SDN Security

Brian Perry
PhD Student
University of California, Davis

Abhishek Gupta
PhD Student
University of California, Davis

S. Sedef Savas
PhD Student
University of California, Davis

Matt Bishop
Professor
University of California, Davis

Sean Peisert
Adjunct Assistant Professor
University of California, Davis

Key points:

- ***Security of what can control the switch is at least as important as security of the switch***
- ***You can gain some security using SDN, but you risk losing some security, too***
- ***Introducing powerful technology introduces risk of that power being used to attack — the threats may not come from where you expect!***

Background: SDN

Traditional network switches are difficult to upgrade:

- built-in routing algorithms

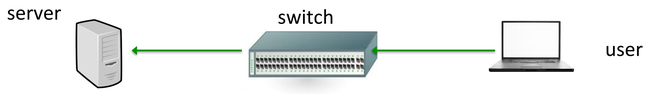
- manufacturer-specific APIs

Adapting to unexpected events is problematic

- updating services usually requires replacing the switch.

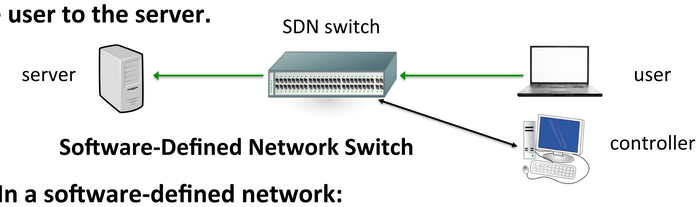
Software-defined networking enables switches to be reprogrammed on the fly.

- routing algorithms can adapt to external events and changes in services, protocols, and so forth are easily made.



Traditional Network Switch

In a traditional network, the switch controls the flow of data from the user to the server.



Software-Defined Network Switch

In a software-defined network:

- the controller can change the routing algorithms and rules on the fly

- the switch can be set to forward packets for which there are no rules to the controller

This has several advantages:

- **Switch interfaces use a standard set of API**
- **Applications can perform additional security functions in a way not possible in traditional networks**
- **Policies for handling flows can be dynamic**

Security Issues, Services, and Questions

So the SDN router can add to the security of the network. Along with this flexibility comes added risk.

- Authorization and authentication in the SDN switch are more complicated.

- E.g., commands reconfiguring the switch must come from a trusted, authorized source, as must updates.

SDN switches can provide several security services to protect their networks:

- **Generated statistics are more accurate than those supplied by ISPs — this can provide better data to an anomaly-based intrusion detection system (IDS)**
- **OpenFlow Random Host Mutation dynamically allocates a random virtual IP address mapped to the real IP address, hiding the real IP address.**
- **The controller can enforce dynamic access control policies based on flow-level information**
- **Applications can act as edge-based authentication gateways (OpenGate)**

Fun questions:

Q. How secure is the switch?

I.e., can the switch be attacked and compromised?

Two issues underlie this:

- ***Software vulnerabilities*. OpenFlow is by far the most widely-used software.**
  - **Several groups are currently checking it**
  - **We found: if the controller begins the setup as version 1, and then switches to version 2 in the middle of set-up, the exchange continues**
- ***Configuration Vulnerabilities.***
  - **If passive listening is on, impersonating the controller is possible;**
  - **if the connection to the controller is not secure, a man-in-the-middle attach is possible**

Our Examination

Q: How secure are the controller and updater?

I.e., can the computers that program the switch be compromised?

A likely vector—vulnerabilities in "ordinary computers" are much better known and exploitable.

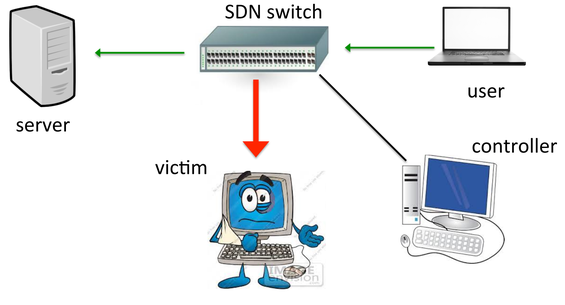
Focus on the controller; it's an ordinary computer

•Misconfigurations at the *host* level, rather than on the switch, can cause problems

•If we can impersonate the controller, that's just as good

Scenario #1: controller programs the switch to route packets through a hostile host that can read (or change) everything in the packet(s)

Scenario #2: use the switch as a denial of service tool



We gratefully acknowledge ther support of the GENI Projects Office through Contract #950012166, Prime Sponsor (NSF) Aard CNS-1344668