OrchSec: An Orchestrator-Based Architecture For Enhancing Network Monitoring and SDN Control Functions

9 May 2014

Dr.-Ing. Kpatcha Bayarou

Head, Mobile Networks

Fraunhofer SIT

Kpatcha.bayarou@sit.fraunhofer.de

CASED

Fraunhofer SIT

Outline

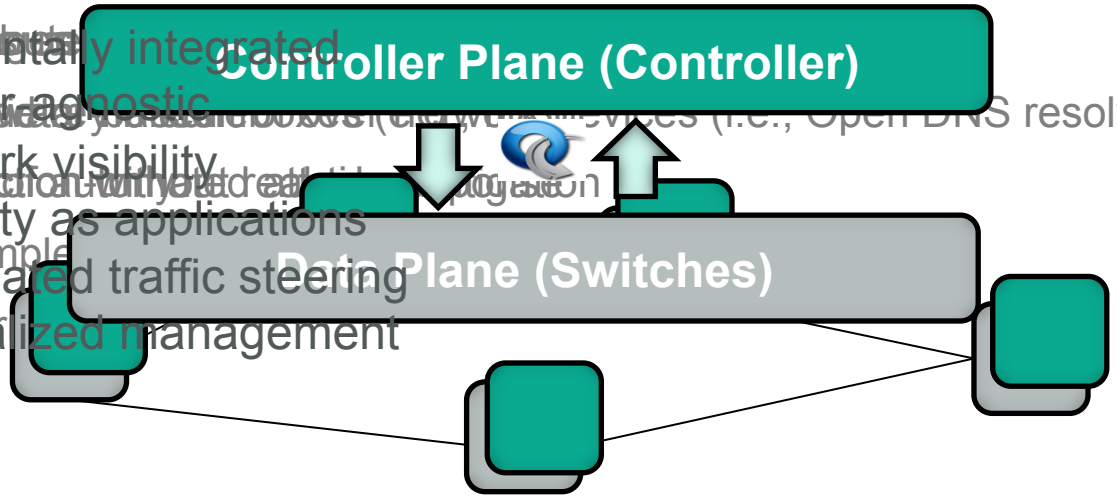
- Introduction
- Architectural Design
- Orchestrator-Based Security
- Experimental Examination
- Conclusion

Introduction

- Many protocols of current Internet expose a set of vulnerabilities.
- One of these protocols is the Address Resolution Protocol (ARP).
  - ARP is Stateless
  - It provides no mechanisms for reply authentication
- These vulnerabilities led to threats such as:
  - ARP spoofing / cache poisoning
  - CAM table overflow
  - Null address attack
- Services in the Internet are provided using a client-server model.
- This later led to threats such as:
  - Denial of Service (DoS) / Distributed Denial of Service (DDoS)
  - Domain Name System (DNS) amplification

Introduction

- Traditional approaches against these threats have their drawbacks.
- **Challenges in traditional ARP security (SDN) address these challenges?**
  - Horizontally integrated
  - Vendor agnostic
  - Network visibility
  - Security as applications
  - Automated traffic steering
  - Centralized management



ARP Security Challenges

- Changes in the network host
- Hardware-based
- Detection-only
- Incomplete threat coverage
- Proprietary

DoS Security Challenges

- Decentralized network management
- Proprietary middle-boxes (e.g., IDS)
- Detection without mitigation
- Performance bottlenecks

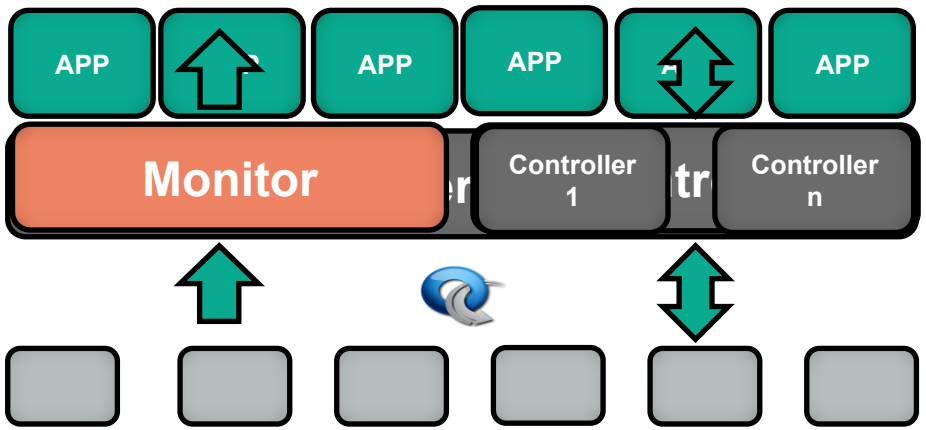
DNS Amplification Security Challenges

- Relies on attack prevention, with no reactive mitigation
- Limited or no control over network devices
- Lack of Automated attack response

