Security Considerations When Deploying a Software-Defined Network

Dylan Smyth

25-01-2021

Beyond IoT
2021

$ whoami

- Dylan Smyth
- Lecturer with CIT/MTU
- Area of research is Software-Defined Network Security

Contents

- Software-Defined Networking
- Security Opportunities
- Non-obvious Security Issues
- Conclusion

Software-Defined Networking

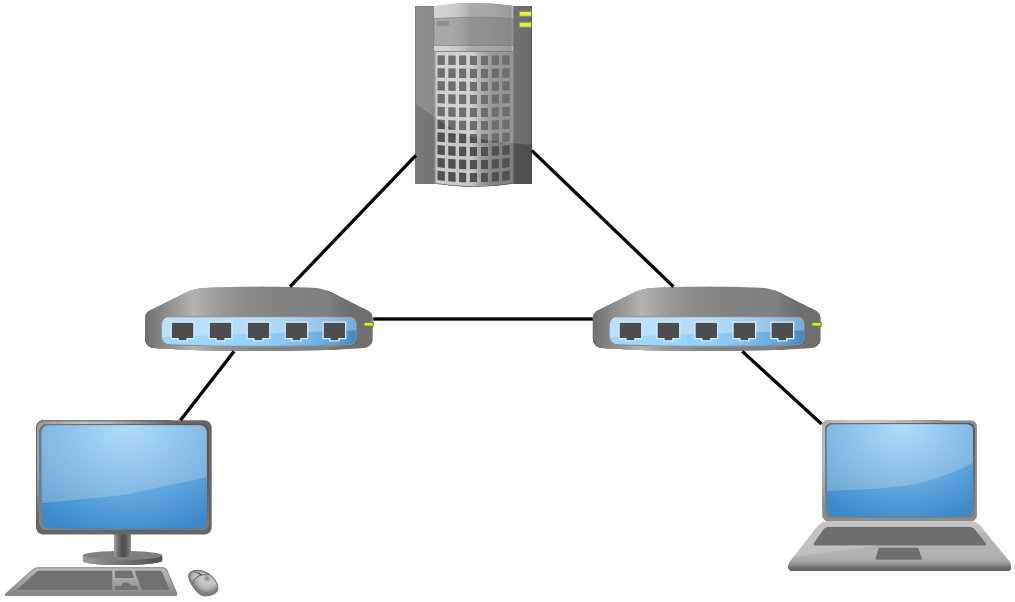
- Software-Defined Networking (SDN) defines a network architecture where network control is centralised

- In a conventional network, forwarding devices make decisions on traffic forwarding themselves

- In SDN, this decision making is centralised

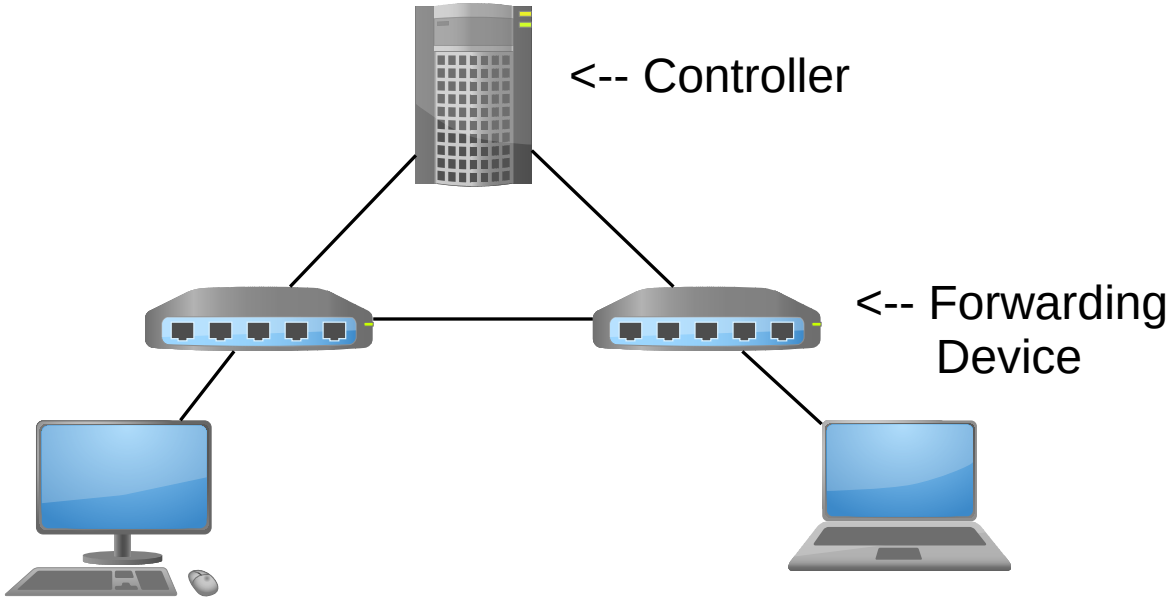
Software-Defined Networking

- A device known as a **controller** instructs **forwarding devices** on how to forward network traffic



