the vulnerability came from Slowloris attempting to maintain as many connections to the target web server open and for as long as possible, causing it to starve the HTTP's resources and resulting in a greater likelihood of a denial of service. However, not all IP addresses returned vulnerabilities. For example, in figure 1e, 192.168.101.203 found port 1947 as the only open port and returned no CSRF, DOM based cross-site scripting, and HTTP-stored cross site script vulnerabilities. In fact, most of the scanned IP's were found to be secure. Of the 56 IP addresses, Nmap identified 15 open ports, which will always remain a vulnerability regardless of level of security, as they provide direct connection to the network.

Risk Analysis

The next phase of the lab was to perform a risk analysis using a separate tool on the same IP range. For this section, I chose the Sparta tool. In Kali Linux, Sparta is also a highly popular tool used by security analysts to conduct risk assessments. Typically used by penetration testers, Sparta does not require knowledge of Linux commands. Sparta automates the scanning process by only requiring the user to enter the network range for scanning. Sparta will generate a command on its own based on input and requires no extra effort from the users. Unlike Nmap, Sparta is an application and has a graphical user interface, or GUI, making it easier to use compared to Nmap's command lines since Sparta consolidates all its features into boxes and tabs that are easy to navigate through. Sparta is regularly used by penetration testers to identify risks and help them conduct penetration test attacks. Moreover, Sparta's uniqueness comes in the form of its integration. While Nmap is a single tool, Sparta effectiveness comes from being a suite of different tools that are also used by many security analysts. For example, in the lab, the Sparta scan that I initiated generated a risk assessment report based on the findings from Nmap, Nikto, and Hydra, increasing its effectiveness through the use of a greater surface of assessments.

First, after launching Sparta, I was greeted with the interface prompting an input for the target IP. After clicking the blank space and entering the range 192.168.101.150/205, I ran the scan. In figures 2a and 2b, Sparta showed the output for 192.168.101.156. First, it identified the operating system as a Windows computer, then the open ports as 135,137,139, and 445. The last tab shows that Sparta ran Nmap to brute-force passwords and continued with the Hydra tool for further analysis. While there were no passwords for it to retrieve, it still reflects Sparta's ability to make use of numerous tools to perform risk assessments. Sparta also classifies all identified ports into three risk categories: low (green), moderate (orange), and high (red). The identified ports in figure 2b were all classified as low risk for that specific IP that Sparta scanned, meaning that all open ports on that interface were relatively secure and not vulnerable.

In figure 2c, Sparta identified the interface's operating system as a Linux host, with four open ports identified: 22, 137, 139, and 445. It also displayed the port protocol for each port. For example, port 22 was a TCP SSH, or secure shell. 13 was the UDP-based NetBIOS NS, and 139 and 445 were TCPWrapped.

Figure 2d provides a different view of the output, In the figure, Sparta allowed me to search for the output based on the tools and scripts it ran to perform the scan and identify vulnerabilities. For example, clicking on Nikto showed that it targeted two IP addresses, reported its port, and gave a lengthy description on the results. It showed that while the risk for the first IP address was low, it identified a few risks in the form of the lack of headers. The Cross Site Scripting header, Strict Transport Security HTTP header, Expert CT header, and X-Content-Type-Options header were not present. The lack of the X-Frame-Option header means that the IP is more vulnerable to clickjacking cyberattacks, which are done to fool users into clicking on objects that they do not wish to select. The lack of a XSS header means that the IPs do not have the protection of filtering out cross-site-scripting attacks. The IP not having a Strict Transport Security header makes it much more prone to man-in-the-middle attacks. Expect-CT headers assist in validating certificates. Without this, attackers could get by with fake certificates and steal data.

Figures 2e and 2f display the Services tab content in Sparta. This tab showed all the services that Sparta scanned and was a good way to filter through the different services to determine risk. One reason this tab is helpful for analysts is that it provides an easy method of filtering data and identifying issues, or troubleshooting. Analysts can find specific vulnerabilities and risks just by selecting the specific service and comb through ports. For example, in figure 2e, I set the filter to SSH to identify all the IP addresses that were identified with having port 22. Because SSH is a service used for remote administration, identifying any risks regarding SSH can be helpful in mitigating potential attacks. This is especially helpful in identifying any secure shells with misconfigurations and potential attack vectors such as privilege escalation. Figure 2f shows that I filtered for all services using TCPWrapped. TCPWrapped appears on Nmap scans when a port is open. However, after receiving the transfer, it closes possibly due to it being configured through firewall rules. TCPWrapped is an example of a low-risk low-vulnerability mechanism when performing scans such as the one I performed. Because it is configured with such security considerations in mind, TCPWrapped ports will generally be very secure. As the figure shows, all of the ports scanned through the TCPWrapped service are green, or low risk. Aside from TCPWrapped and SSH, Sparta found no risks. All of the IP addresses and the ports from each interface that Sparta scanned were green, meaning that no critical risks were found suggesting that the IP range was secure enough to be considered safe by Sparta.

