Dynamic Reconfiguration of Network Based on Security Events

Vignesh Goutham Ganesh, Ram Gandhi Arumugaperumal, Varsha Sundar

Department of Electrical Engineering, San Jose State university, San Jose, California 95192.

Introduction

This project aims to create a secure, robust and a proactive network environment which can counteract ever-evolving cyber threats in a more efficient way than the current methods. In order to achieve this goal we utilize Software Defined Networking (SDN) technology by adding a security control application on top of SDN network controller. Our SDN security application will take evasive actions (reconfiguration) on its own rather than placing all security action responsibility to the user.

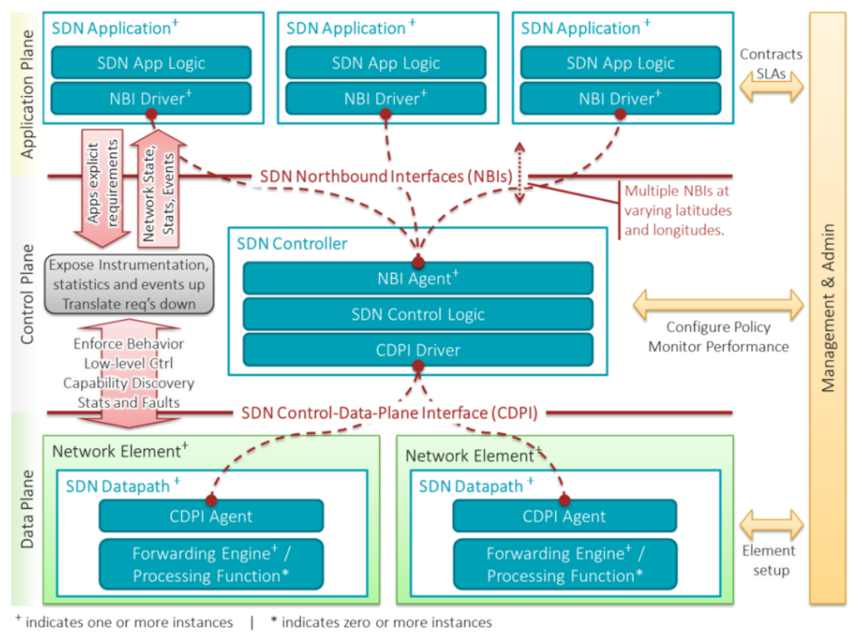
The main attacks taken into consideration would be DoS, DDoS and other prevalent cyber-security threats. A study will also take place on the rare and other types of attacks that leads to security threat. A key part of our security controller application is the reconfiguration to take place automatically, various parameters of the network have been defined, which when exceeds the threshold value, can force the controller to reconfigure the network to make it more secure.

The security of the network, then, can be measured by comparing the flow of packets in the network prior to and after the reconfiguration of the system takes place. The effect of the policy change by the controller affects the various layers of the network and the each type of attack warrants different policy change to make the counter-attack more versatile.