Software-Defined Networking

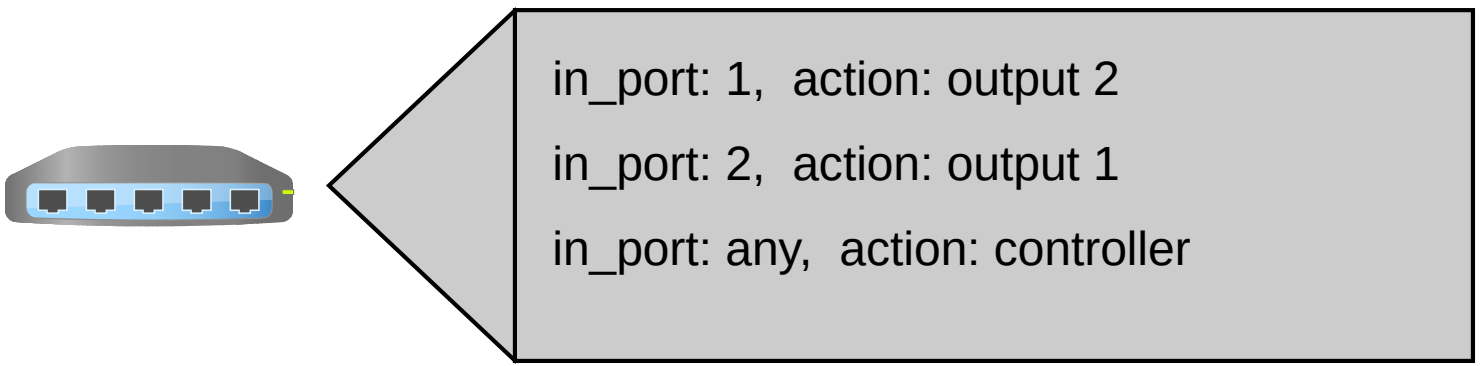
- A device known as a **controller** instructs **forwarding devices** on how to forward network traffic



Software-Defined Networking

- The forwarding devices (or *switches*) contain a list of forwarding rules in a flow table

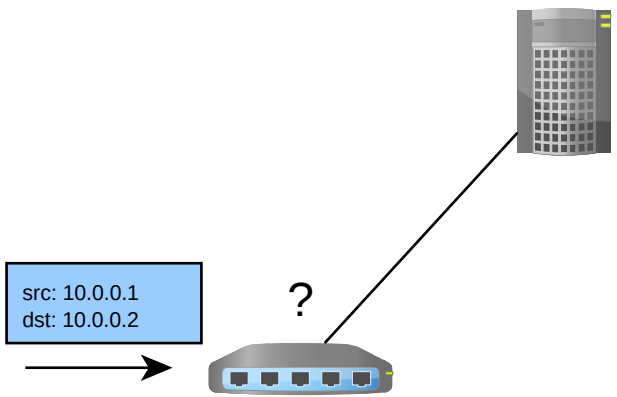
- The controller will populate this flow table with forwarding rules, allowing the switches to forward traffic

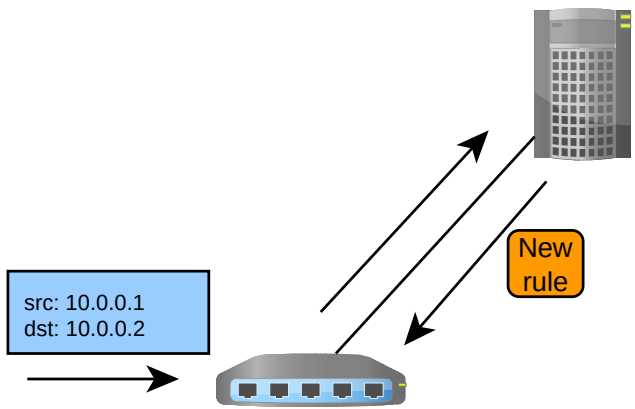


Software-Defined Networking

- If a packet cannot be matched to a forwarding rule then the switch will send that packet to the controller