Introduction

Security-Centric SDN

- Using the features provided by SDN to improve or enable security in traditional networks.
- To solve these problems, the following adjustments can be considered:
  - Tight coupling of SDN applications to the controllers
  - Tight coupling of network monitoring and control functions
  - Using multiple controllers for a more reliable and diverse architecture



Outline

- Introduction
- **Architectural Design**
- Orchestrator-Based Security
- Experimental Examination
- Conclusion

Architectural Design – Architectural Requirements

Secure & reliable SDN architecture:

- Using multiple controller instances for reliability and diversity.

Flexibility in application development:

- Develop applications using a Northbound API.

Decoupling control & monitoring functions:

- Decouple network monitoring from control functions to reduce the overhead on the controller.

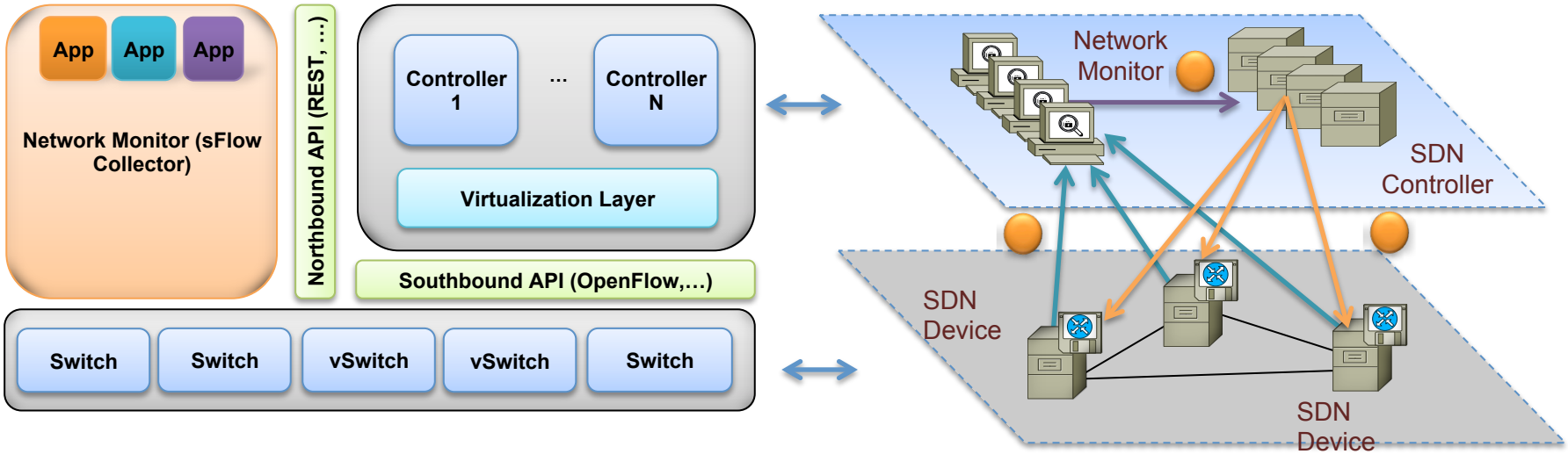
Providing high-resolution attack-detection:

- Provide more information as an input for attack detection.
- Detect attacks that require access to all packets.

Architectural Design – Proposed Architecture I

Requirements
- Secure & reliable SDN architecture
- Flexibility in application development
- Decoupling control & monitoring functions
- Providing high-resolution attack detection