Nmap and Sparta Assessment

Based on the outputs generated by both tools, Nmap and Sparta provided a thorough and informative analysis on the risk and vulnerabilities present on the IP address range 192.168.101.150/205. Despite this, they offered differing assessments of the network. For example, Nmap was primarily used for vulnerability assessments. It performed a network scan and identified service versions, ports, and risks. On the other hand, Sparta was a more comprehensive assessment as it used Nmap along with other tools such as Nikto and Hydra. Nmap identified the SMBv1 and HTTP Slowloris vulnerabilities and classified their risks accordingly. Sparta provided a GUI-based analysis of all the ports and allowed filtering based on tools and services related to the scan. It classified the risks on each address based on the three risk colors unlike the text-based output that Nmap generated. Moreover, Sparta's assessment was easier to analyze due to the interface and the multitude of tools it used during the scan.

Conclusion

This lab served as a continuation of vulnerability analysis from the previous lab. However, this lab also placed emphasis on risk analysis. Based on the results, the scans identified all of the ports from each address and classified the risks and vulnerabilities in an organized manner. Such assessments are important for security analysts as they provide invaluable insights into the risks and vulnerabilities present in a network as well as minimize the potential of cyberattacks. The second lab was an important introduction into risk analysis and vulnerability assessment.

Moreover, it prepared me better by allowing me to understand the responsibilities of security analysts in regard to risk assessments and vulnerability analysis.

References

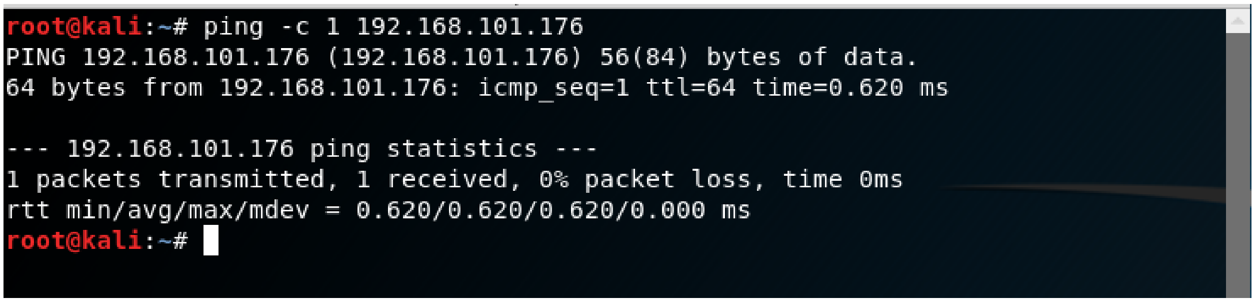
CVE-2007-6750 Detail. NVD. (n.d.). https://nvd.nist.gov/vuln/detail/cve-2007-6750

Microsoft Security bulletin MS17-010 - critical. Microsoft Learn. (n.d.). https://learn.microsoft.com/en-us/security-updates/securitybulletins/2017/ms17-010

Nagarajan, J. (n.d.). *How to perform a cybersecurity risk assessment*. CrowdStrike. https://www.crowdstrike.com/en-us/cybersecurity-101/advisory-services/cybersecurity-risk-assessment/

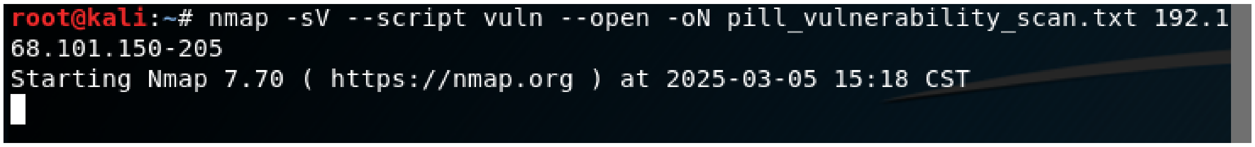
TechTarget. (2024, March 11). *What is a vulnerability assessment (vulnerability analysis)? definition from searchsecurity*. Search Security. https://www.techtarget.com/searchsecurity/definition/vulnerability-assessment-vulnerability-analysis