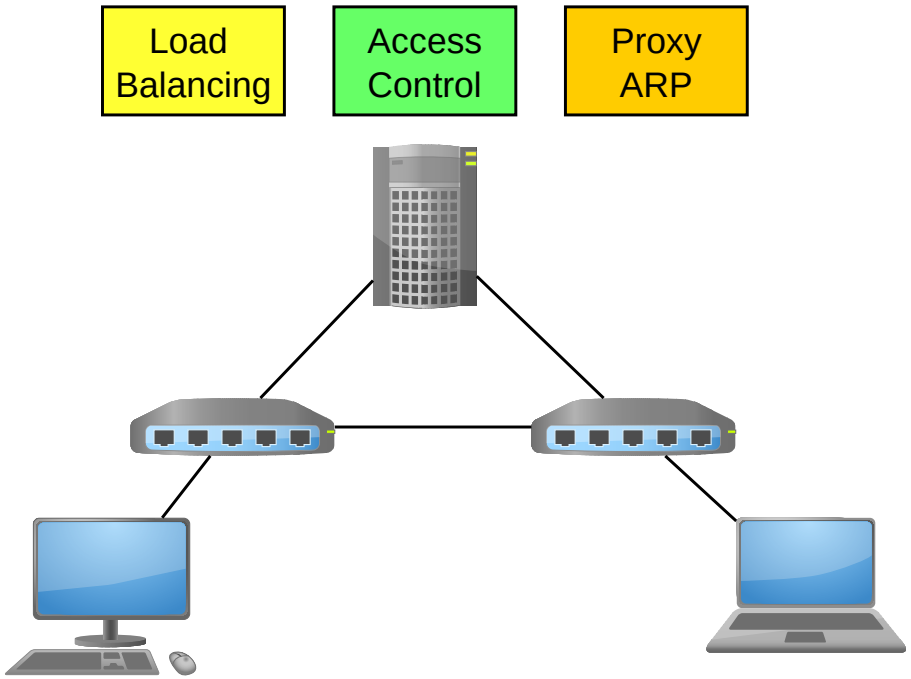
- The controller will then install a rule allowing the switch to forward the packet





Software-Defined Networking

- Centralised control allows applications to be used to influence forwarding decisions and network behaviour



Software-Defined Networking

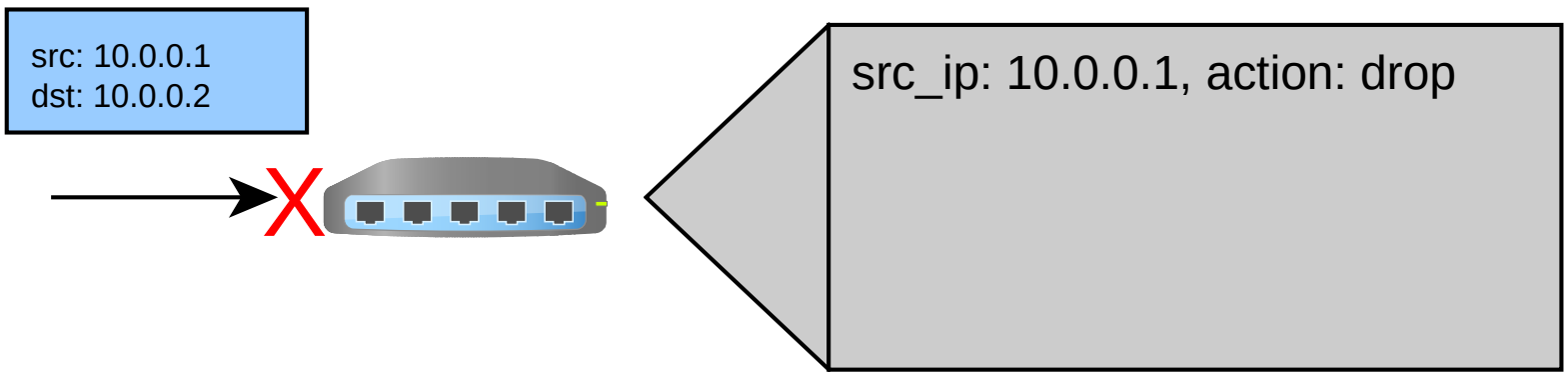
- This architecture would be a “classic” OpenFlow enabled SDN architecture [1]
- Hybrid SDN architectures exist [2]
- In a hybrid architecture the forwarding devices retain some decision making capability