First Iteration: Sampling-based Security



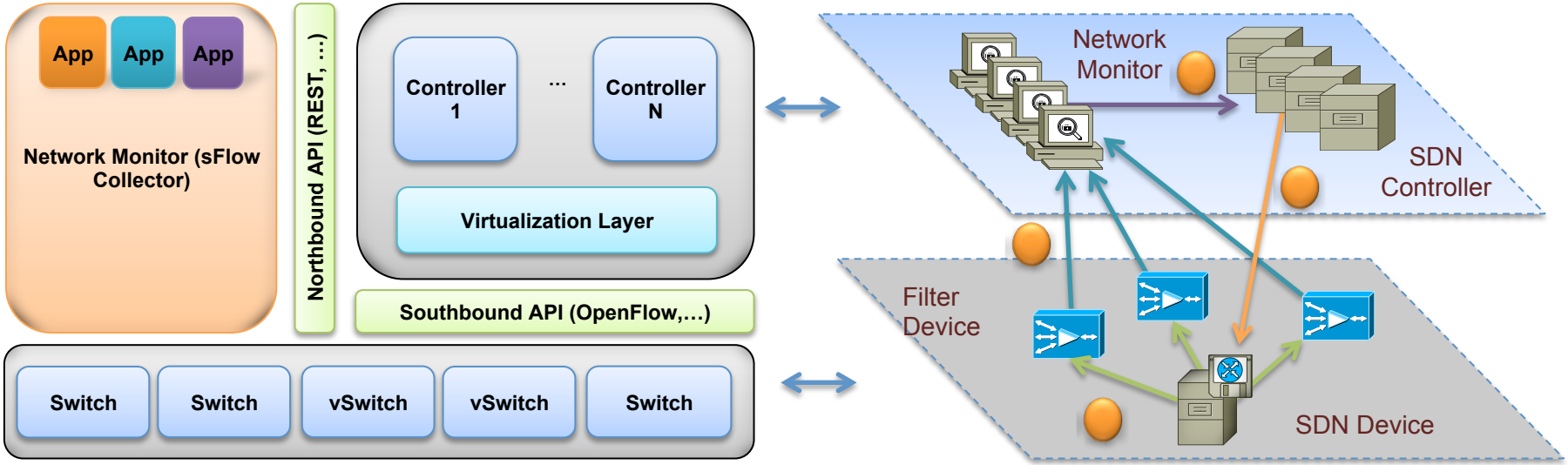
Pros
- Northbound applications
- Multiple controllers
- Decoupled monitoring & control

Cons
- Flow-shortening
- Flow-reduction

Architectural Design – Proposed Architecture II

Requirements
- Secure & reliable SDN architecture
- Flexibility in application development
- Decoupling control & monitoring functions
- Providing high-resolution attack detection

Second Iteration: High Resolution Sampling



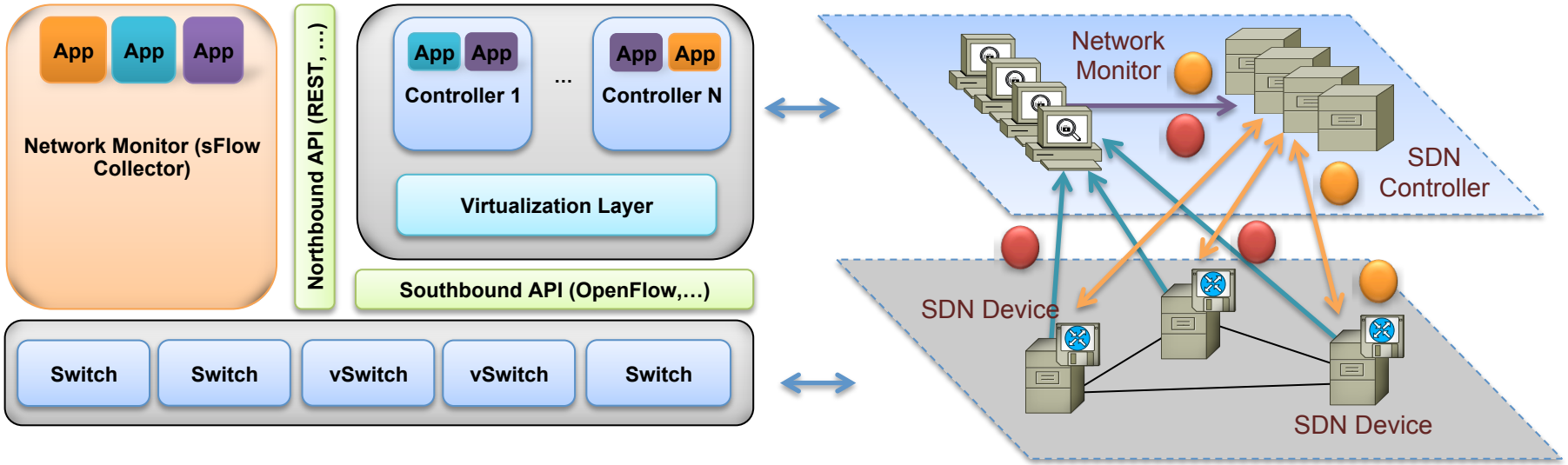
Pros
- Higher sampling budget
- Northbound applications
- Multiple controllers
- Decoupled monitoring & control

Cons
- Flow shortening was not completely solved

Architectural Design – Proposed Architecture III

Requirements
- Secure & reliable SDN architecture
- Flexibility in application development
- Decoupling control & monitoring functions
- Providing high-resolution attack detection

