**NMAP**

**Introduction:**

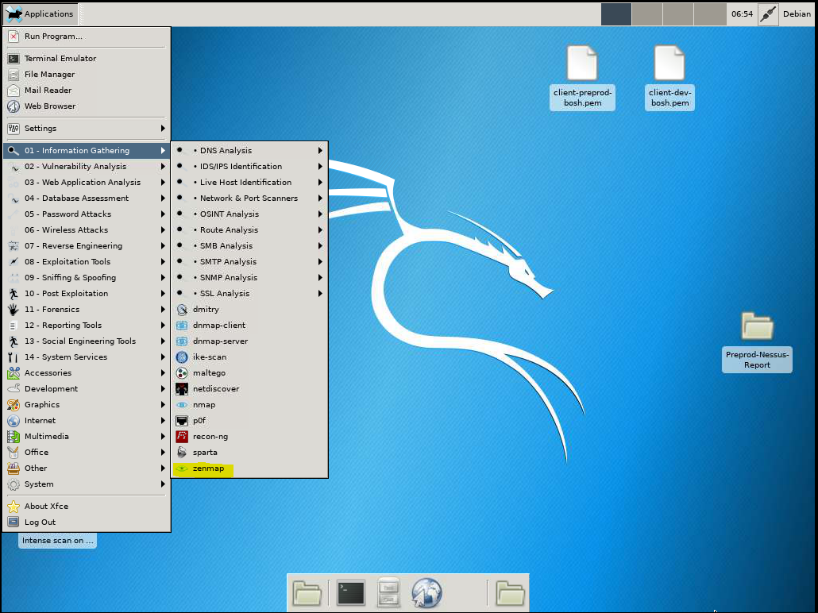
Nmap, short for Network Mapper, is a free, open-source tool for vulnerability scanning and network discovery. Network administrators use Nmap to identify what devices are running on their systems, discovering hosts that are available and the services they offer, finding open ports and detecting security risks.

Nmap can be used to monitor single hosts as well as vast networks that encompass hundreds of thousands of devices and multitudes of subnets.

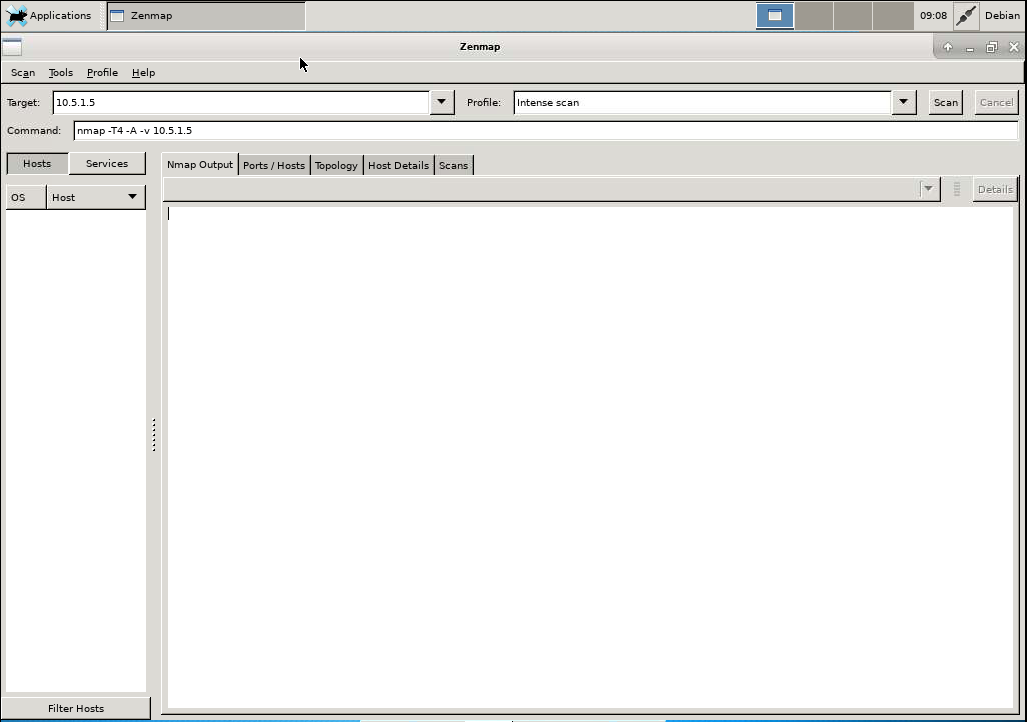
Though Nmap has evolved over the years and is extremely flexible, at heart it's a port-scan tool, gathering information by sending raw packets to system ports. It listens for responses and determines whether ports are open, closed or filtered in some way by, for example, a firewall. Other terms used for port scanning include port discovery or enumeration.