Security Opportunities

Security Opportunities

- Every SDN switch can act as a firewall

- Forwarding rules can be used to enforce security policies



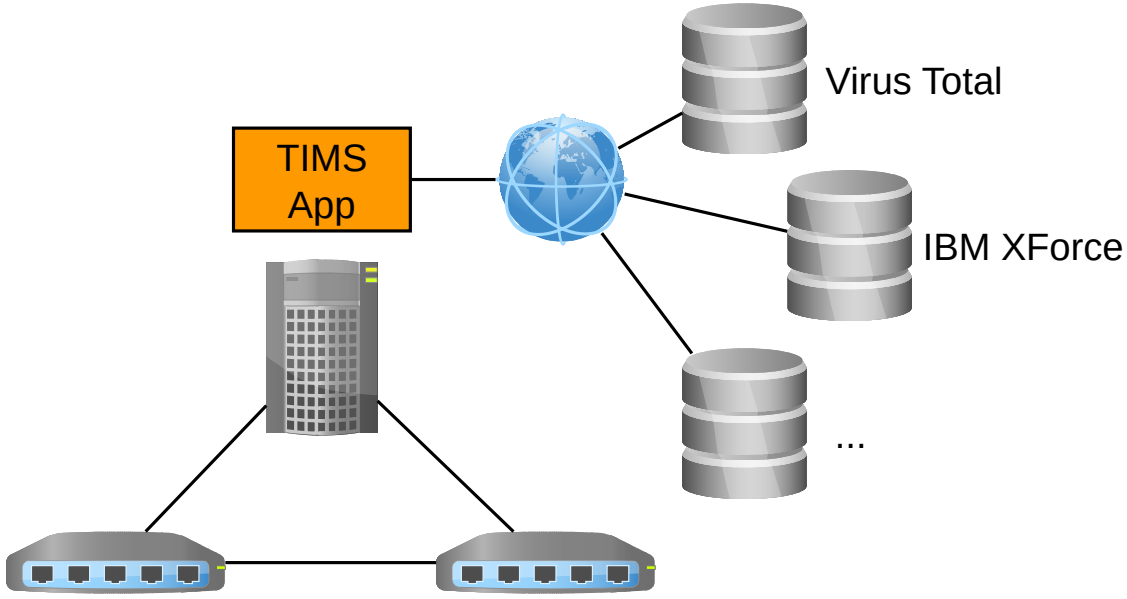
Security Opportunities

- The controller can insert, remove, and update rules across the network

- Access control can therefore be dynamically adjusted throughout the network in response to active attacks or new security policies

Security Opportunities

- A Threat Intelligence Management System (TIMS) could be integrated in the controller to help manage security rules