Third Iteration: Delegating Attack Detection



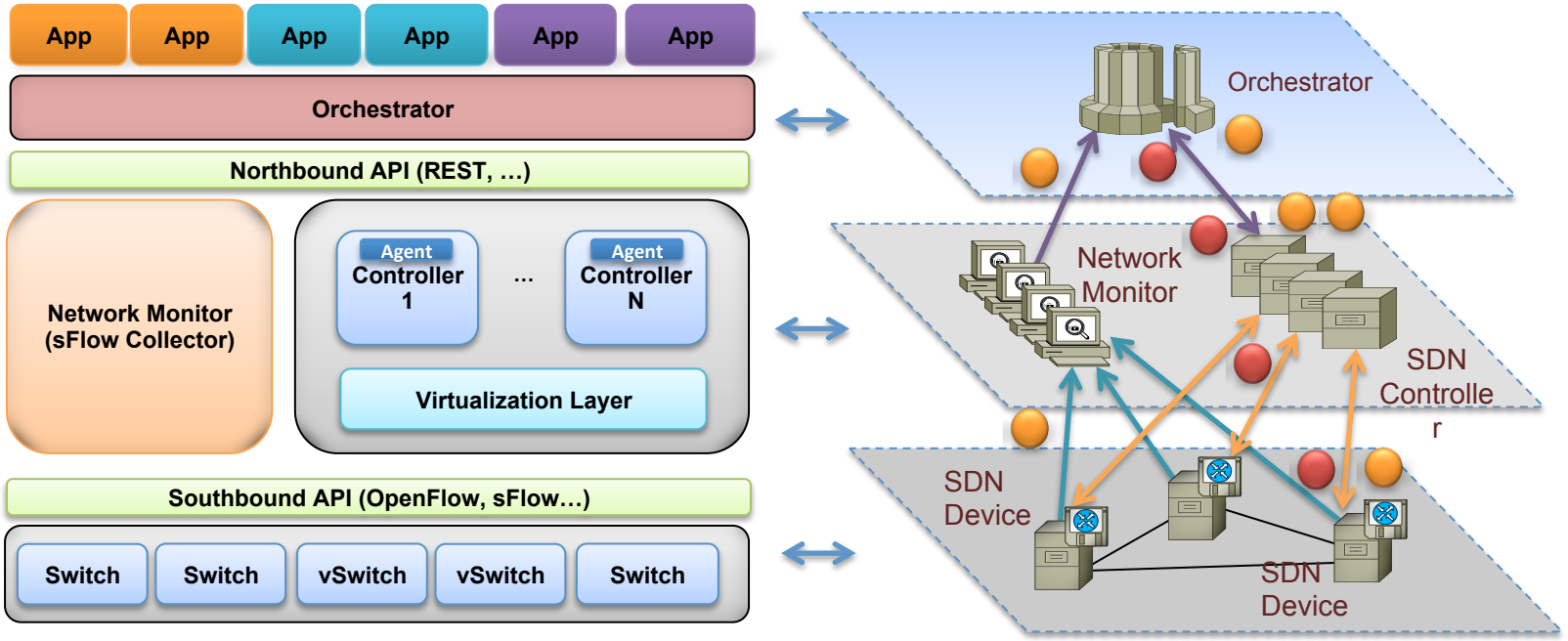
Pros
- High resolution attack detection using delegation
- Multiple controllers
- Decoupled monitoring & control

Cons
- Tightly-coupled applications

Architectural Design – Proposed Architecture IV

Requirements
- Secure & reliable SDN architecture
- Flexibility in application development
- Decoupling control & monitoring functions
- Providing high-resolution attack detection

Orchestrator-based Architecture



Pros
- High resolution attack detection
- Northbound applications
- Multiple controllers
- Decoupled monitoring and control

Cons
- Overhead for high resolution attack detection

Outline

- Introduction
- Architectural Design
- **Orchestrator-Based Security**
- Experimental Examination
- Conclusion

