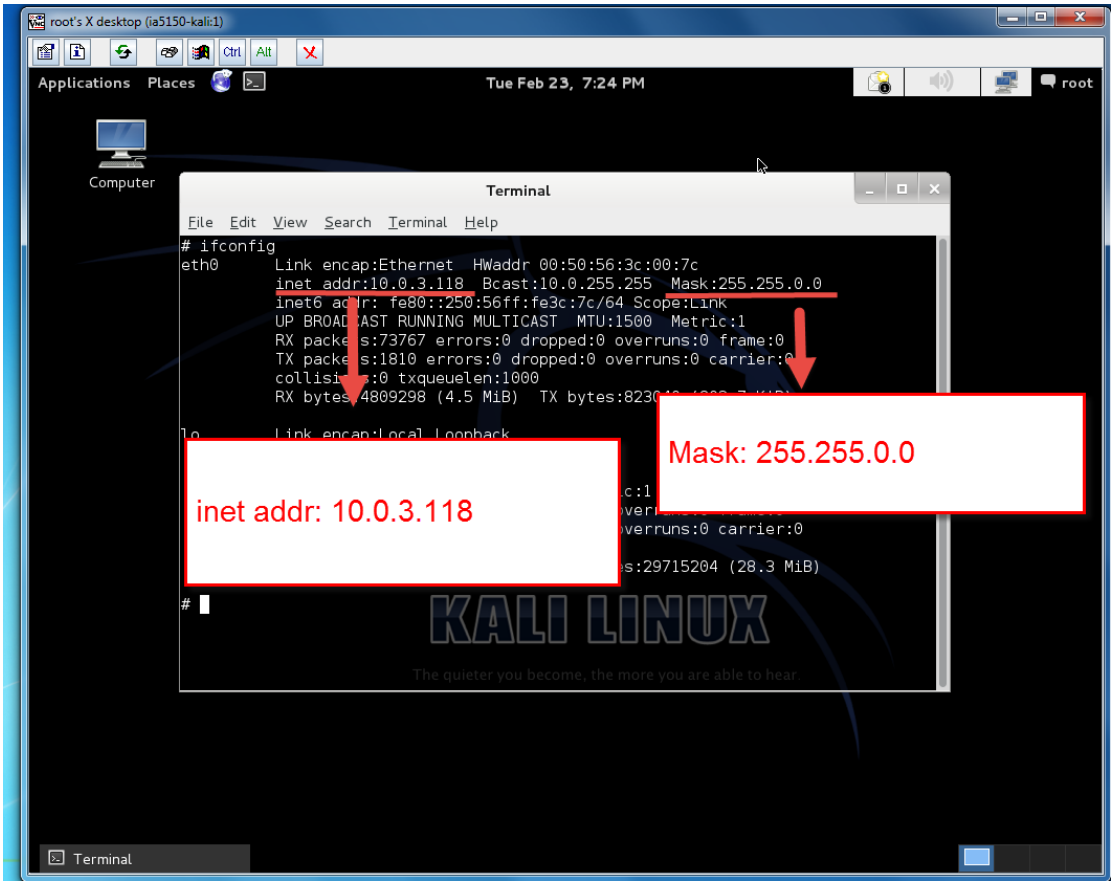
Lab 2

1. POD IP range and Network CIDR address for the entire LAN



POD IP Range: 10.0.3.118 to 10.0.3.121 (10.0.3.118 | 10.0.3.119 | 10.0.3.120 | 10.0.3.121)

Network CIDR address for entire LAN: 10.0.3.0/16

The mask of this machine shows us that the network part of network's IP address is 16 bits and the hosts part of the network's IP address is 16 bits too.

2. Assets inventory from Zenmap network scanning

Host IP	OS Type	OS Version	Ports	Application Info
10.0.1.3	Linux	Linux 3.11 -3.14	22	ssh Protocol 2.0
10.0.1.3	Linux	Linux 3.11 -3.14 (Ubuntu)	80	Apache httpd 2.4.7
10.0.1.3	Linux	Linux 3.11 -3.14 (Ubuntu)	443	Apache httpd 2.4.7
10.0.1.4	Linux	Linux 3.11 -3.14	22	ssh Protocol 2.0
10.0.1.4	Linux	Linux 3.11 -3.14	25	Postfix smtpd
10.0.1.5	Linux	Linux 3.11 -3.14	22	ssh Protocol 2.0
10.0.1.6	Linux	Linux 3.11 -3.14	22	ssh Protocol 2.0
Continue…				

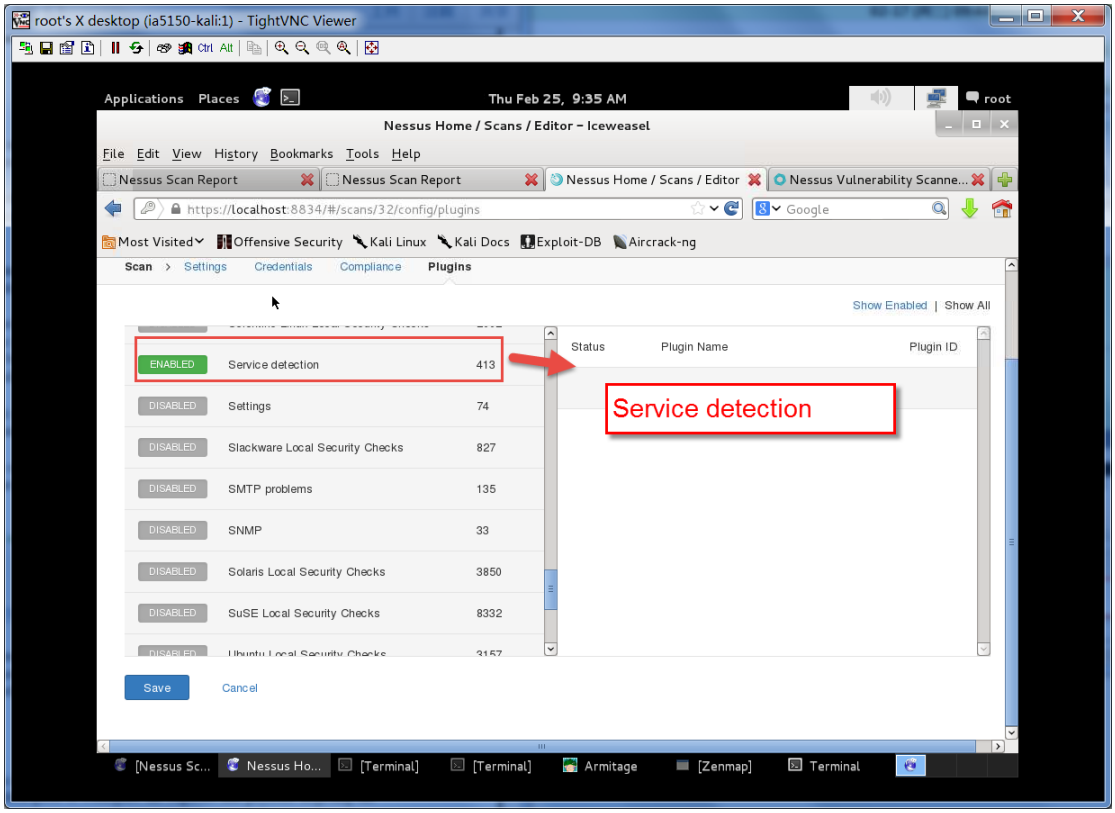
Host IP	OS Type	OS Version	Ports	Application Info
10.0.1.7	Linux	Linux 3.11 - 3.14	22	ssh Protocol 2.0
10.0.1.8	N/A	N/A	22	ssh version: N/A
10.0.1.8	N/A	N/A	139	Netbios-ssn version: N/A
10.0.1.8	N/A	N/A	445	Microsoft-ds Version: N/A
10.0.1.8	N/A	N/A	8080	http-proxy version: N/A
10.0.3.118	Linux	Linux 3.7 - 3.15	5901	VNC protocol 3.8
10.0.3.118	Linux	Linux 3.7 – 3.15	6001	X11 version :N/A
10.0.3.119	Windows	Microsoft Windows XP Professional SP2 or Windows Server 2003	135	Microsoft Windows RPC
10.0.3.119	Windows	Microsoft Windows XP Professional SP2 or Windows Server 2003	139	Netbios-ssn Version: N/A
10.0.3.119	Windows	Microsoft Windows XP Professional SP2 or Windows Server 2003	445	Microsoft Windows XP microsoft-ds