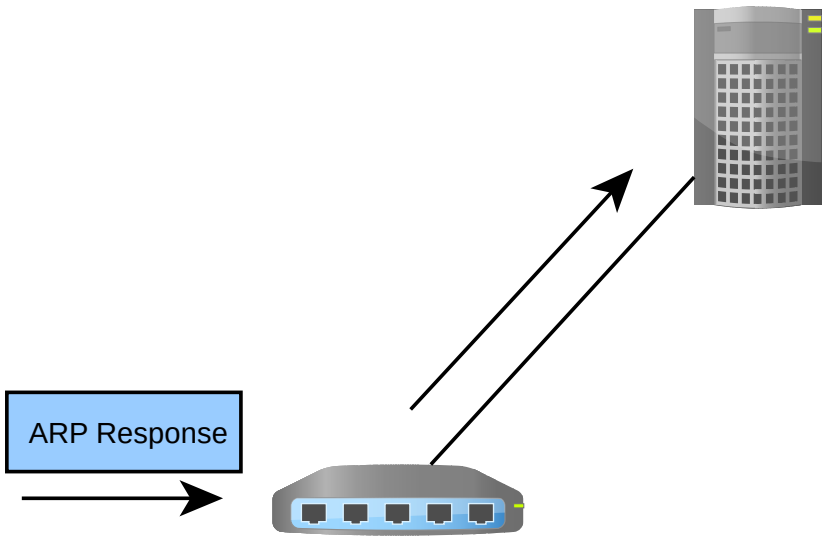
Non-obvious Security Issues

Non-obvious Security Issues

- Inexact forwarding rules can be exploited

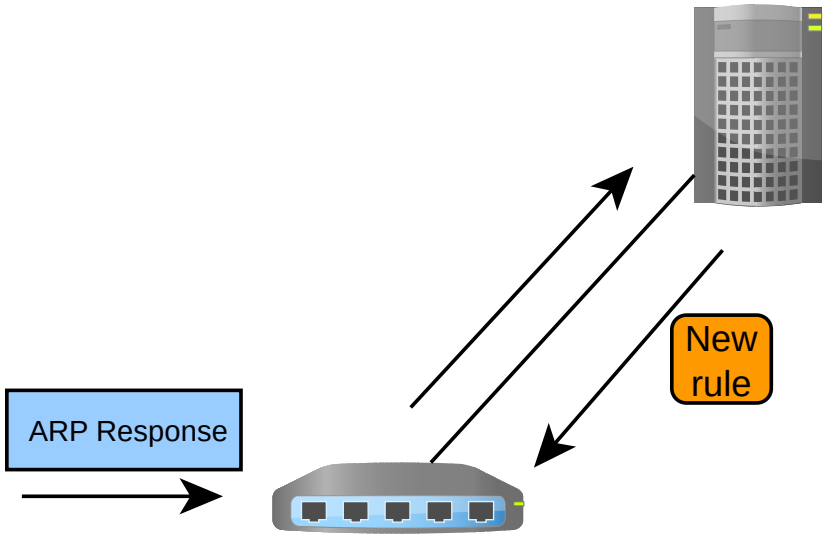
Non-obvious Security Issues

- For example, take a new ARP response arriving at a switch



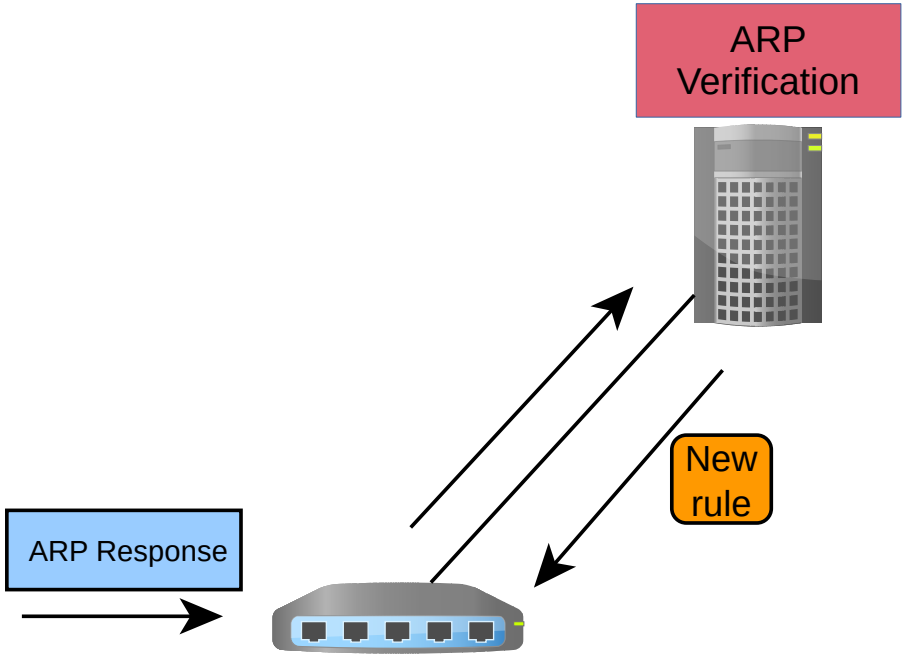
Non-obvious Security Issues

- The switch forwards the packet to the controller, which returns a forwarding rule