Orchestrator-Based Security – DNS Amplification Security

DNS Attack

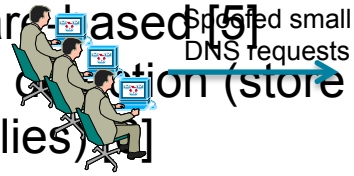
- Flooding-based DNS amplification DD

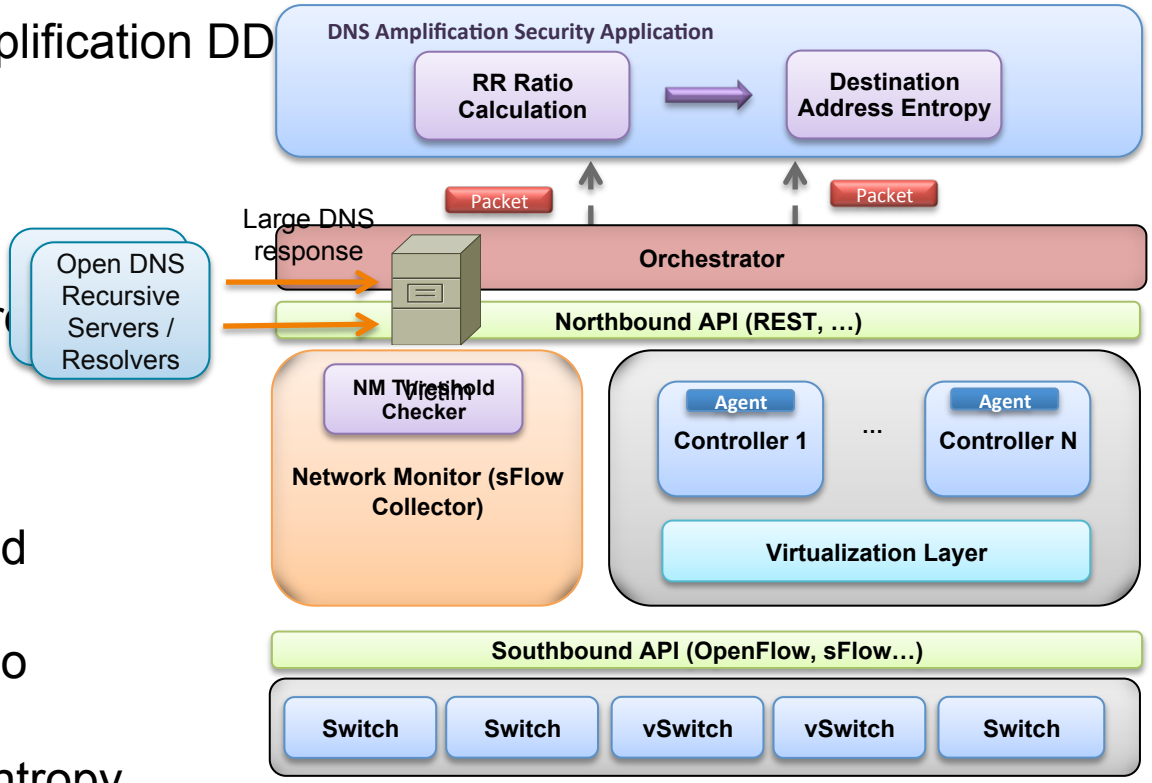
Related Work

- Hardware-based [5]
- Stateful detection (store r
  and replies) [6]