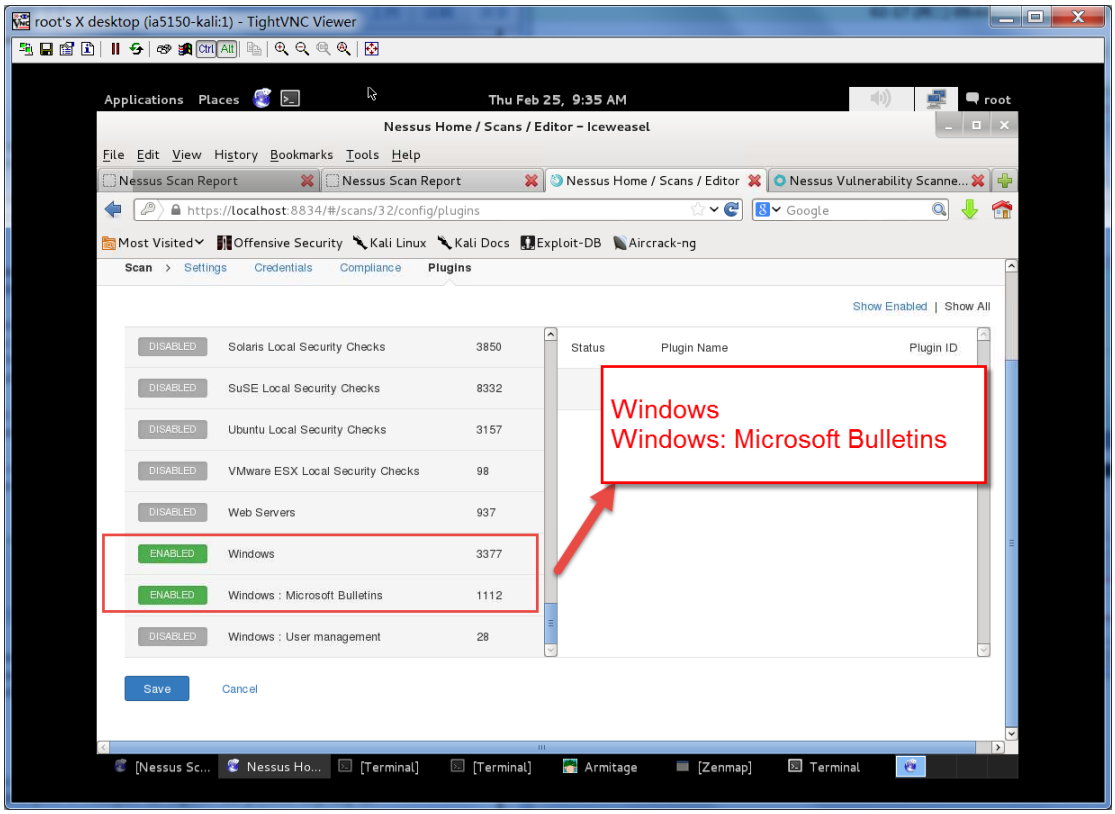
As we saw above on IP 10.0.3.118, there is a service named X11 which doesn't have any version information through Zenmap intensive scan. X11 is a service that provides graphic interface to end users.

Besides, the IP 10.0.1.8, we found that no operation system info and service version info returned by Zenmap. However, we can guess that the host must be running on some version of Windows OS but we cannot 100 percent assure that it is Windows xp, windows 7 or other versions of Windows. The reason why we guess it is a Windows host is that the service netbios-ssn and Microsoft-ds only exist on Windows OS.

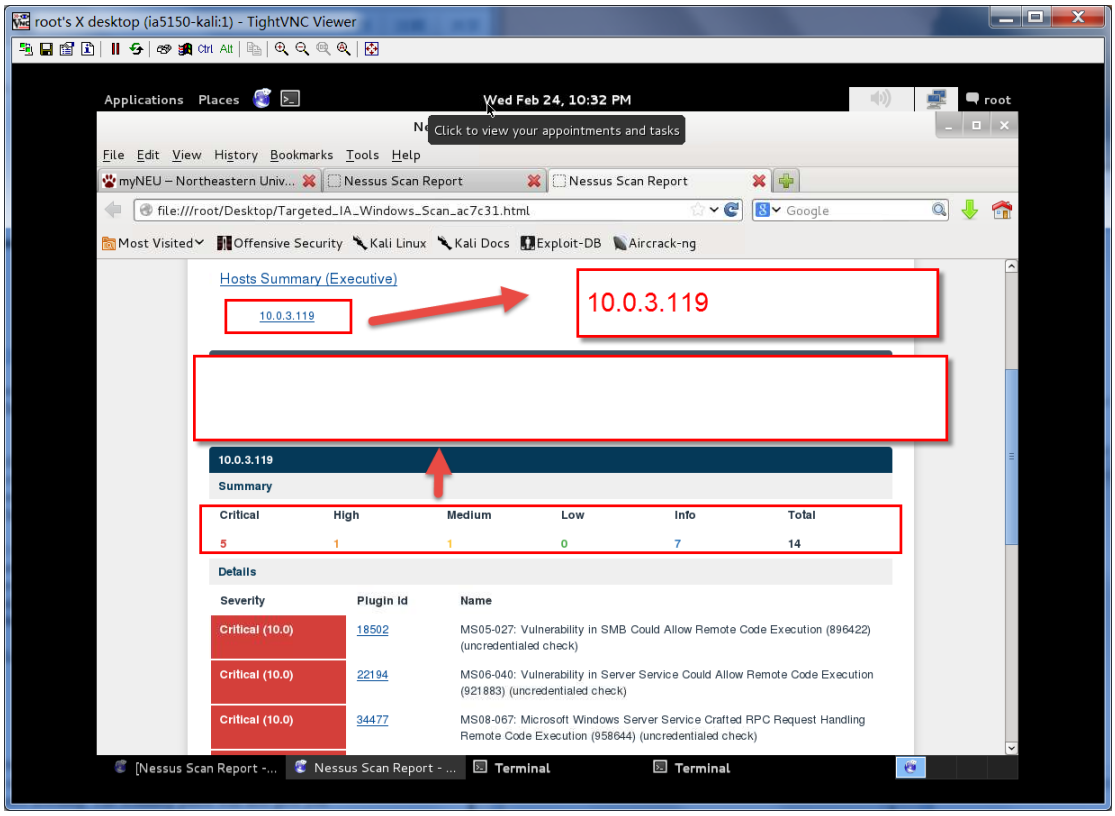
And also, on the IP 10.0.3.119, although the Netbios-ssn didn't disclose its version info, we still can guess its version based on the Windows version so that we might find corresponding vulnerabilities in tools like google hacking database or NVD (National Vulnerability Database) etc.