Appendix



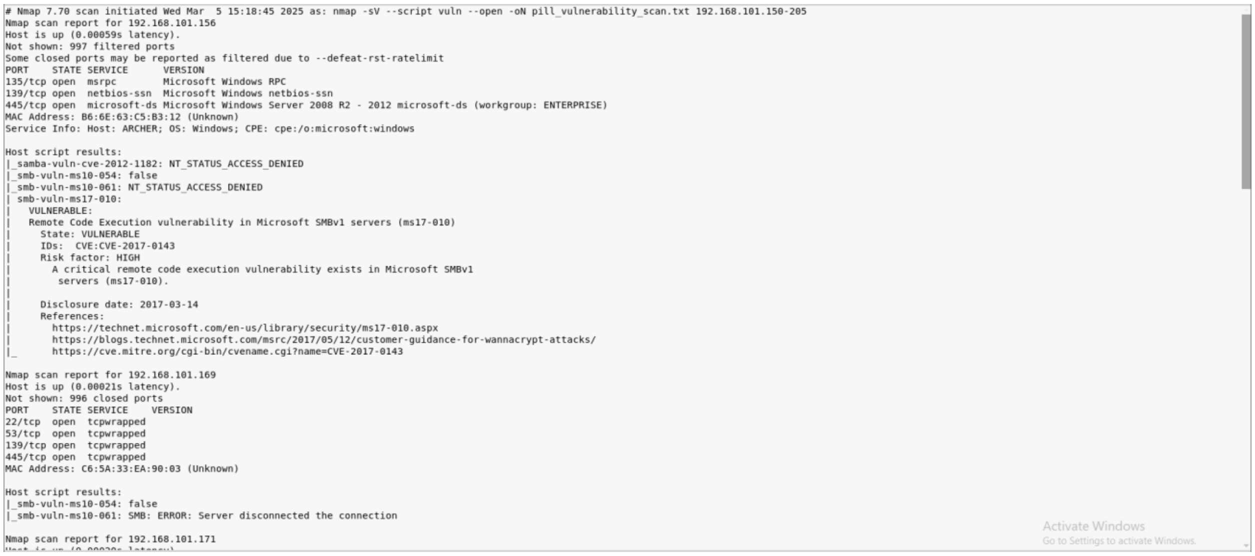
```
root@kali:~# ping -c 1 192.168.101.176
PING 192.168.101.176 (192.168.101.176) 56(84) bytes of data.
64 bytes from 192.168.101.176: icmp_seq=1 ttl=64 time=0.620 ms

--- 192.168.101.176 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.620/0.620/0.620/0.000 ms
root@kali:~#
```

Figure 1a: Basic ping command to the range



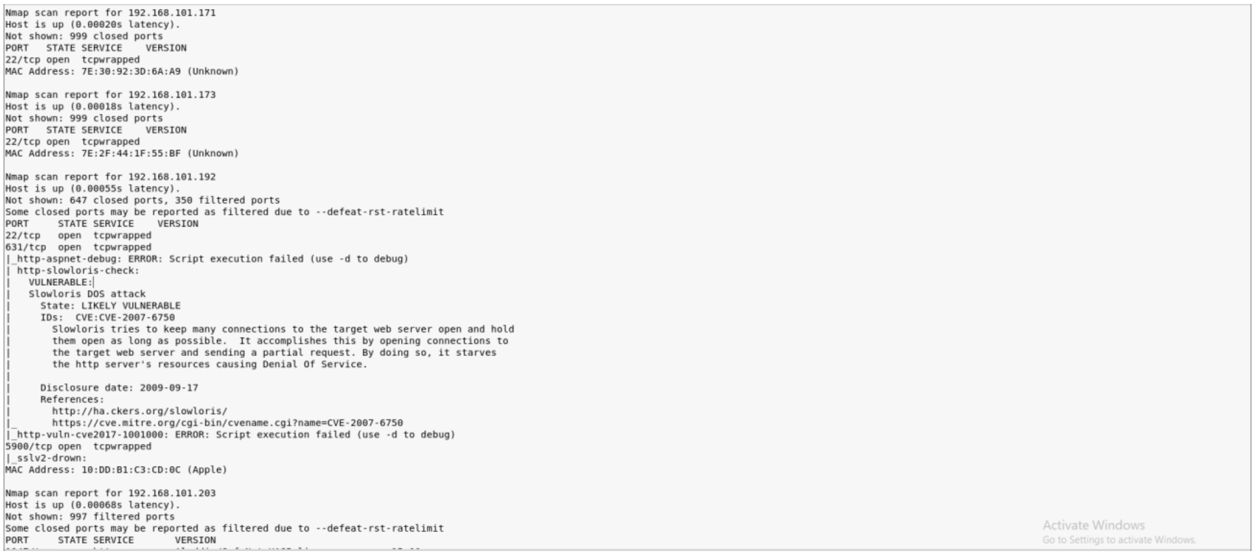
```
root@kali:~# nmap -sV --script vuln --open -oN pill_vulnerability_scan.txt 192.1
68.101.150-205
Starting Nmap 7.70 ( https://nmap.org ) at 2025-03-05 15:18 CST
```