Methodology

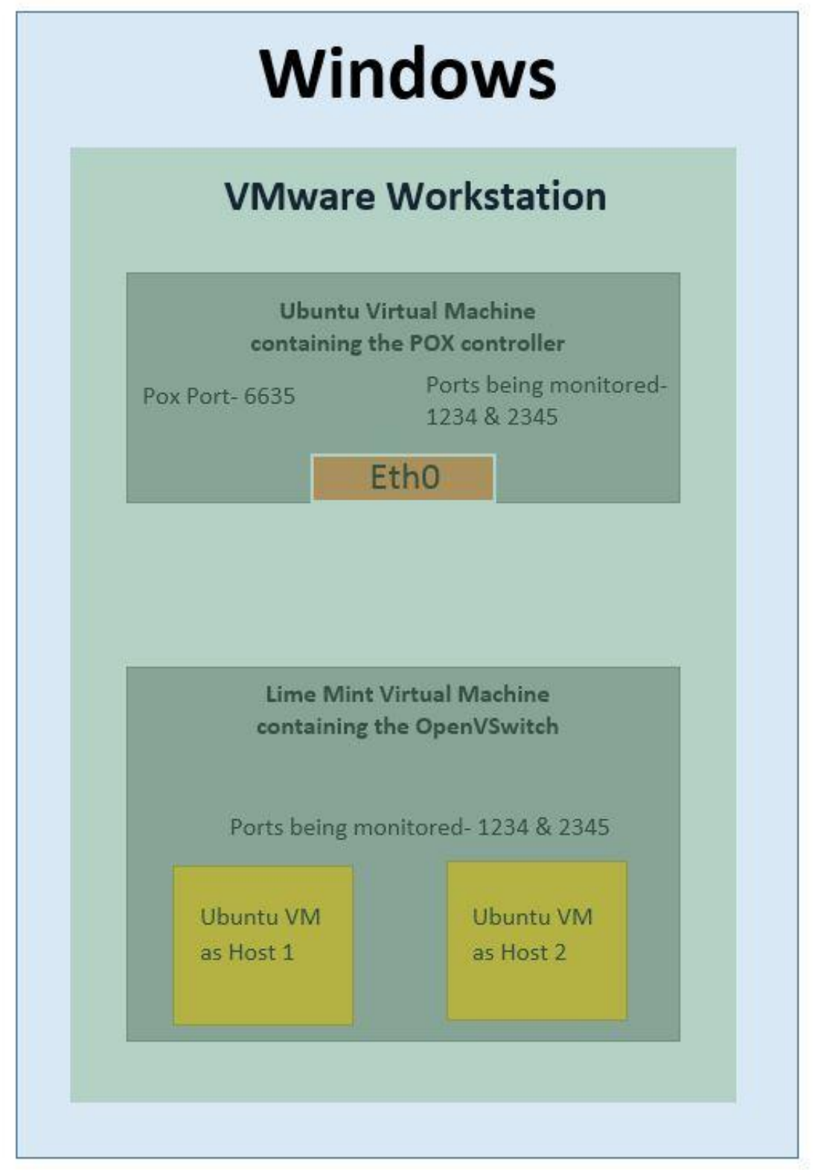
SDN:

Software Defined Network has the feature of decoupled control plane and data plane. SDN possess an irtificial intelligence control module. It control all the components it is connected with.



The controller is intelligent but it is not intelligent enough to act when certain security events like Dos, Ddos and other attacks take place. This is the key area of focus which evolved in the implementation of a dynamically reconfigurable controller.

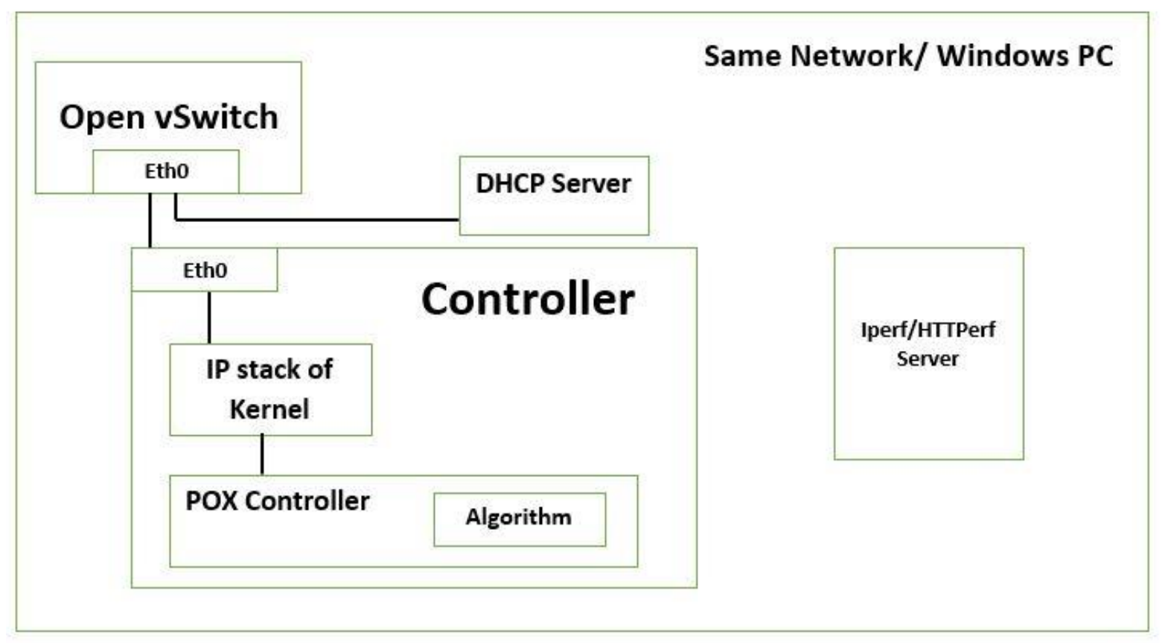
Methodology


Windows
VMware Workstation
Ubuntu Virtual Machine containing the POX controller
Pox Port- 6635
Ports being monitored- 1234 & 2345
Eth0
Lime Mint Virtual Machine containing the OpenVSwitch
Ports being monitored- 1234 & 2345
Ubuntu VM as Host 1
Ubuntu VM as Host 2

Key Steps involved in the project.