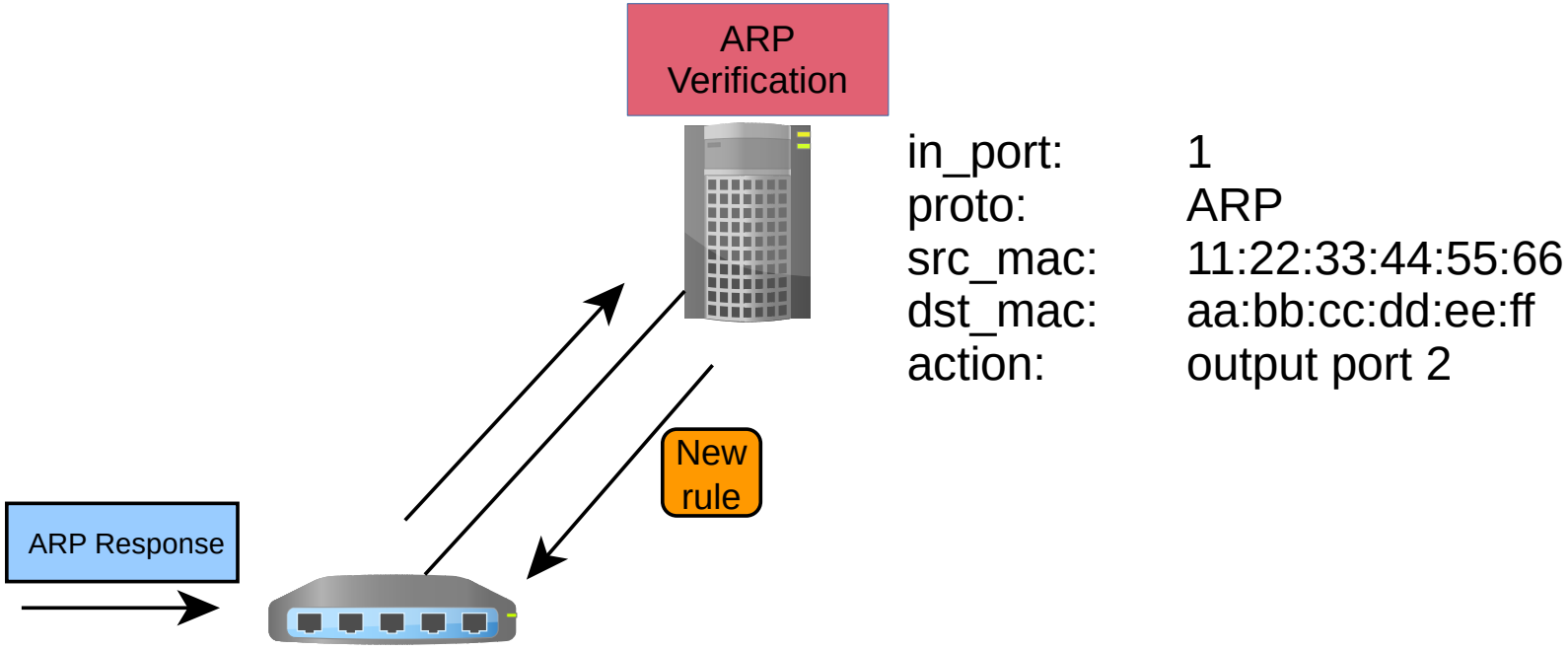
Non-obvious Security Issues

- The controller may have an app to detect ARP spoofing



Non-obvious Security Issues

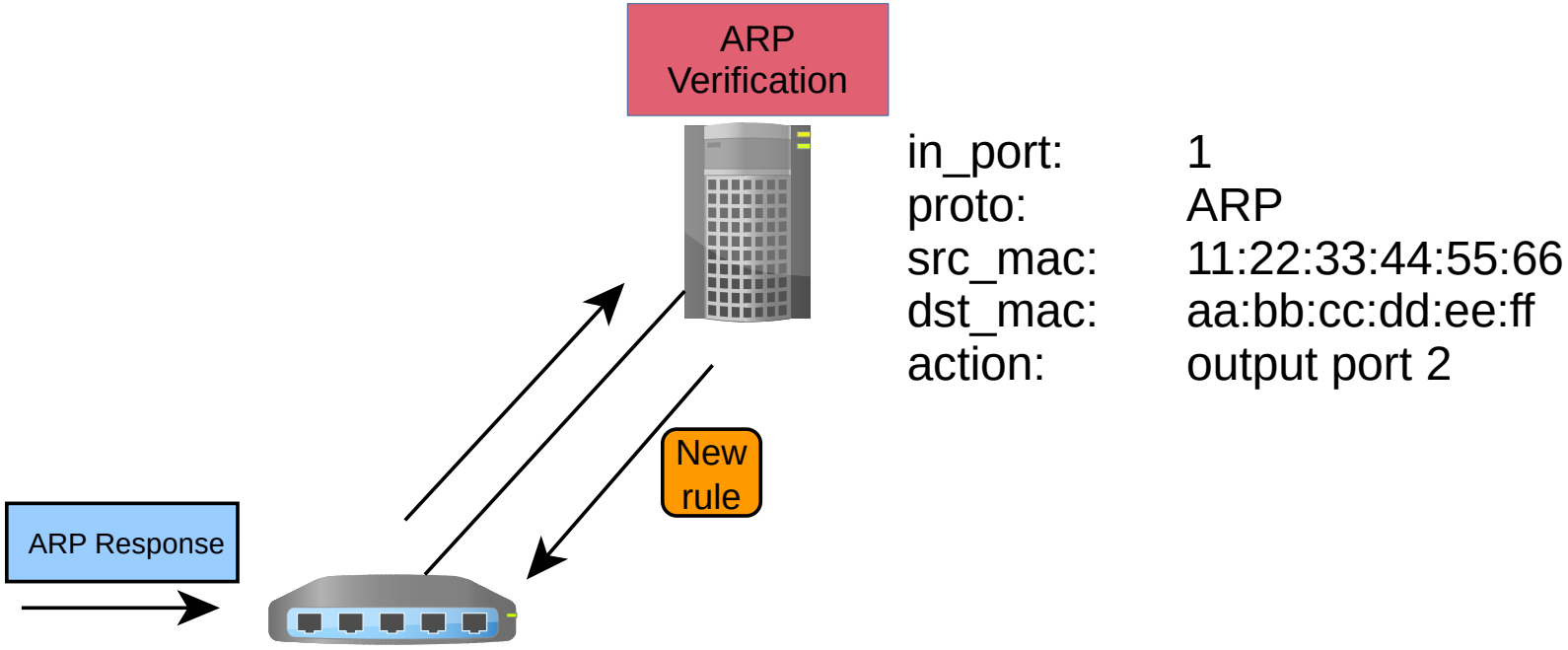
- The forwarding rule installed by the controller might look like this