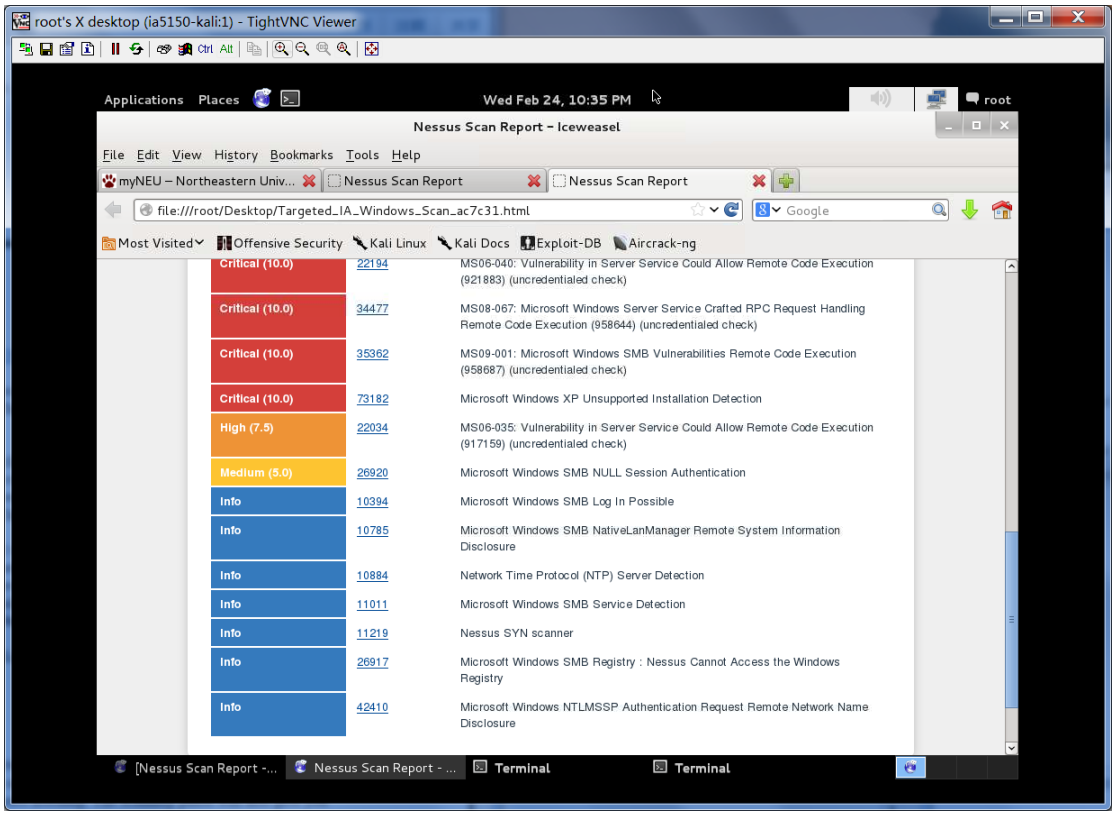
3. Executive html results from the Nessus vulnerability scanning
   a. Targeted IA Windows Scan



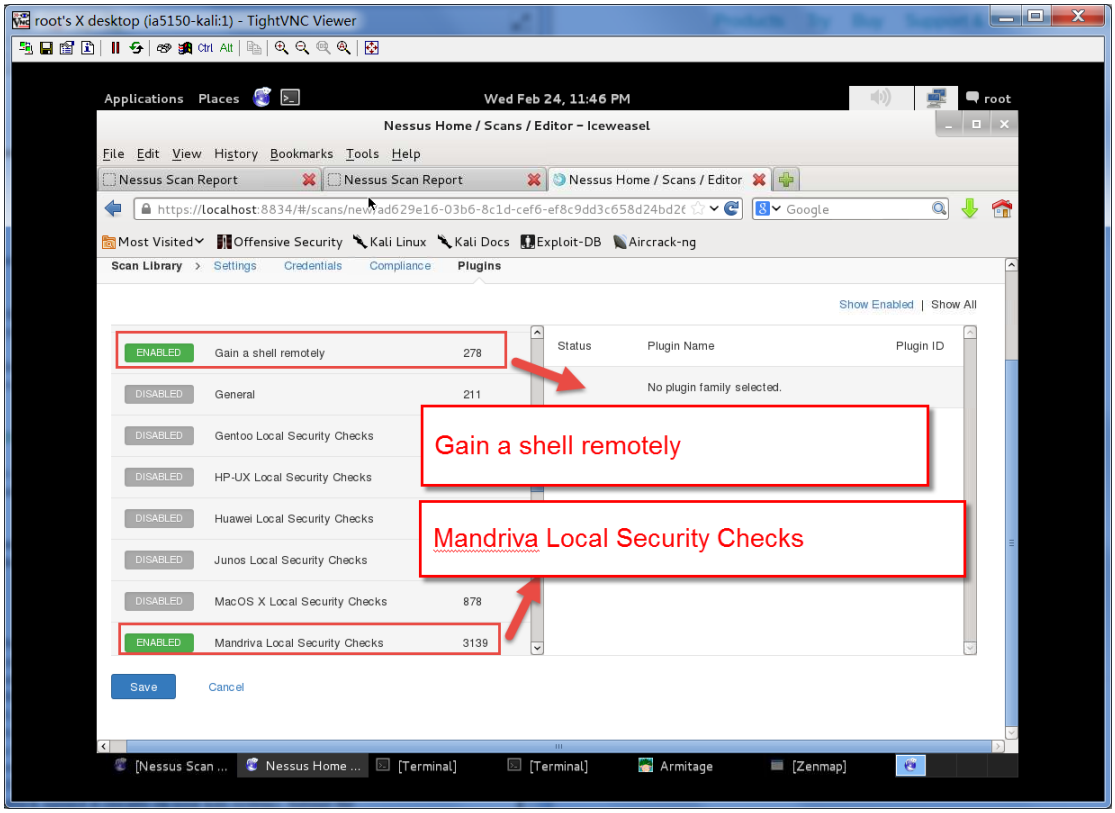


According to ZenMAP results, we know that on Windows it runs several services which belong to Nessus plugin family: Service detection, Windows and Windows: Microsoft Bulletins.