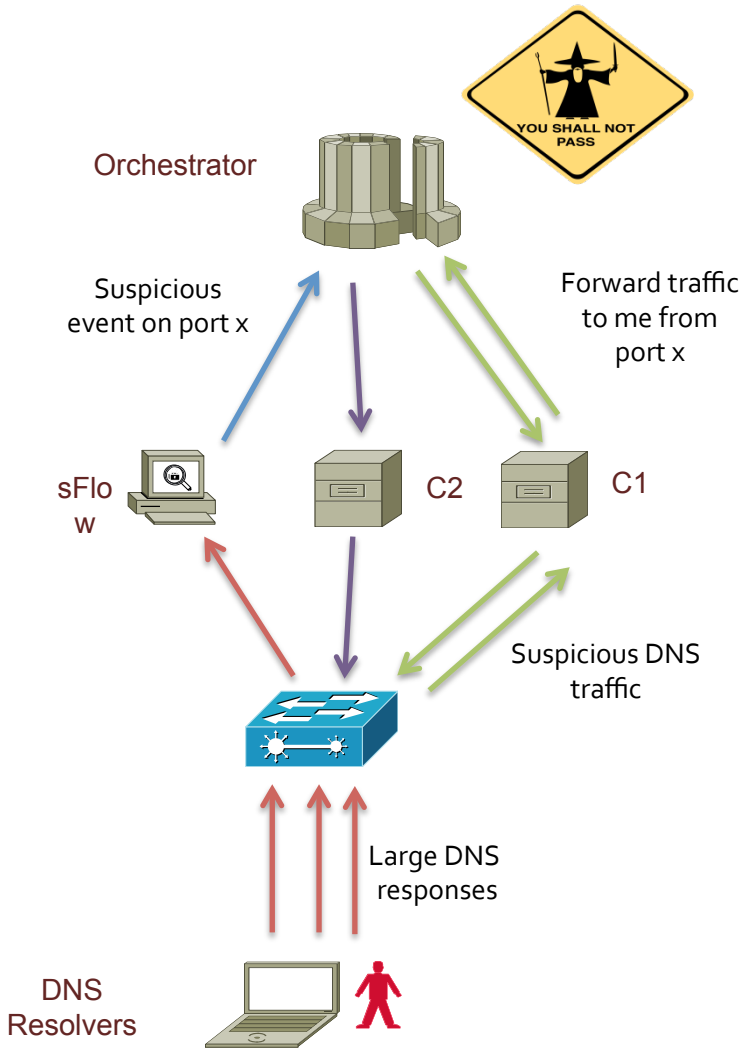
Security Blocks

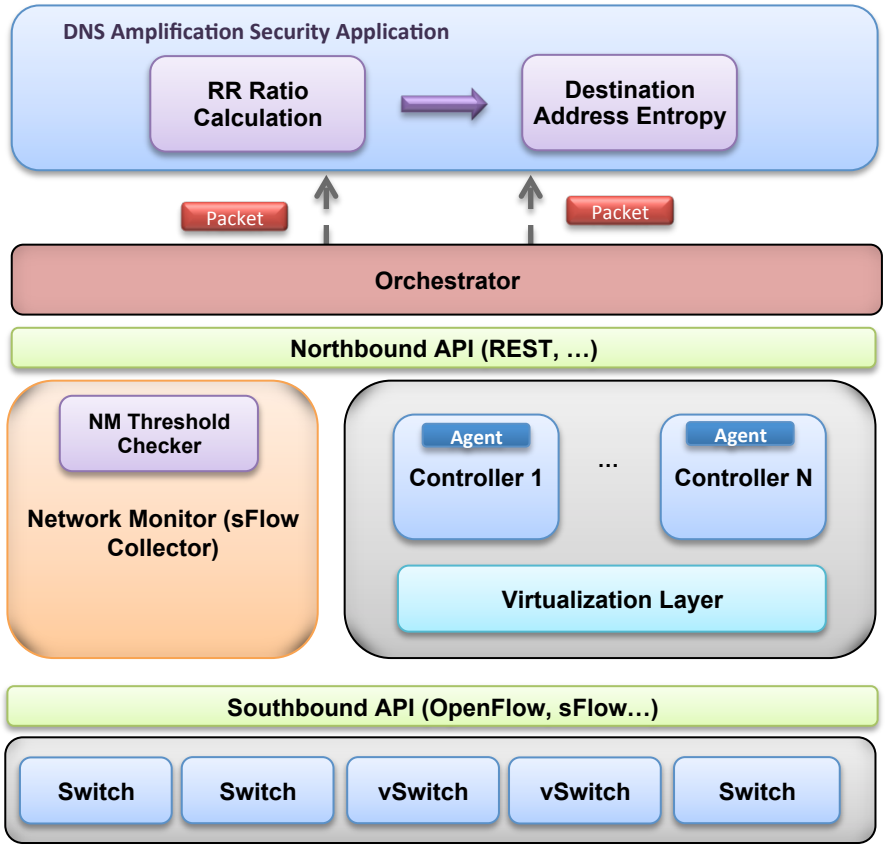
- Network Monitor threshold checker
- Received-Reply (RR) ratio calculation
- Destination IP address entropy





Orchestrator-Based Security – DNS Amplification Security





Outline

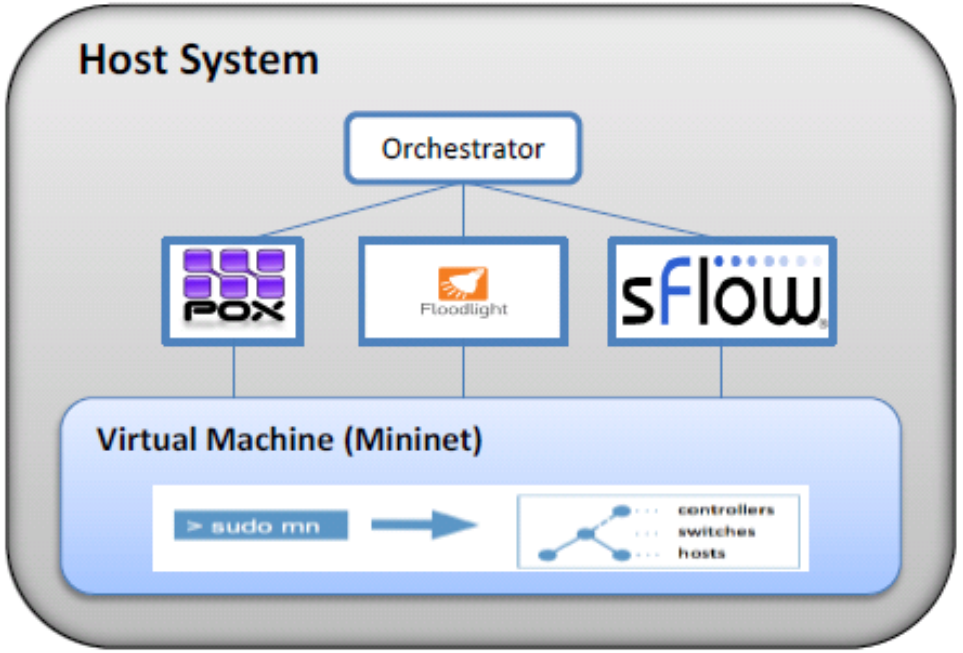
- Introduction
- Architectural Design
- Orchestrator-Based Security
- **Experimental Examination**
- Conclusion