Figure 1b: nmap command to the range



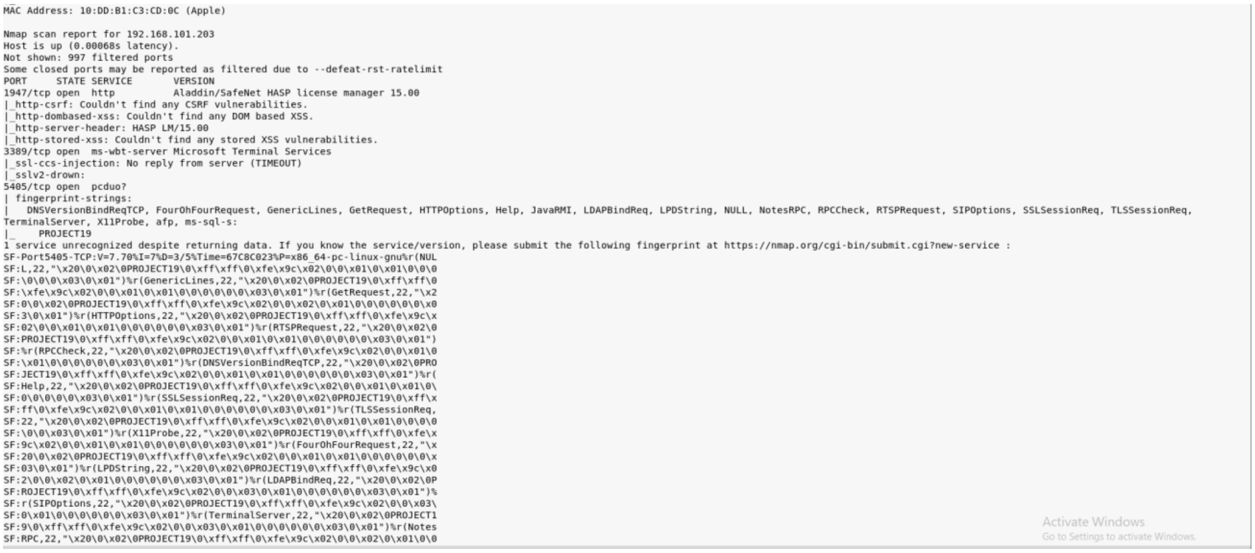
```
# Nmap 7.70 scan initiated Wed Mar  5 15:18:45 2025 as: nmap -sV --script vuln --open -oN pill_vulnerability_scan.txt 192.168.101.150-205
Nmap scan report for 192.168.101.156
Host is up (0.00059s latency).
Not shown: 997 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT    STATE SERVICE      VERSION
135/tcp open  msrpc        Microsoft Windows RPC
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: ENTERPRISE)
MAC Address: B6:6E:63:C5:B3:12 (Unknown)
Service Info: Host: ARCHER; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143

Nmap scan report for 192.168.101.169
Host is up (0.00021s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE    VERSION
22/tcp  open  tcpwrapped
53/tcp  open  tcpwrapped
139/tcp open  tcpwrapped
445/tcp open  tcpwrapped
MAC Address: C6:5A:33:EA:90:03 (Unknown)

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: SMB: ERROR: Server disconnected the connection

Nmap scan report for 192.168.101.171
```

Figure 1c: nmap output



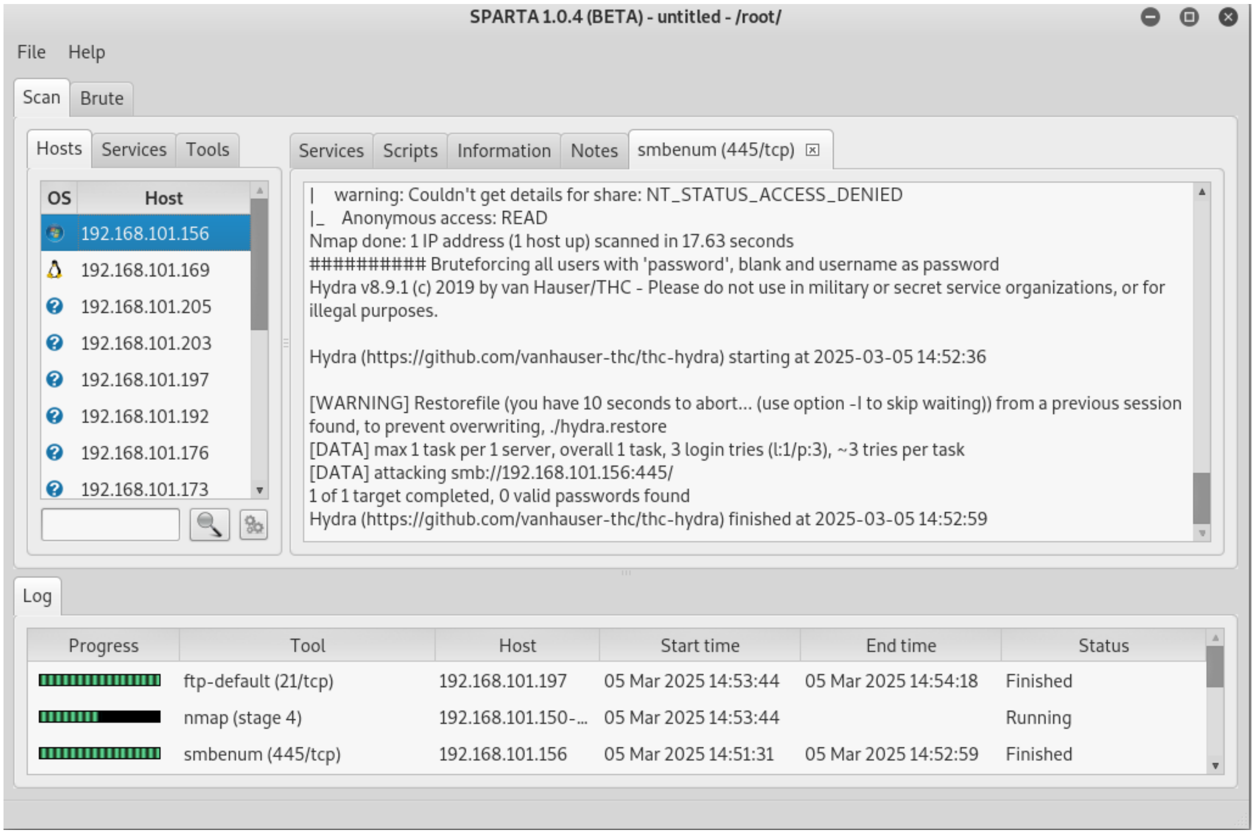
```
Nmap scan report for 192.168.101.171
Host is up (0.00020s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE    VERSION
22/tcp open  tcpwrapped
MAC Address: 7E:30:92:3D:6A:A9 (Unknown)

Nmap scan report for 192.168.101.173
Host is up (0.00018s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE    VERSION
22/tcp open  tcpwrapped
MAC Address: 7E:2F:44:1F:55:BF (Unknown)

Nmap scan report for 192.168.101.192
Host is up (0.00055s latency).
Not shown: 647 closed ports, 350 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE    VERSION
22/tcp   open  tcpwrapped
631/tcp  open  tcpwrapped
|_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       http://ha.ckers.org/slowloris/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)
5900/tcp open  tcpwrapped
|_sslv2-drown:
MAC Address: 10:DD:B1:C3:CD:0C (Apple)

Nmap scan report for 192.168.101.203
Host is up (0.00068s latency).
Not shown: 997 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE       VERSION
```