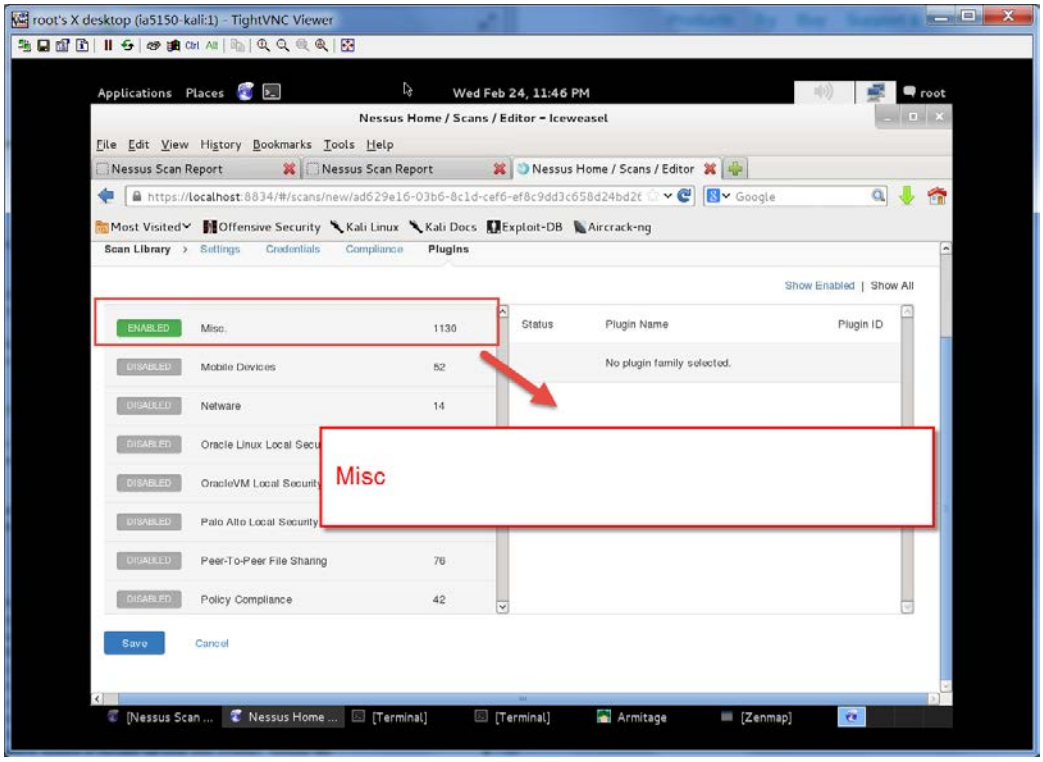




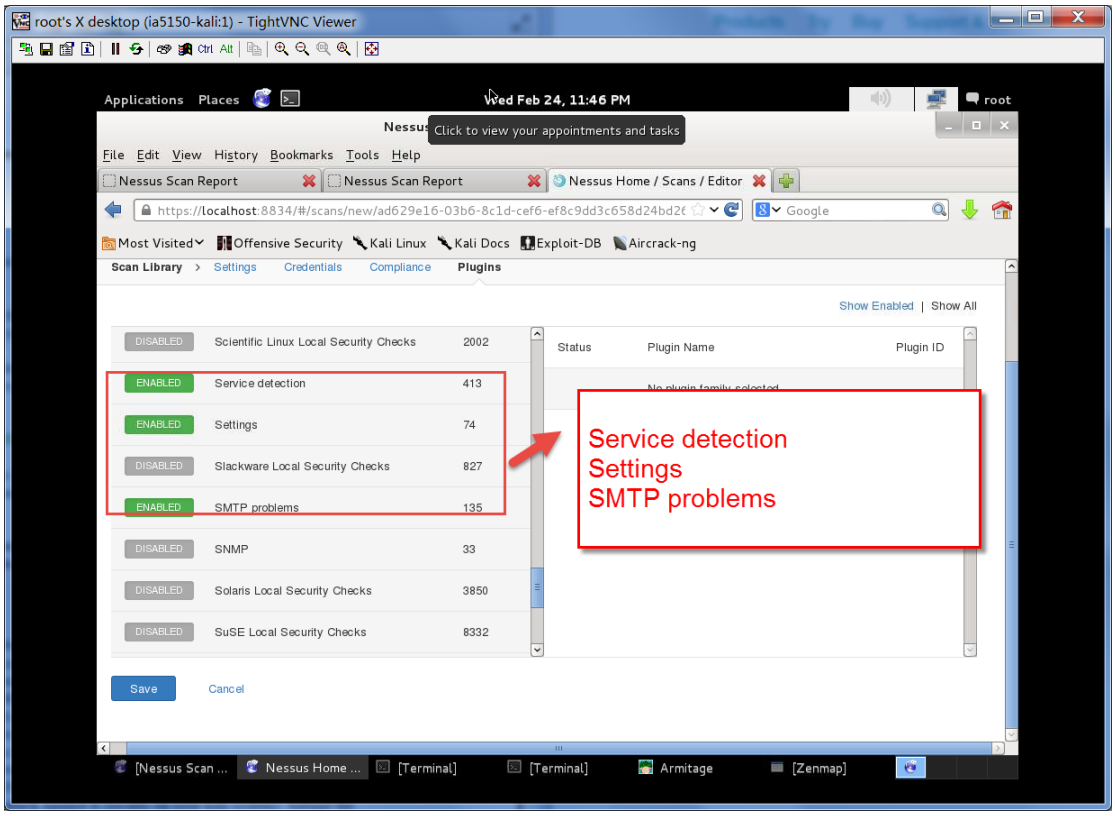
b. Targeted IA Scan

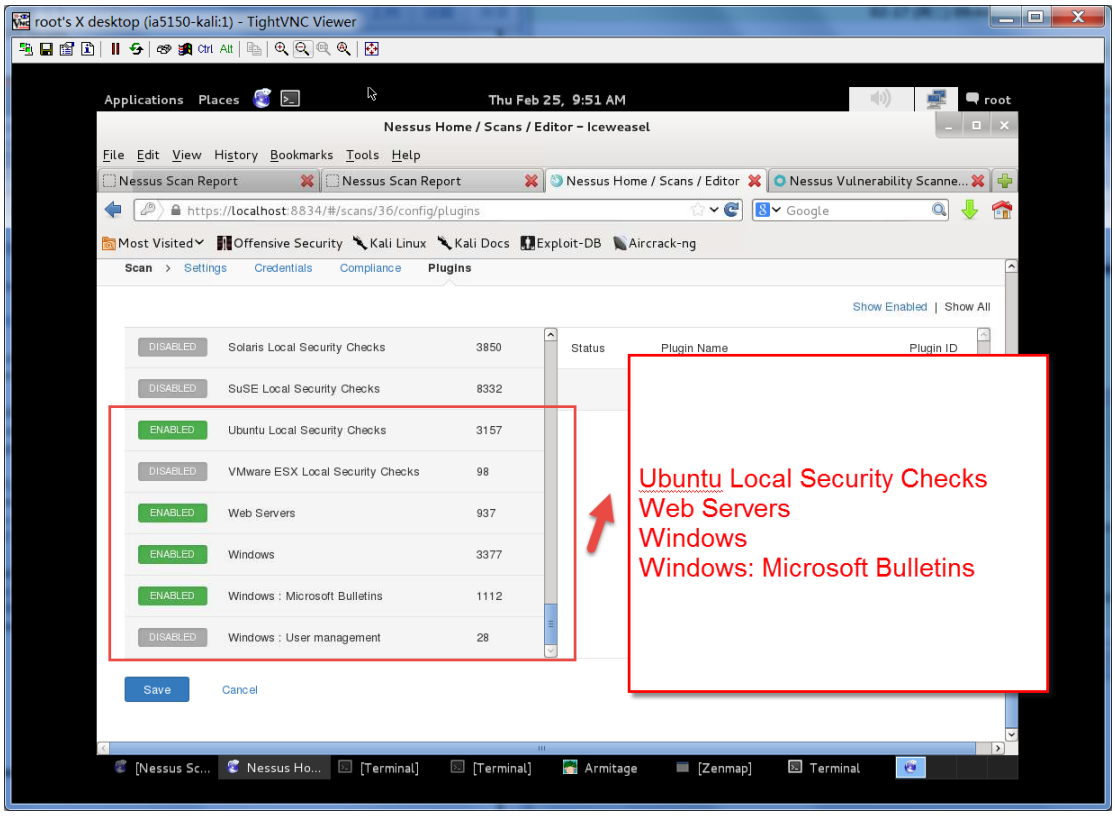


These two was chosen because of the SSH version running on out IP POD range and DMZ IP range.



Misc was chosen because of the VNC related services are running.



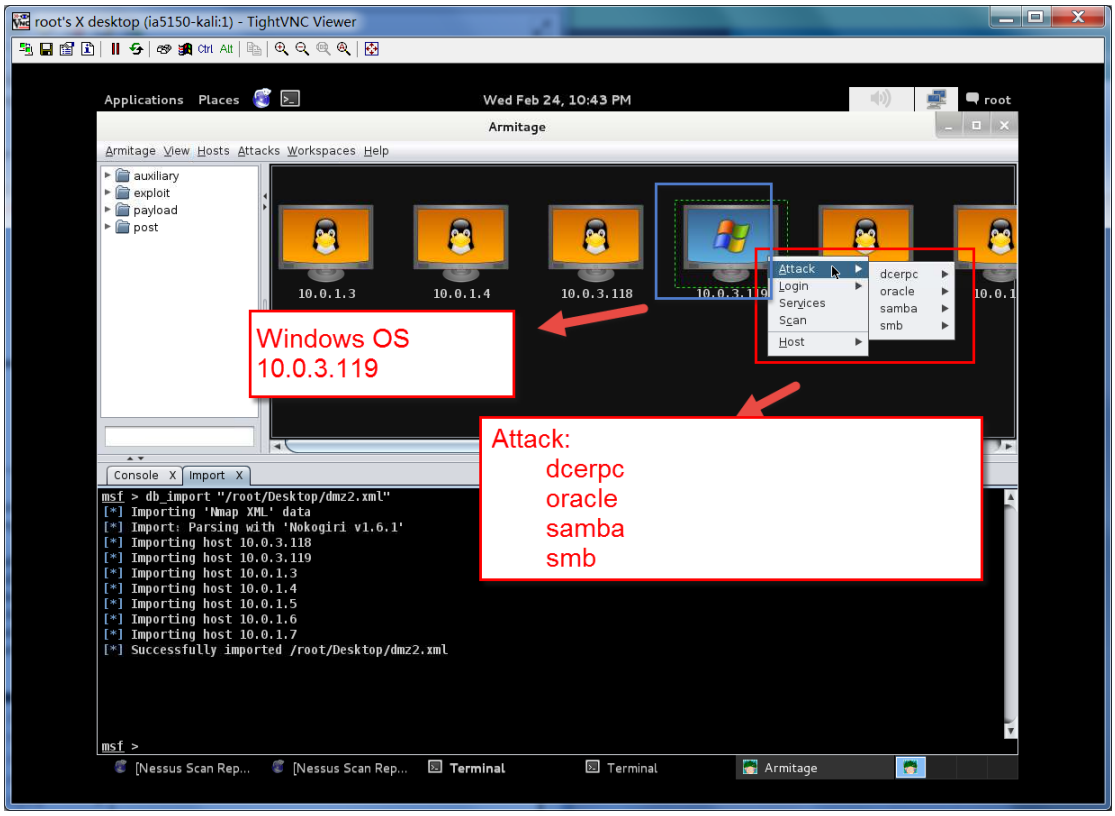


The “Ubuntu Local security Checks” because we identify that the target OS type is Ubuntu. And we choose “Web Server” because on some part of those Linux machines, Apache runs on it which provides HTTP services. Both windows plugin are chosen because there is a windows server running.

4. Which exploits did you choose to try on the windows target and why? Which exploit worked and which didn’t

As we run Armitage tool in Kali and import the intensive scan report, we found that this server might contain the below attacks.

