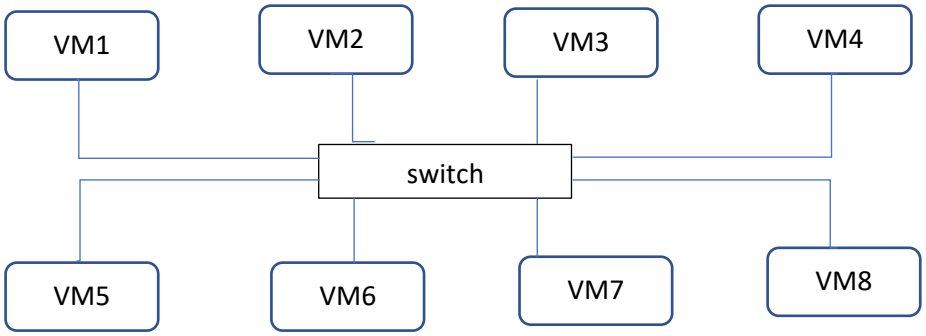
COEP Hackathon 2019

Problem Statement

In this problem you are helping the security administrator determine the impact of a breach (for instance, a malware attack). Your solution will highlight the risk to other VMs(Virtual Machines) based on the knowledge that a specific VM has been infected.

To simplify the problem so that it to be solved in a single day, consider the following setup:

You would be given a set of VM (IPs) all connected over switched ethernet (in a single logical subnet).



Along with the list of VMs you will get a sample set of iptables rules for each VM. The rules will be of the form <destination ip, protocol, ALLOW/DENY>. For instance, consider the following rule:

```
<192.168.1.25, tcp, ALLOW>
<192.168.1.66, udp, DENY>
<192.168.1.1, all, ALLOW>
<-, all, DENY>
```

If these rules are published in this order, then the current VM will explicitly accept tcp traffic from source IP(192.168.1.25), reject udp traffic from source IP (192.168.1.66), accept all traffic from source IP (192.168.1.1) and reject all other traffic from all other source IPs.

*For those who wish to dig further, assume that the iptables rules only affect the **filter** table (rules applicable to other tables like nat, mangle, raw or security can be ignored).*

These rules determine the reachability of a VM to the other VMs for a select set of protocols {tcp, udp, icmp, all}. Each protocol is associated with a (risk) cost which signifies the total risk of infection associated with the protocol. The (hypothetical) cost of the risk involved is a function of, amongst other things, prevalence of protocol usage, known CVE (common vulnerability and exposures) count, and protocol security features. You can assume the risk associated with each protocol as per the table below. A lower cost of risk indicates a higher impact.

Protocol	Cost of Risk
tcp	45
udp	50
icmp	100
all	0

Your **goal** is to come up a solution which, given the above inputs, allows the user to tag a specific VM as vulnerable and based on the potential of the vulnerability to jump the VM firewalls, assess the security risk to other VMs in the subnet.

For instance, VM firewall rules may allow VM1 <--> VM2 connectivity and VM2 <--> VM3 connectivity, but no direct access from VM1 <!-> VM3. Despite this rule, VM3 may be vulnerable, since a malware affecting VM1 may first affect VM2 and then affect VM3.

The outcome expected is a list of "at risk" VM(s) (IPs) in descending order of risk.

Do note that the sample inputs provided early on to assist with developing your solution may differ from the inputs used to test the correctness of your solution. Do not develop a solution for a fixed input set.

Bonus Points:

1. If a new VM IP is provided along with its iptables rules, how easy (or difficult) is it to recompute the vulnerability assessment again. The evaluation would be based on additional time taken to compute the results as well as additional space required.
2. A graphical interface that shows the network topology, takes user inputs indicating the infected VM and displays the vulnerability map will fetch additional points.

The Approach

There are many ways to implement the solution. Since we don't wish to dictate your approach, here is some generic guidance.

1. Understand the problem statement (use the mentor(s) assigned to the problem statement)
2. Come up with a technical solution (design) to the problem.
3. Identify, if needed, any open source components which fit the design needs
4. Identify the gaps + glue required to satisfy the remaining functionality
5. Design the user interface for the feature
6. Implement the solution by leveraging the open source components identified in step 3.
7. Demonstrate the feature for a specific use case
8. Explain the entire solution, and the challenges faced in the form of a presentation.

Take Away for the Students

The purpose of this problem is to expose the students to a pertinent, real world use case and encourage them to come up with an innovative solution to address the problem.