Figure 1d: nmap output



```
MAC Address: 10:DD:B1:C3:CD:0C (Apple)

Nmap scan report for 192.168.101.203
Host is up (0.00068s latency).
Not shown: 997 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE       VERSION
1947/tcp open  http          Aladdin/SafeNet HASP license manager 15.00
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-server-header: HASP LM/15.00
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-ccs-injection: No reply from server (TIMEOUT)
|_sslv2-drown:
5405/tcp open  pcduo?
| fingerprint-strings:
|   DNSVersionBindReqTCP, FourOhFourRequest, GenericLines, GetRequest, HTTPOptions, Help, JavaRMI, LDAPBindReq, LPDString, NULL, NotesRPC, RPCCheck, RTSPRequest, SIPOptions, SSLSessionReq, TLSSessionReq,
TerminalServer, X11Probe, afp, ms-sql-s:
|_    PROJECT19
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port5405-TCP:V=7.70%I=7%D=3/5%Time=67C8C023%P=x86_64-pc-linux-gnu%r(NUL
SF:L,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0\x01\0\0\0
SF:\0\0\0\x03\0\x01")%r(GenericLines,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0
SF:\xfe\x9c\x02\0\0\x01\0\x01\0\0\0\0\0\0\x03\0\x01")%r(GetRequest,22,"\x2
SF:0\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x02\0\x01\0\0\0\0\0\0\x0
SF:3\0\x01")%r(HTTPOptions,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x
SF:02\0\0\x01\0\x01\0\0\0\0\0\0\x03\0\x01")%r(RTSPRequest,22,"\x20\0\x02\0
SF:PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0\x01\0\0\0\0\0\0\x03\0\x01")
SF:%r(RPCCheck,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0
SF:\x01\0\0\0\0\0\0\x03\0\x01")%r(DNSVersionBindReqTCP,22,"\x20\0\x02\0PRO
SF:JECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0\x01\0\0\0\0\0\0\x03\0\x01")%r(
SF:Help,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0\x01\0\
SF:0\0\0\0\0\x03\0\x01")%r(SSLSessionReq,22,"\x20\0\x02\0PROJECT19\0\xff\x
SF:ff\0\xfe\x9c\x02\0\0\x01\0\x01\0\0\0\0\0\0\x03\0\x01")%r(TLSSessionReq,
SF:22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0\x01\0\0\0\0
SF:\0\0\x03\0\x01")%r(X11Probe,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x
SF:9c\x02\0\0\x01\0\x01\0\0\0\0\0\0\x03\0\x01")%r(FourOhFourRequest,22,"\x
SF:20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x01\0\x01\0\0\0\0\0\0\x
SF:03\0\x01")%r(LPDString,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x0
SF:2\0\0\x02\0\x01\0\0\0\0\0\0\x03\0\x01")%r(LDAPBindReq,22,"\x20\0\x02\0P
SF:ROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x03\0\x01\0\0\0\0\0\0\x03\0\x01")%
SF:r(SIPOptions,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x03\
SF:0\x01\0\0\0\0\0\0\x03\0\x01")%r(TerminalServer,22,"\x20\0\x02\0PROJECT1
SF:9\0\xff\xff\0\xfe\x9c\x02\0\0\x03\0\x01\0\0\0\0\0\0\x03\0\x01")%r(Notes
SF:RPC,22,"\x20\0\x02\0PROJECT19\0\xff\xff\0\xfe\x9c\x02\0\0\x02\0\x01\0\0
```