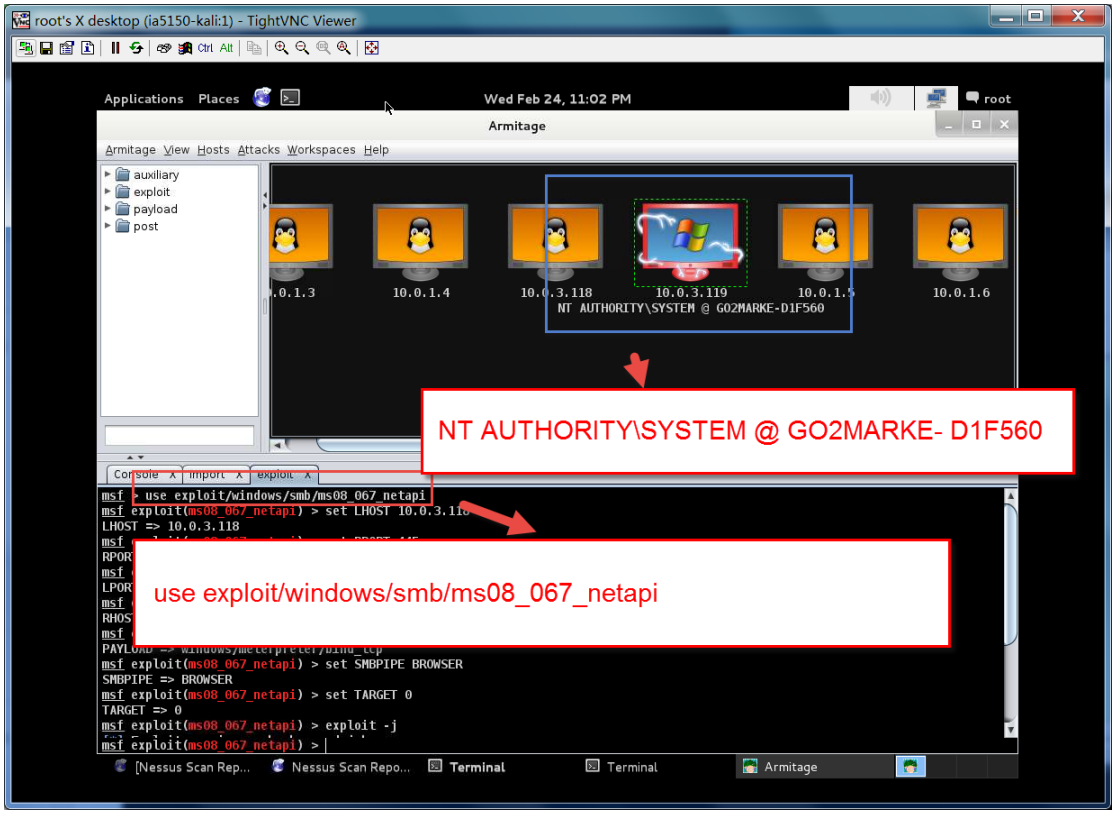
- Decrpc
- Oracle
- Samaba
- Smb



As we mentioned, these are some potential attack surfaces that exists on the target windows server. When we look one step deeper, we can found that we can use this tool to test which of these attacks might take effect. We tried all attacks on dcerpc, oracle and we found that the windows server immune to those attacks. But instead of trying every option there, we can use the Nessus report to narrow down the choices.

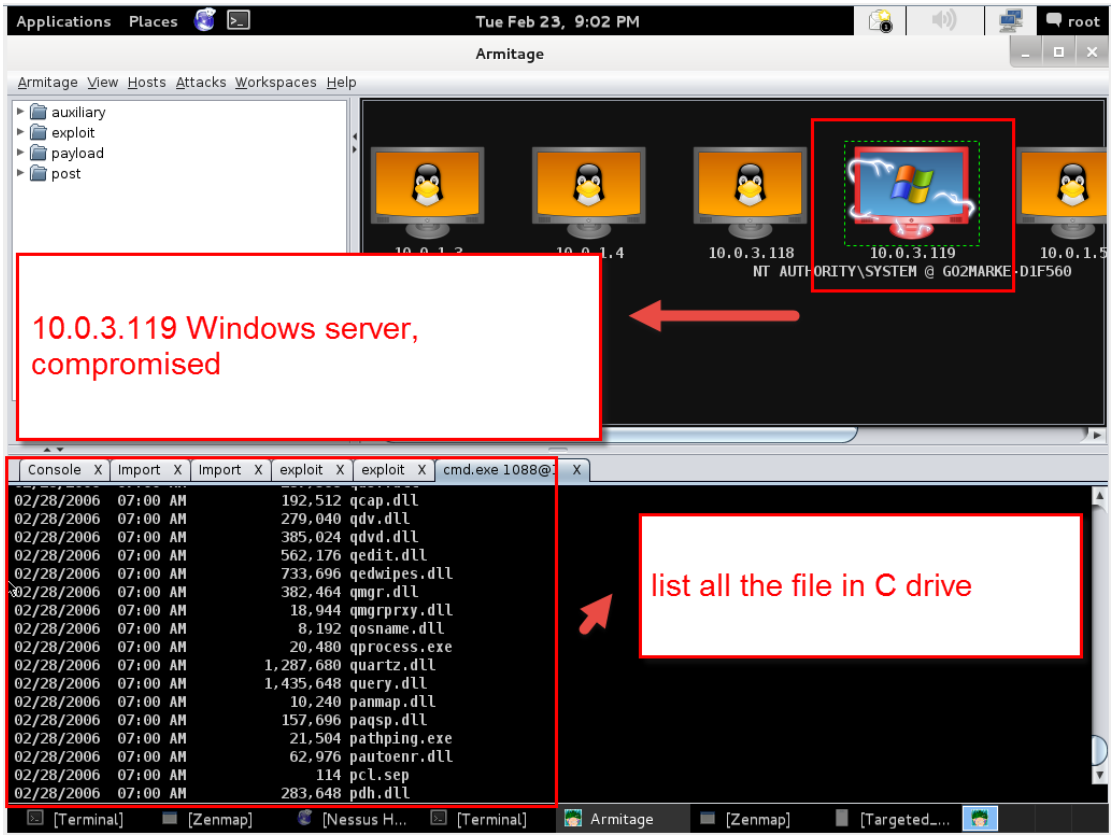
As showed on the part 4 in Nessus report, we found that the biggest

vulnerability should be related to SMB service which has critical marks on the report. Though, we directly choose SMB and check whether it is exploitable. In this step, we found that the vulnerability of smb related function of ms08_067_netapi can be exploited. So, we use Armitage to launch an attack.



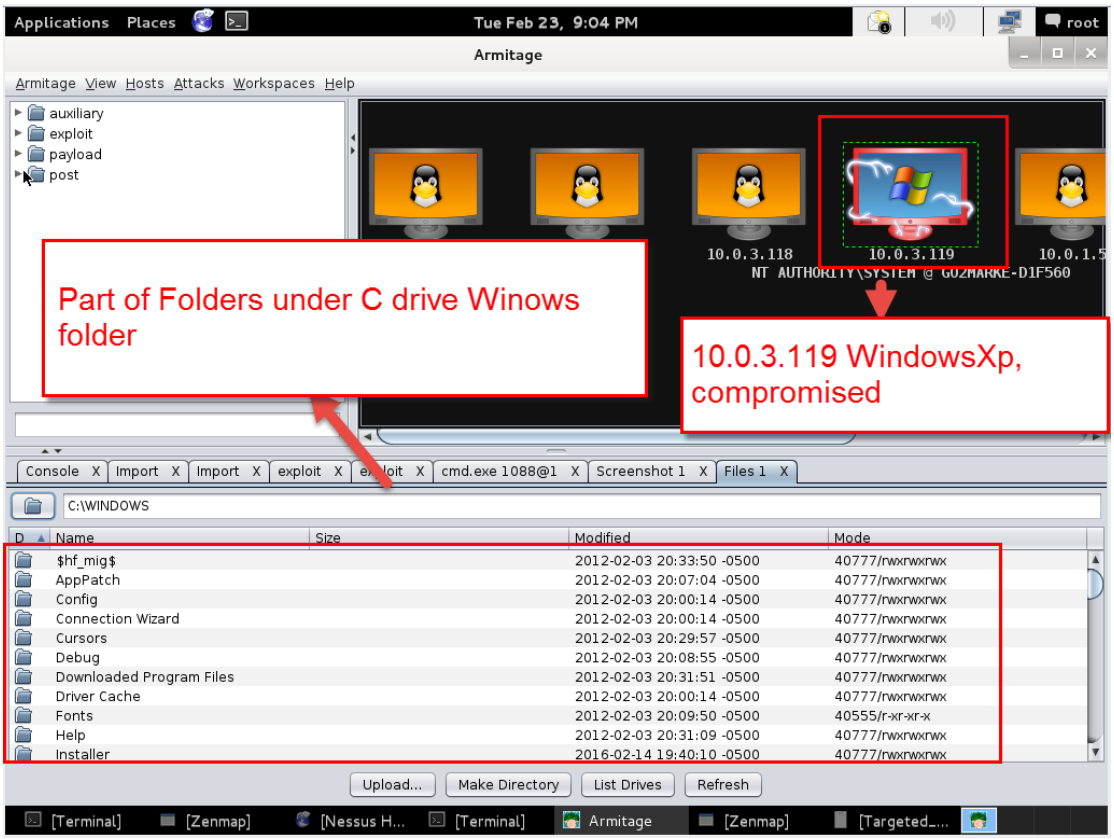
As we saw, the attack was successfully launched.