Experimental Examination – Testing Enviroment

Host System
- Ubuntu 12.04 LTS
- Intel Core i7-3630QM
- 8 GB RAM

Controllers
- Floodlight
- POX

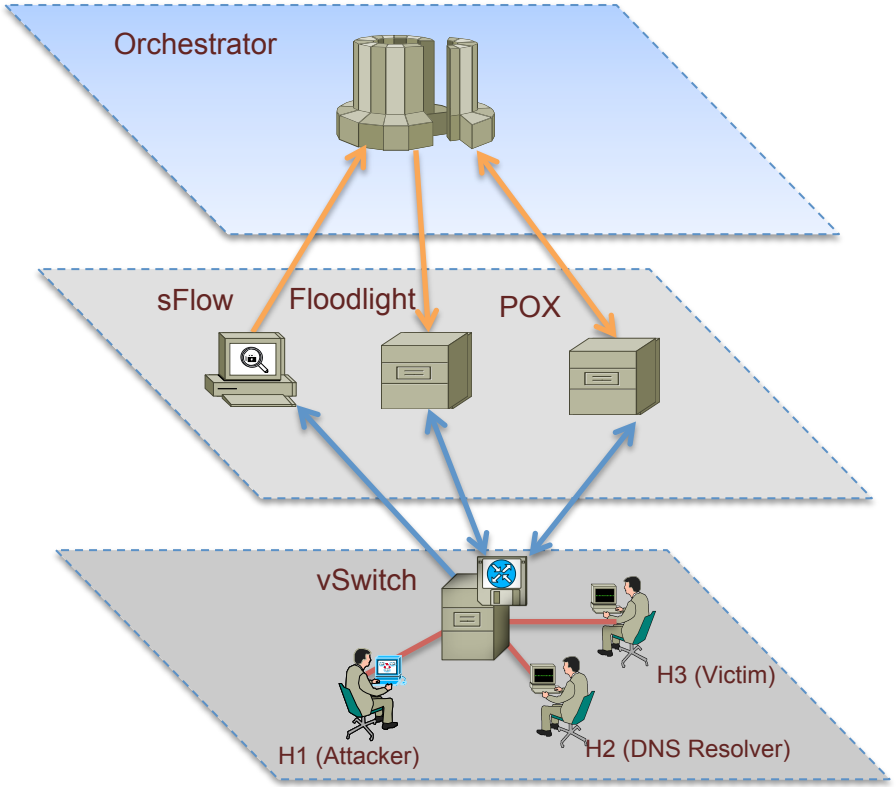
Network Monitor
- sFlow



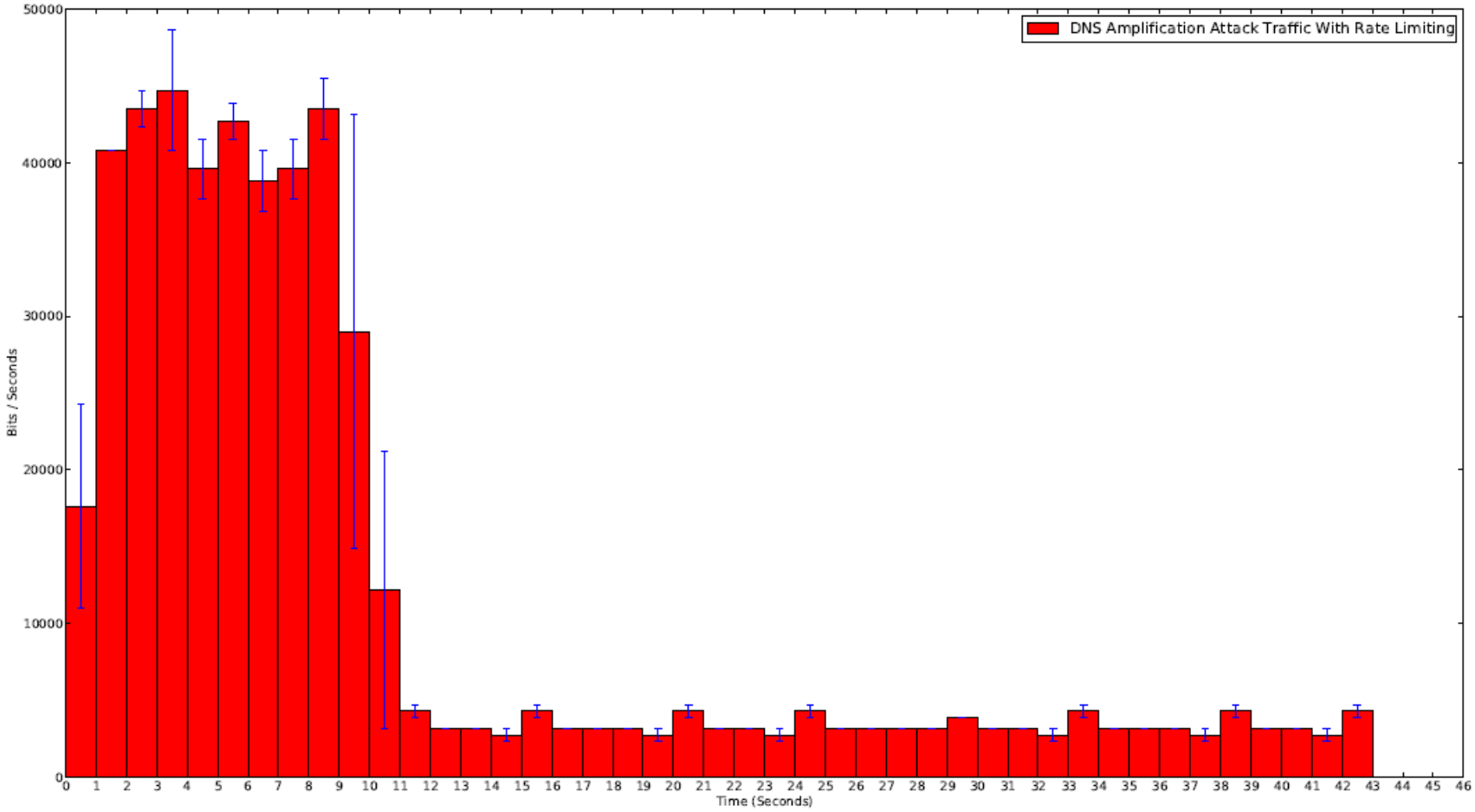
Virtualization
- Virtualbox VM (with a NAT adapter and a host-only adapter)
- Mininet

Experimental Examination – DNS Security Experiment I

DNS Amplification Experiment



Experimental Examination – DNS Security Experiment II



Conclusion & Future Work

Conclusion

- SDN provides features that can enhance network security.
- However, SDN has some architectural deficiencies when it comes to security.
- To address these deficiencies, an Orchestrator-based architecture is proposed.
- The proposed architecture provides:
  - Reliability through the use of multiple controllers
  - Flexibility in application development
  - Decoupled monitoring and control functions
  - High-resolution attack detection
- Using the proposed architecture, applications to mitigate against ARP cache poisoning, DoS /DDoS and DNS amplifications were developed.
- The proposed architecture provides flexibility at the cost of increased latency.

Future work

- Orchestrator-agents support
- Further attack analysis
- Threshold optimization
- Attack mitigation strategies

Contact for specific questions

Fraunhofer Institute for Secure Information Technology (SIT)