in_port:	1
proto:	ARP
src_mac:	11:22:33:44:55:66
dst_mac:	aa:bb:cc:dd:ee:ff
action:	output port 2

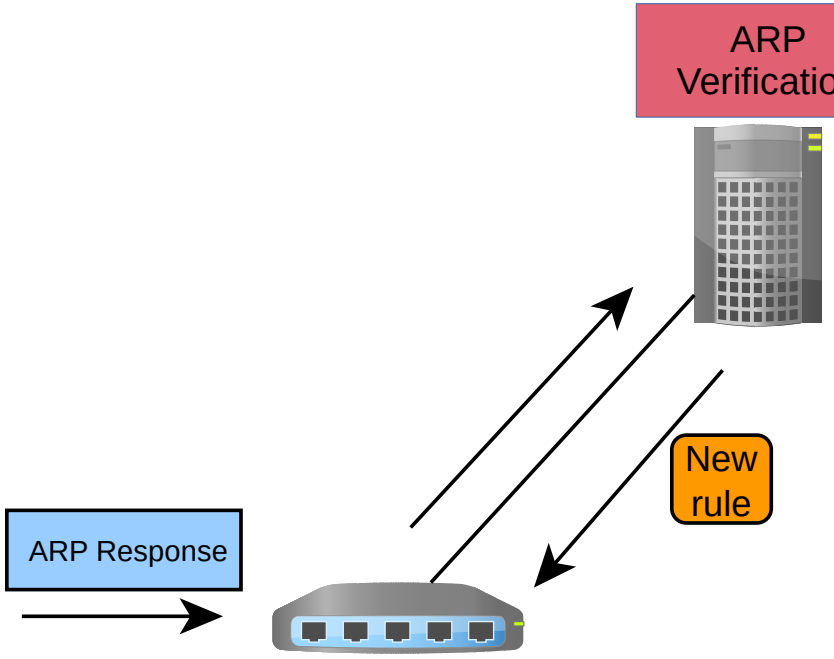
Non-obvious Security Issues

- This rule will match the ARP response and have the switch forward the message out port 2