Figure 1e: nmap output

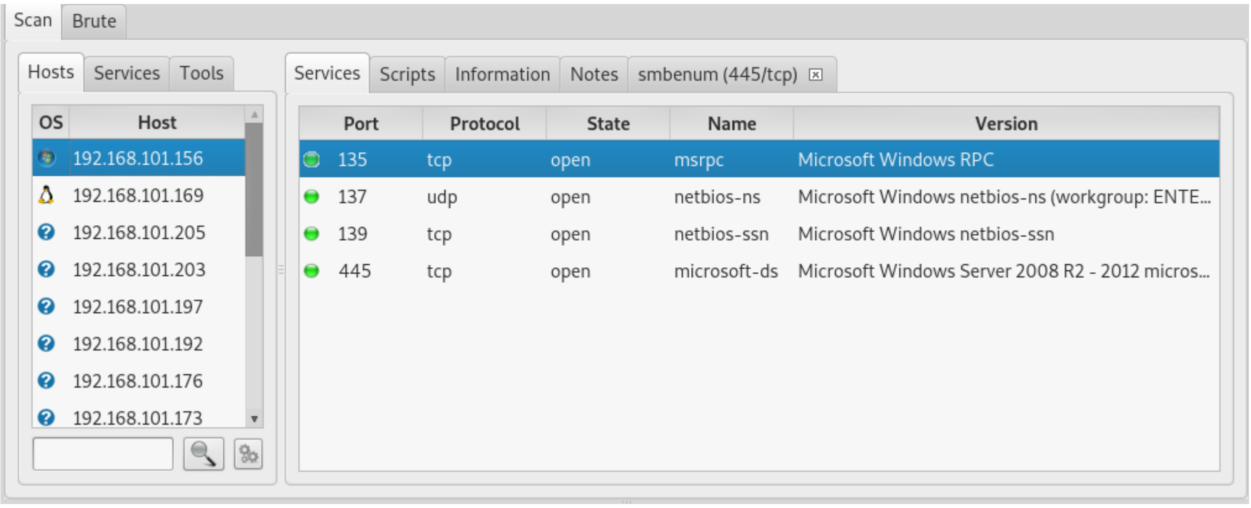


OS	Host
	192.168.101.156
	192.168.101.169
?	192.168.101.205
?	192.168.101.203
?	192.168.101.197
?	192.168.101.192
?	192.168.101.176
?	192.168.101.173

```
|   warning: Couldn't get details for share: NT_STATUS_ACCESS_DENIED
|_   Anonymous access: READ
Nmap done: 1 IP address (1 host up) scanned in 17.63 seconds
########### Bruteforcing all users with 'password', blank and username as password
Hydra v8.9.1 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for
illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-05 14:52:36

[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session
found, to prevent overwriting, ./hydra.restore
[DATA] max 1 task per 1 server, overall 1 task, 3 login tries (l:1/p:3), ~3 tries per task
[DATA] attacking smb://192.168.101.156:445/
1 of 1 target completed, 0 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-05 14:52:59
```

Progress	Tool	Host	Start time	End time	Status
	ftp-default (21/tcp)	192.168.101.197	05 Mar 2025 14:53:44	05 Mar 2025 14:54:18	Finished
	nmap (stage 4)	192.168.101.150-...	05 Mar 2025 14:53:44		Running
	smbenum (445/tcp)	192.168.101.156	05 Mar 2025 14:51:31	05 Mar 2025 14:52:59	Finished

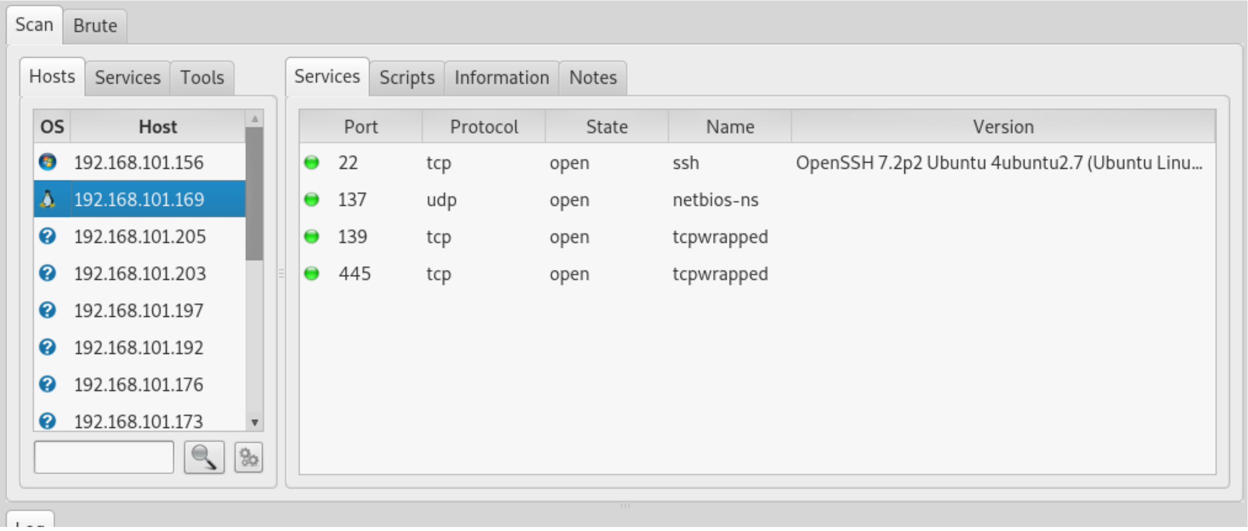
Figure 2a: Sparta output



OS	Host
	192.168.101.156
	192.168.101.169
?	192.168.101.205
?	192.168.101.203
?	192.168.101.197
?	192.168.101.192
?	192.168.101.176
?	192.168.101.173