Rheinstr. 75, Darmstadt, Germany

- Rahamatullah Khondoker rahamatullah.khondoker@sit.fraunhofer.de
- Ronald Marx ronald.marx@sit.fraunhofer.de

RWTH Aachen University Aachen, Germany

- Adel Zaalouk adel.zaalouk@rwth-aachen.de

References

1. Carnut, M., and J. Gondim. "ARP spoofing detection on switched Ethernet networks: A feasibility study." Proceedings of the 5th Simposio Seguranca em Informatica. 2003.
2. Gao Jinhua; Xia Kejian, "ARP spoofing detection algorithm using ICMP protocol," Computer Communication and Informatics (ICCCI), 2013 International Conference on , vol., no., pp.1,6, 4-6 Jan. 2013
3. Kim, Myung-Sup, et al. "A flow-based method for abnormal network traffic detection." *Network Operations and Management Symposium, 2004. NOMS 2004. IEEE/IFIP*. Vol. 1. IEEE, 2004.
4. Jun, Jae-Hyun, Hyunju Oh, and Sung-Ho Kim. "DDoS flooding attack detection through a step-by-step investigation." *Networked Embedded Systems for Enterprise Applications (NESEA), 2011 IEEE 2nd International Conference on*. IEEE, 2011.
5. Sun, Changhua, Bin Liu, and Lei Shi. "Efficient and low-cost hardware defense against DNS amplification attacks." *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*. IEEE, 2008.
6. Kambourakis, Georgios, et al. "A fair solution to DNS amplification attacks." *Digital Forensics and Incident Analysis, 2007. WDFIA 2007. Second International Workshop on*. IEEE, 2007.