5. Screenshot of the directory listing located on the compromised system



This output was generated by the command line commands “dir” which is used to show all the files/directories under current path. Actually, we exploit the vulnerability of SMB and use it to execute metasploit shell code payload which makes us can connected to the Windows command line interface which is the same as the cmd utility on Windows. Once we get the CLI prompter, we can execute any commands which are allowed under the permissions of the current process. In this case, we take advantage of SMB service which has the “root/Administrator” permission – highest privileged user in Linux and Windows. Thus, we are able to do numerous things.

The figure below shows the metasploit attack on browse all files. Similarly, I think that it works like the attack process mentioned in above paragraph – gain system CLI and run arbitrary commands. In this case, after metasploit found vulnerability, it searches the best-match payload to get the access to CLI and then execute commands like “dir” or something. So, we are able to get the lists of files/directories of victim machine.



6. What is missing from the lab to cover all steps of the attack architecture? What kali tools can accomplish this?

For attack architecture, we have reconnaissance, scanning, gaining Access using OS and APP attacks, gaining Access using network attacks, maintaining access and covering tracks. Currently, in this lab we cover first three and don't cover the rest.

For gaining access through network attacks, Kali provides us bunch of network sniffing tools like Wireshark, TCPdump, WebScarab, Burp Suite, p0f2, ettercap etc.

For maintaining access, we also have bunch of tools to choose like CryptCat Cymothoa, dbd, dns2tcp, PowerSploit,Webshells etc.

For covering tracks, we can use Armitage, metasploit to erase our footprint after an attack. Plus, we can use tool like backdoor factory to gain the shell prompter and escalate our privilege. Then we can use simple utilities provided by targets OS to clear out track. For example, in windows, we can just use clearlogs.exe to clear all the event logs. For Linux target, we also can take advantages of system settings like minimize the command history buffer, for example set them to 0. Then, it will not record what we did in the shell prompt.

7. How would you defend against the phases of the attack architecture covered in this lab? Be detailed in your response.

   Generally, the first step is to reduce the attack surface, we should shutdown unnecessary services and ports for instance, the smb services. Then, we need to upgrade and patch our system so that much old vulnerability would not take effect any more.

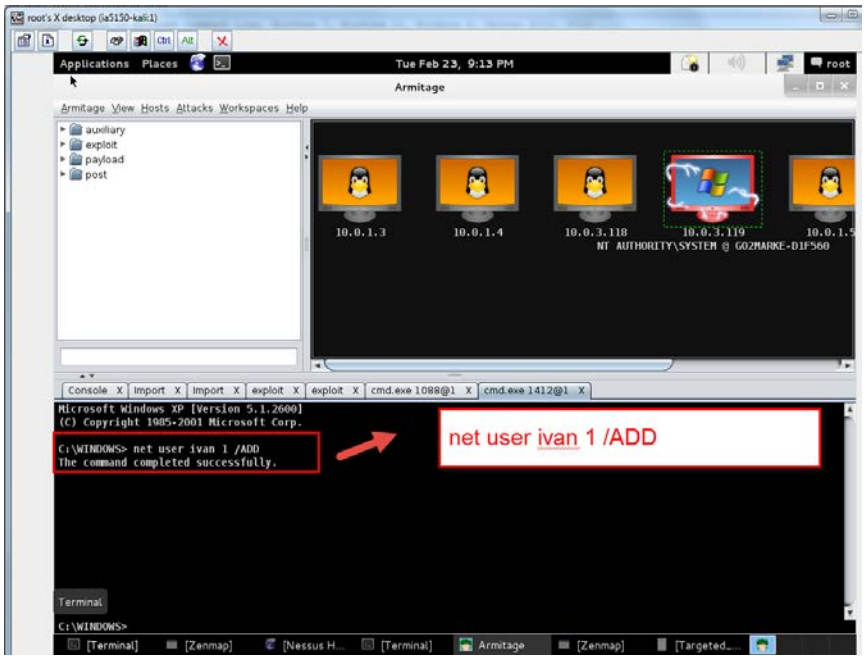
   Specifically, for the Nmap scan, we need set up firewall to filter ICMP reply packets out which will make Namp scanner believe that the host and port is shutdown. And, another way is to monitor the traffic which only works for aggressive probe sniff and not suitable for passive sniff because it is so hard to distinguish small traffic changes.

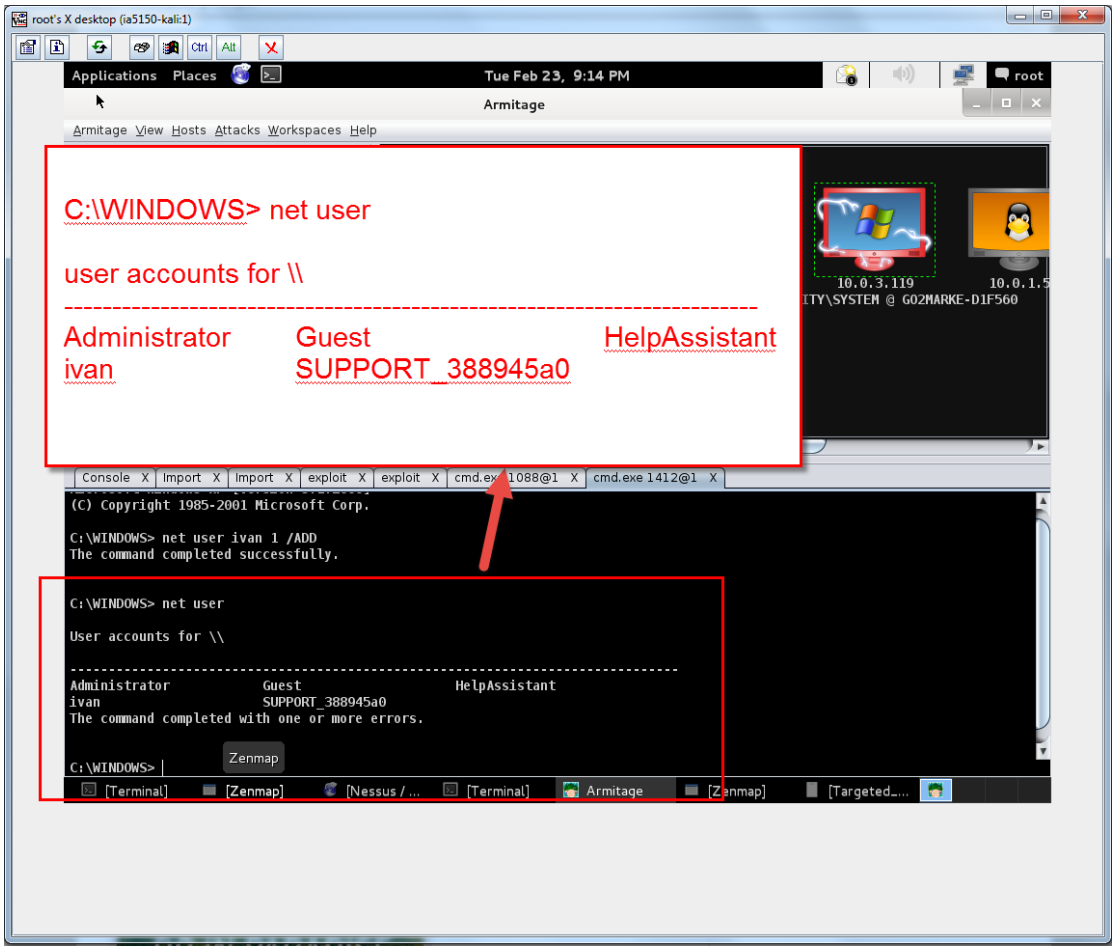
   For metasploit attack, in some cases, we should run our services/program follow least-privilege principle. The program/services should not run as highest privilege if their work has been done. Even if the attacker exploits the vulnerability in these applications, they still cannot do too much on the target system like install backdoor or something else like swipe data.