Port	Protocol	State	Name	Version
135	tcp	open	msrpc	Microsoft Windows RPC
137	udp	open	netbios-ns	Microsoft Windows netbios-ns (workgroup: ENTE...
139	tcp	open	netbios-ssn	Microsoft Windows netbios-ssn
445	tcp	open	microsoft-ds	Microsoft Windows Server 2008 R2 - 2012 micros...

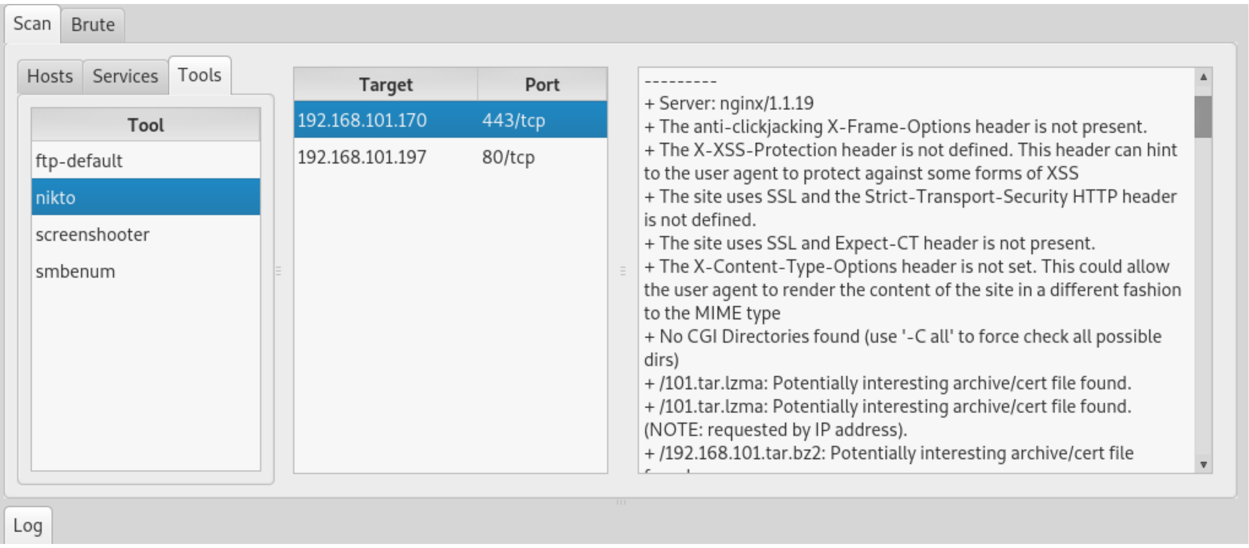
Figure 2b: Sparta output



OS	Host
	192.168.101.156
	192.168.101.169
?	192.168.101.205
?	192.168.101.203
?	192.168.101.197
?	192.168.101.192
?	192.168.101.176
?	192.168.101.173

	Port	Protocol	State	Name	Version
●	22	tcp	open	ssh	OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linu...
●	137	udp	open	netbios-ns	
●	139	tcp	open	tcpwrapped	
●	445	tcp	open	tcpwrapped	