- Determine a security attack faster than the traditional detection speed
- Determine the security attack before it even hits the targeted network or host
- Shutdown the host / remote host / remote network from which the attack is generated before the attack even reaches the targeted server or network.

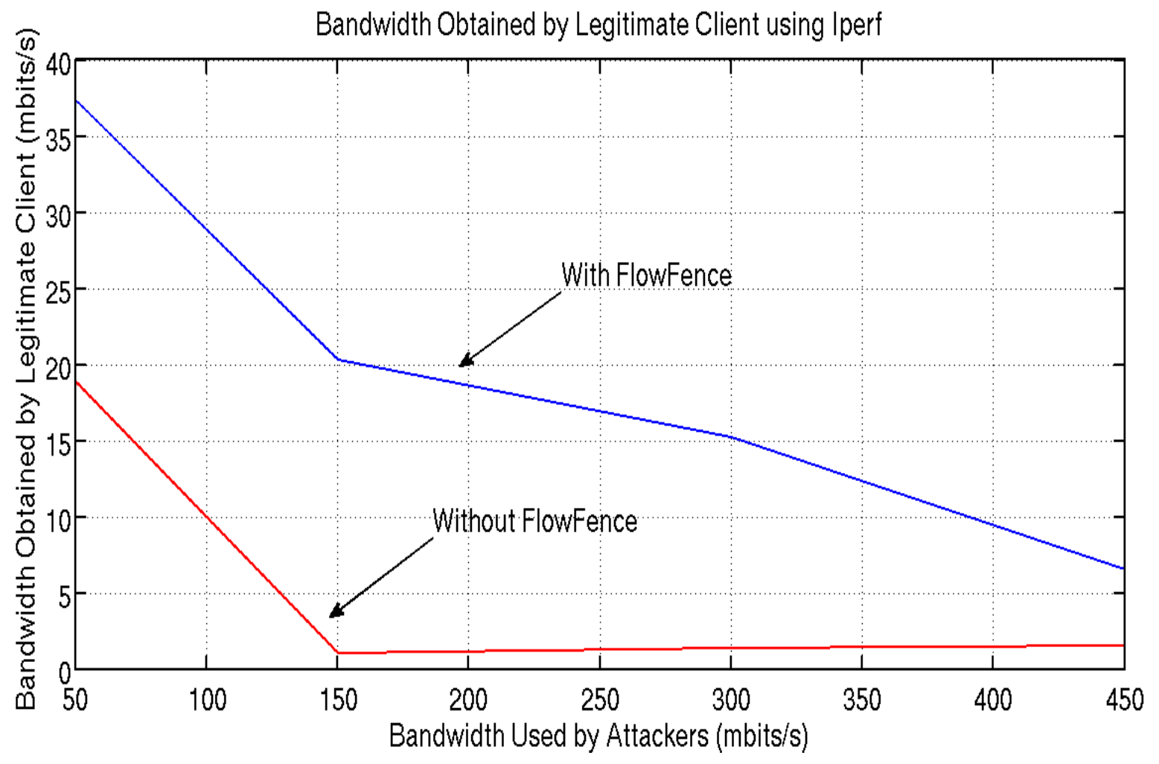
The outcome expected out of this project is a simple to configure yet complex in operation network that is capable of detecting the incoming security attacks and isolate the source network from which the attacker is being intruding the network in question. Then if the network from which the attacker is intruding from is a public network then targeting the attacker a virus or some mechanisms to shut down the remote attacker or drop all the packets from the attacker's host is taken place. If the network from which the attacker is attacking is a private network then measures are taken to shut down the remote network itself.