**Following are the steps for NMAP scan:**

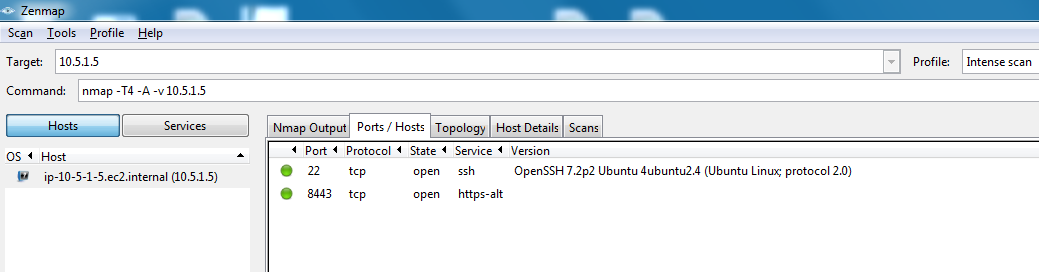
1. In our scan setup, we have already done setup for Kali Linux instance on AWS(EC2) with GUI interface, which is required for Nmap scan. Please refer document ‘Kali linux Instance.doc’.
2. To perform Nmap scan on VMs, we need to have **ZenMap** application which is GUI for Nmap. Kali Linux by default provide this application. Open this app from Kali Linux GUI. On GUI, left most corner, click on **Applications**. Then select **01-Information Gathering** **> zenmap** as shown below**.**

****

1. Now we can perform Nmap port scan on VMs. We need to enter IP of VM in **Target**: section. We can also add CIDR range for scan. Zenmap provides different type of scan like Intense scan, Regular scan, Quick scan, Ping scan etc. We can select type of scan from **Profile:** dropdown given in Zenmap. As shown below. (Zenmap also provide **Command:** section, we can also perform Nmap scan using command) as shown below.

****

1. Once entering IP and selecting type of scan hit on **scan** button to start scan. Command for scan is shown in command section. It will take some time to scan VMs, depending upon number of VMs.
2. In scan output zenmap listed out ports, protocol and state of port, which service is running on that port and version of service running on port, as shown below.



1. At start we were facing issue while scanning VMs in other VPC. As per the list of VMs, VMs are present in 3 different VPCs. i.e. aws-preprod-client, preprod & global VPC. Our Kali Linux VM is present in aws-preprod-client VPC, so we were facing issue while accessing VMs of preprod & global VPC. Also, global VPC is present in Oregon region.
2. To solve this issue, we need to ensure VPC peering between aws-preprod-client & preprod VPC, and also between aws-preprod-client & global VPC. Also, there were changes required in few Security groups to be able to reach all the VMs required. In our case there was VPC peering present between aws-preprod-client & preprod VPC and also between aws-preprod-client & global VPC.
3. In some security group of VMs present in other VPC we add following inbound rule:

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Protocol** | **Port Range** | **Source** |
| All traffic | All | 0-65535 | Custom - IP of Kali linux VM |

These are some security group in which we have added inbound above rule:

1) sg-7581160d (preprod-director) in N.Virginia region.

2) sg-0c48e76148eb9d659 (global-bosh-deployment) in Oregon region.

1. Once scan is completed, we can save scan generated by zenmap for particular VM/CIDR range. To save scan click on **Scan** tab, present at left most corner of zenmap tool. Then click on **Save scan**. We can save scan file in 2 formats i.e. .xml & .nmap(Nmap text format).
2. We have consolidated the scan result of all VMs in excel report as following table format. For that we need to store scan file in .xml format. We open this file on local machine in Zenmap GUI tool and consolidated the scan result of all VMs.