   For host perspective, we should install host firewall and we should set up a series policies which help us to harden our system like implement strengthen password policy and so on.

8. Bonus

a. Create a user on the remote windows XP system



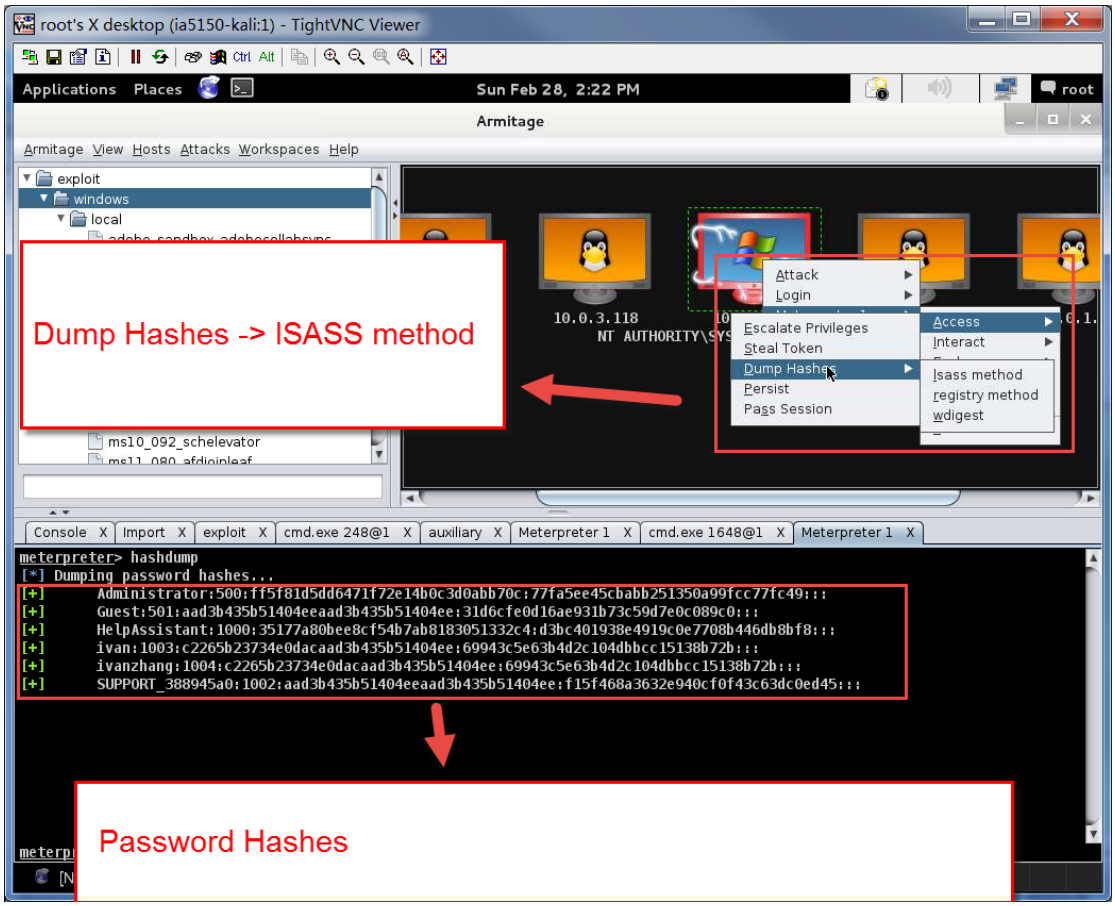


b. Add username and password to Nessus and make Nessus can login to Windows XP and Perform “Windows: User Management” family scan of XP system

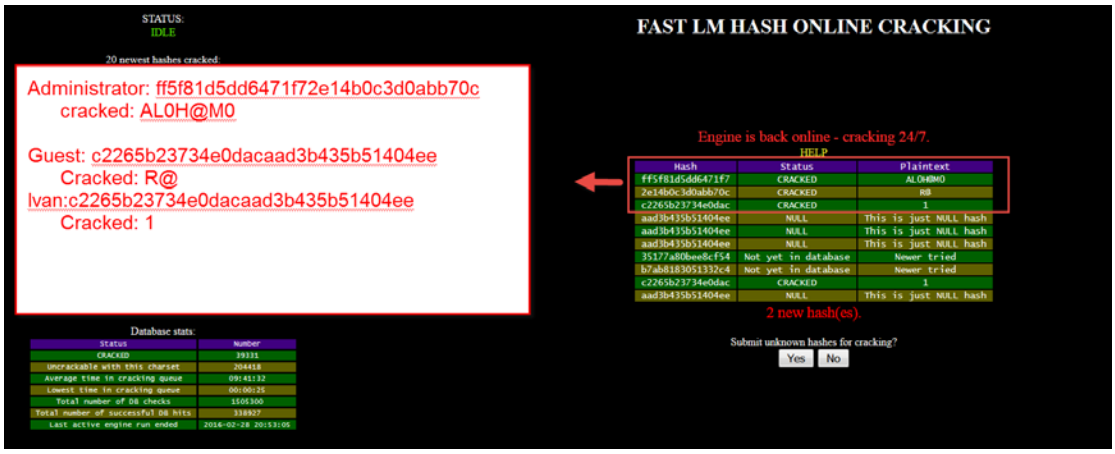
c. Scan DMZ hosts from all IPs in your POD. Find vulnerability on a DMZ host and get shell access.

d. Find a user/password file on the file system and perform an offline password attack against it

As we can get the Windows system exploited, we now can use Armitage ->hash dump functionality to dump the username/password as following pictures show



Then, we can use the website of “http://rainbowtables.it64.com/” to decrypt these hashes into plaintext



Role	Hash	Plain text
Administrator	ff5f81d5dd6471f72e14b0c3d0abb70c	AL0H@M0
ivanZhang	c2265b23734e0dacaad3b435b51404ee	1
Ivan	c2265b23734e0dacaad3b435b51404ee	1
Guest	aad3b435b51404eeaad3b435b51404ee	N/A (Empty password)