- Determine a security attack faster than the traditional detection speed
- Determine the security attack before it even hits the targeted network or host
- Shutdown the host / remote host / remote network from which the attack is generated before the attack even reaches the targeted server or network.
- Use the controller of the software defined networks to shut down the network from which attack originated.
- Do frequent sampling of the network and perform differential analysis and update the database of normal packet flow type.
- Create an artificial intelligence module that updates its own database and its own operating methods or procedure to detect the incoming attack

Results

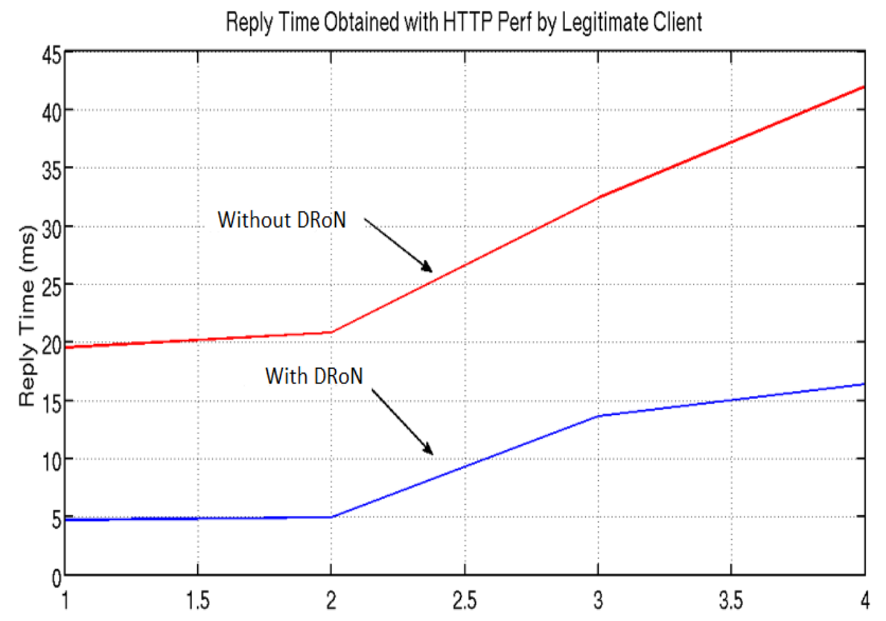
The testing of the algorithm was done using a series of methods. The most important were the iperf and the HHTPerf.



An iperf server was hosted in a separate server outside the current network. The iperf clients are run in the host level 2 VMs. When the test is run, the time for each measurement is given as 1 second. That is the time windows is given as 1 second and the test is run for 40 seconds. So now the bandwidth measurement for 40 seconds is got with data for each second. The data sent exceeded the threshold for congestion when it crosses few seconds of iperf testing.

```
ovs-vsctl -- set Port $1 qos=@fenceqos -- --id=@fenceqos
create QoS type=linux-htb other-config:max-
rate=10000000 queues:0=@queue0 queues:1=@queue1
-- --id=@queue0 create Queue other-config:min-
rate=1000000 other-config:max-rate=1000000 -- --
id=@queue1 create Queue other-config:min-
rate=8000000 other-config:max-rate=8000000
```

The above is an example command that the open vSwitch executes to set the queue bandwidth and assign QoS thereby doing it.



The connection block and the reply block of the output will show light on the statistics that vary on using the DRoN Algorithm module in the POX controller

Summary

This project aims a building an intelligent network which would stay secured and has a zero down time. These networks have an increased security feature that would make it more impeccable. This network can be implemented in places which needs high security like military, and space machines. The future plans of this network is that extending the artificial intelligence capability to automatically adjust the bandwidth and also other Qos and Cos parameters.

Key References

[1] Chun-Jen Chung; Khatkar, P.; Tianyi Xing; Jeongkeun Lee; Dijiang Huang, "NICE: Network Intrusion Detection and Countermeasure Selection in Virtual Network Systems," Dependable and Secure Computing, IEEE Transactions on , vol.10, no.4, pp.198,211, July-Aug. 2013.

[2] M.-K. Shin, K.-H. Nam, and H.-J. Kim, "Software-defined networking (sdn): A reference architecture and open apis," in 2012 International Conference on ICT Convergence (ICTC), Oct 2012, pp. 360–361.

[3] R. Thomas, B. Mark, T. Johnson, and J. Croall, "NetBouncer: Client-Legitimacy-Based High-Performance DDoS Filtering," Proc. DARPA Information Survivability Conf. and Exposition, vol. 1, pp. 14- 25, Apr. 2003

Acknowledgements

The authors wish to thank Dr. Greg Bernstein for his assistance in the field of networking and his expertise in guidance. We also thank Dr. Nadir Mir for his simulating suggestions and encouragement in the field of networking. At the end we thank all those engineers who have given their contribution to make the society better.