INVESTIGATING SDN AS A SECURITY TOOL

Karan Desai
Rutgers University

Openflow in Security
Rule Based Forwarding
Anti Arp Poisoning Switch
Further implementation

Why SDN?

- SDN allows for experimentation in optimizing and configuring how the network functions.
- Additionally, SDN can be controlled using commodity server hardware
- This flexibility is network design is in part accomplished by separating the **switch's control plane from the data plane**
- Its benefits over the existing AS model
- Having this new level of control can be of great benefit to security engineers, and we will cover some potential use cases for SDN and information security

Security Benefits

- Your networking equipment can take care of the security for you . Our example of this : the anti ARP poisoning logic
- Greater control (and more work) for the administrator

(Does it violate end-to-end??)

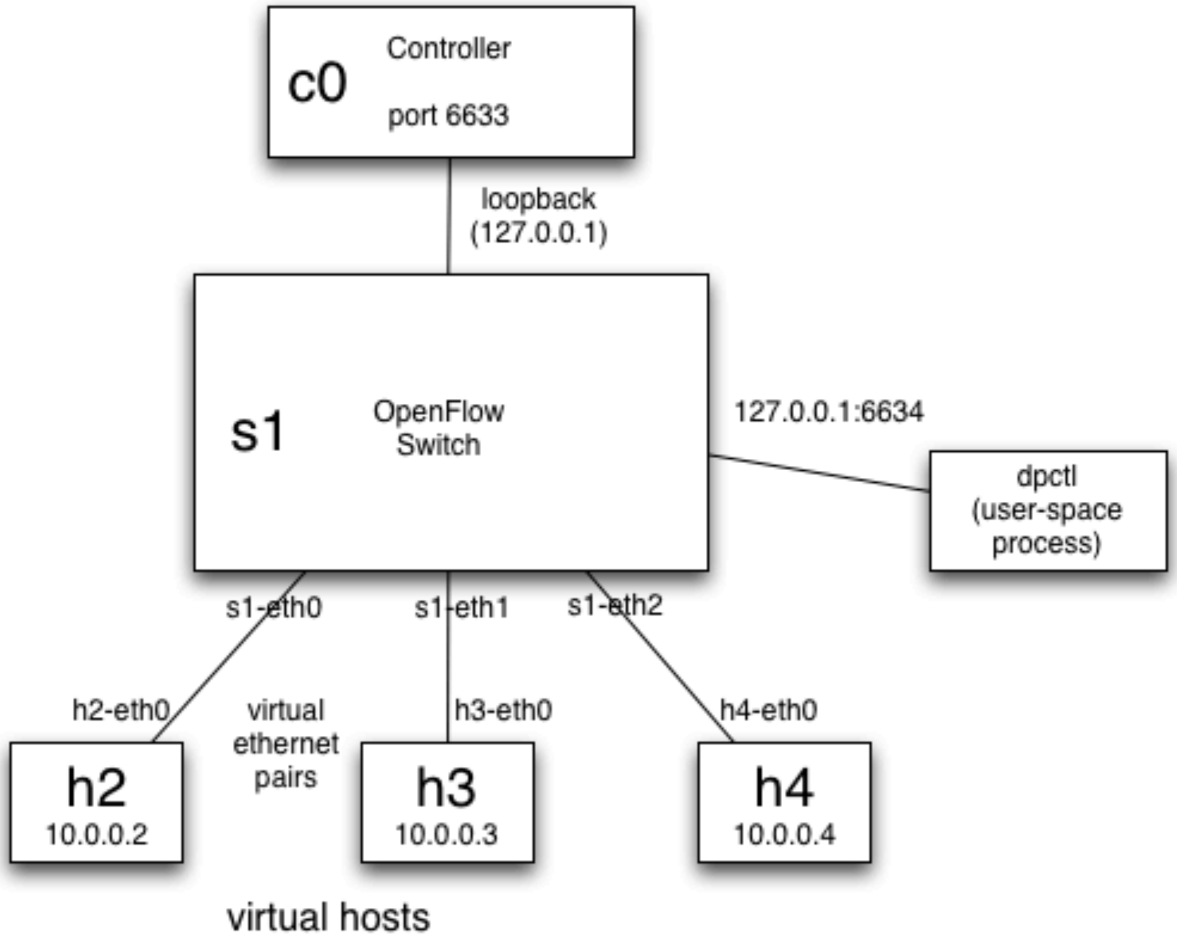
- Logic/control easy to program and change. Developing standards like openflow.
- Switch is now working on upper layers too
- Early detection
- It can not only detect but trigger appropriate actions
- Can be made generic to applications

Implementation

- Openflow (1.0) , Mininet , OpenVSwitch
- POX controller (python)
- Virtualization – Vmware
- SSH terminal from OS X

The further implementation relies on more core functions in the upcoming versions of openflow/ openswitch. More drastically portrayed by Cisco's upcoming SDN OnePK. It will have a lot of features that we implement already built in.

Demo Configuration



Anti ARP poisoning

- The POX- openflow implementation does not set a timeout for any of the flow tables.
- We implement this and set this time to 5-10 seconds.
- The logic is triggered if a duplicate mapping is found
- Demo is just alerting – You can do much more

Rule Based Forwarding

- Consider host H2 as the server
- H3 is the client
- H3 wants to use WGET (HTTP call) and get some file/information
- We have defined a rule that does not allow H3 to make http calls to H2 . It can still do everything else. Example - ping it (ICMP)
- If this flow is found – our switch tries to block it
- You can simply NOT forward the packet
- How is it useful ?

Further Implementation Example

- Consider the following scenario
- You have an employee that use mobile devices, laptops, etc. All are previously registered devices. They access workspace environment from all types of devices
- By default he does not have access to any files on your specific server
- Now, using the rule based idea you can implement is if employee somehow reaches there (faulty implementation), your implementation can detect this early on and send an alert
- Or if the employee keeps on printing iteratively or keeps on trying to access files he is not supposed to, your logic and log him out of the system from all devices immediately and follow up with appropriate procedures
- Remember, all this triggering is not happening on the upper layers but is embedded into devices like the switch

REFERENCES

[1] Open Networking Foundation, "OpenFlow Switch Specification 1.3.0," 25 June 2012. [Online].

[2] "POX Wiki," Stanford University, [Online]. Available: https://openflow.stanford.edu/display/ONL/POX+Wiki . [Accessed 12 12 2012].

[3] I. Aggarwal, Implementation and Evaluation of ELK, an ARP scalability enhancement, 2011.

[4] S. Shin and G. Gu, CloudWatcher: Network Security Monitoring Using OpenFlow in Dynamic Cloud Networks (or: How to Provide Security Monitoring as a Service in Clouds?), 2012.

[5] E. Kissel, G. Fernandes, M. Jaffee and M. Swany, Driving Software Defined Networks with XSP, 2012.

[6] G. Yao, J. Bi and P. Xiao, Source Address Validation Solution with OpenFlow/NOX Architecture, 2011.

[7] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker and J. Turner, OpenFlow: Enabling Innovation in Campus Networks, 2008.

[8] Adrian Crenshaw , Security and Software Defined Networking:
Practical Possibilities and Potential Pitfalls (2012) [online – irongeek.com]

[9] RSA 2013, Keynote – SDN Security , Cisco , John Chambers