Figure 2c: Sparta output

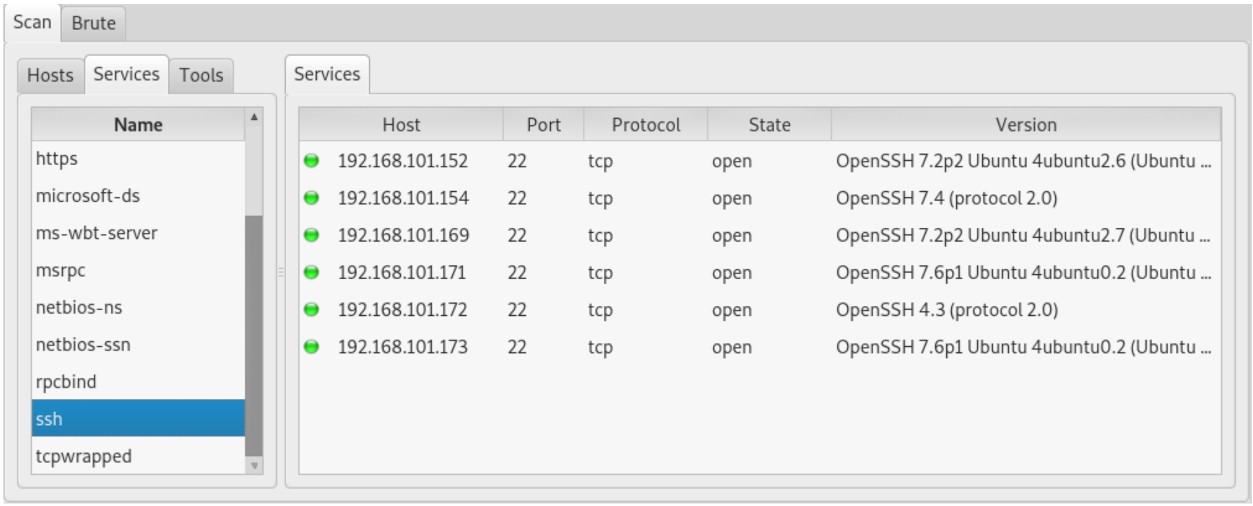


Tool
ftp-default
nikto
screenshooter
smbenum

Target	Port
192.168.101.170	443/tcp
192.168.101.197	80/tcp

```
---------
+ Server: nginx/1.1.19
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint
to the user agent to protect against some forms of XSS
+ The site uses SSL and the Strict-Transport-Security HTTP header
is not defined.
+ The site uses SSL and Expect-CT header is not present.
+ The X-Content-Type-Options header is not set. This could allow
the user agent to render the content of the site in a different fashion
to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible
dirs)
+ /101.tar.lzma: Potentially interesting archive/cert file found.
+ /101.tar.lzma: Potentially interesting archive/cert file found.
(NOTE: requested by IP address).
+ /192.168.101.tar.bz2: Potentially interesting archive/cert file
```

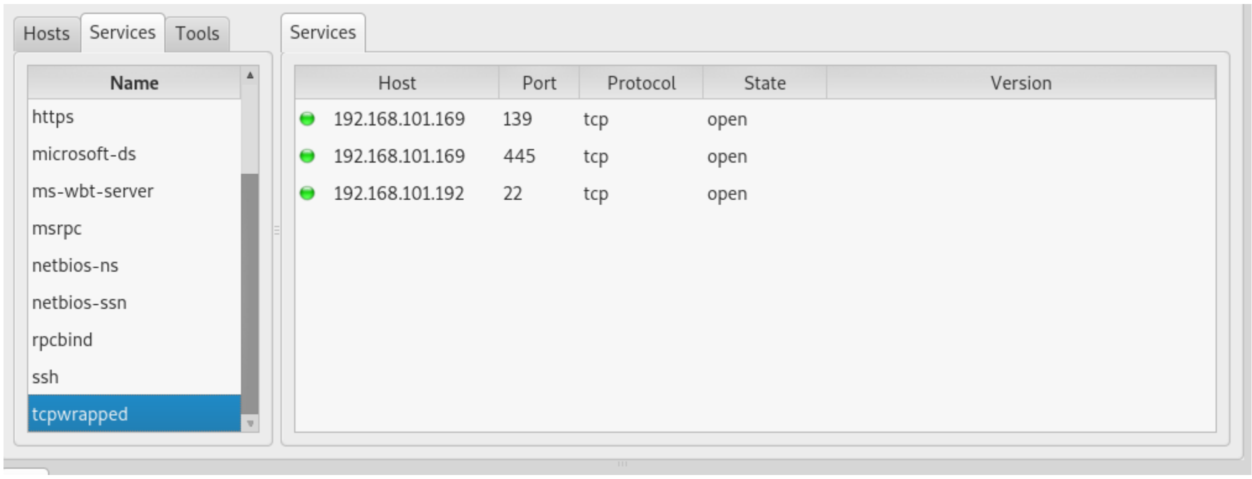
Figure 2d: Sparta output



Name
https
microsoft-ds
ms-wbt-server
msrpc
netbios-ns
netbios-ssn
rpcbind
ssh
tcpwrapped

	Host	Port	Protocol	State	Version
●	192.168.101.152	22	tcp	open	OpenSSH 7.2p2 Ubuntu 4ubuntu2.6 (Ubuntu ...
●	192.168.101.154	22	tcp	open	OpenSSH 7.4 (protocol 2.0)
●	192.168.101.169	22	tcp	open	OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu ...
●	192.168.101.171	22	tcp	open	OpenSSH 7.6p1 Ubuntu 4ubuntu0.2 (Ubuntu ...
●	192.168.101.172	22	tcp	open	OpenSSH 4.3 (protocol 2.0)
●	192.168.101.173	22	tcp	open	OpenSSH 7.6p1 Ubuntu 4ubuntu0.2 (Ubuntu ...

Figure 2e: Sparta output



Name
https
microsoft-ds
ms-wbt-server
msrpc
netbios-ns
netbios-ssn
rpcbind
ssh
tcpwrapped

	Host	Port	Protocol	State	Version
●	192.168.101.169	139	tcp	open	
●	192.168.101.169	445	tcp	open	
●	192.168.101.192	22	tcp	open	

Figure 2f: Sparta output