Non-obvious Security Issues

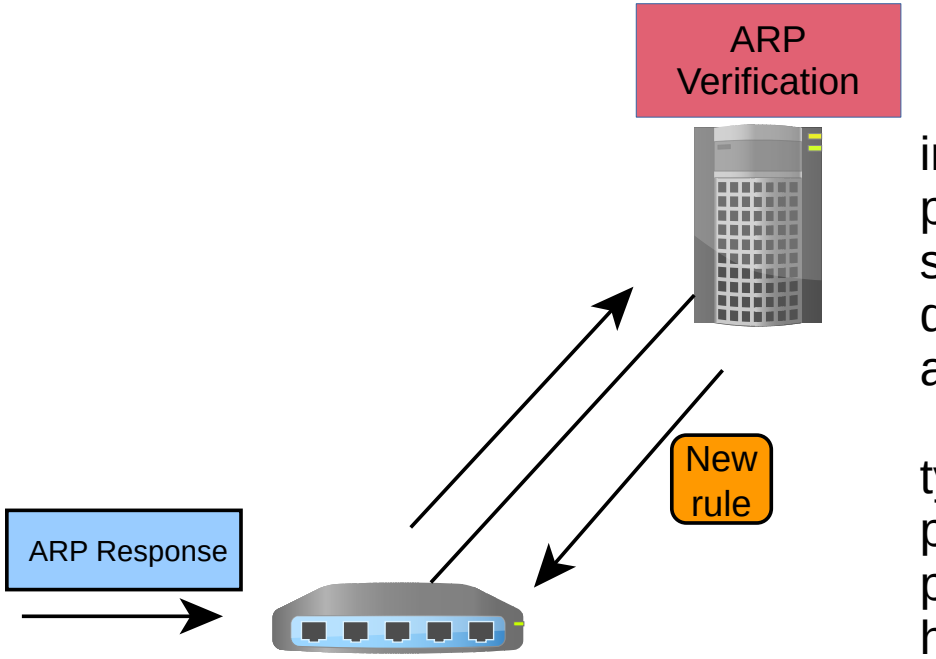
- The issue with this is that it doesn't match on anything in the ARP header



in_port:	1
proto:	ARP
src_mac:	11:22:33:44:55:66
dst_mac:	aa:bb:cc:dd:ee:ff
action:	output port 2
type:	is-at
psrc:	10.0.0.1
pdst:	10.0.0.2
hwsrc:	11:22:33:44:55:66
hwdst:	aa:bb:cc:dd:ee:ff

Non-obvious Security Issues

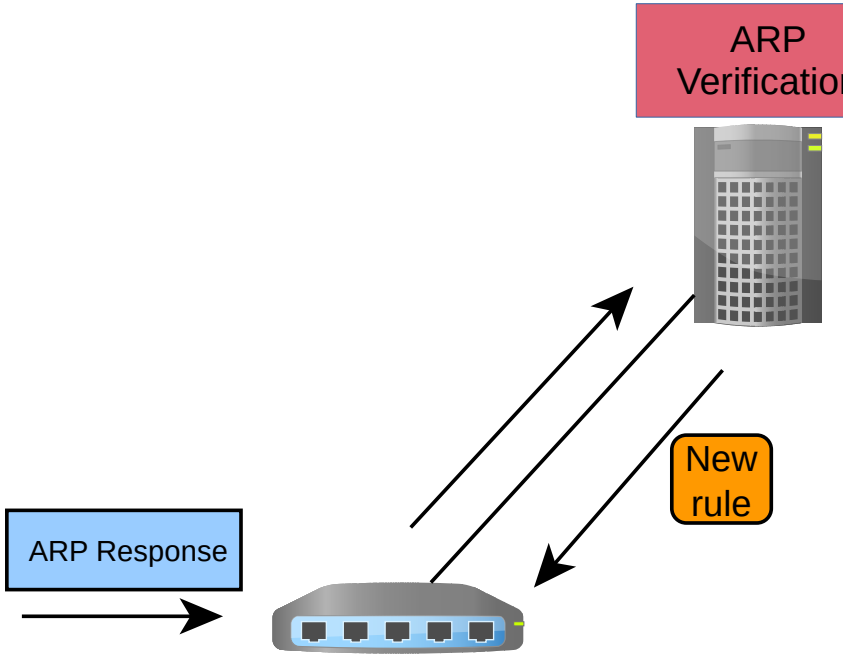
- The issue with this is that it doesn't match on anything in the ARP header



in_port:	1
proto:	ARP
src_mac:	11:22:33:44:55:66
dst_mac:	aa:bb:cc:dd:ee:ff
action:	output port 2
type:	is-at
psrc:	10.0.0.254
pdst:	10.0.0.2
hwsrc:	11:22:33:44:55:66
hwdst:	aa:bb:cc:dd:ee:ff

Non-obvious Security Issues

- The modified ARP response can be piggyback on the existing forwarding rule so the controller will never observe it



in_port:	1
proto:	ARP
src_mac:	11:22:33:44:55:66
dst_mac:	aa:bb:cc:dd:ee:ff
action:	output port 2
type:	is-at
psrc:	10.0.0.254
pdst:	10.0.0.2
hwsrc:	11:22:33:44:55:66
hwdst:	aa:bb:cc:dd:ee:ff

Non-obvious Security Issues

- This technique is called Data-plane ARP Cache Poisoning (DPACP)

- This attack can be extended to allow attackers to bypass access control and firewalling [3]

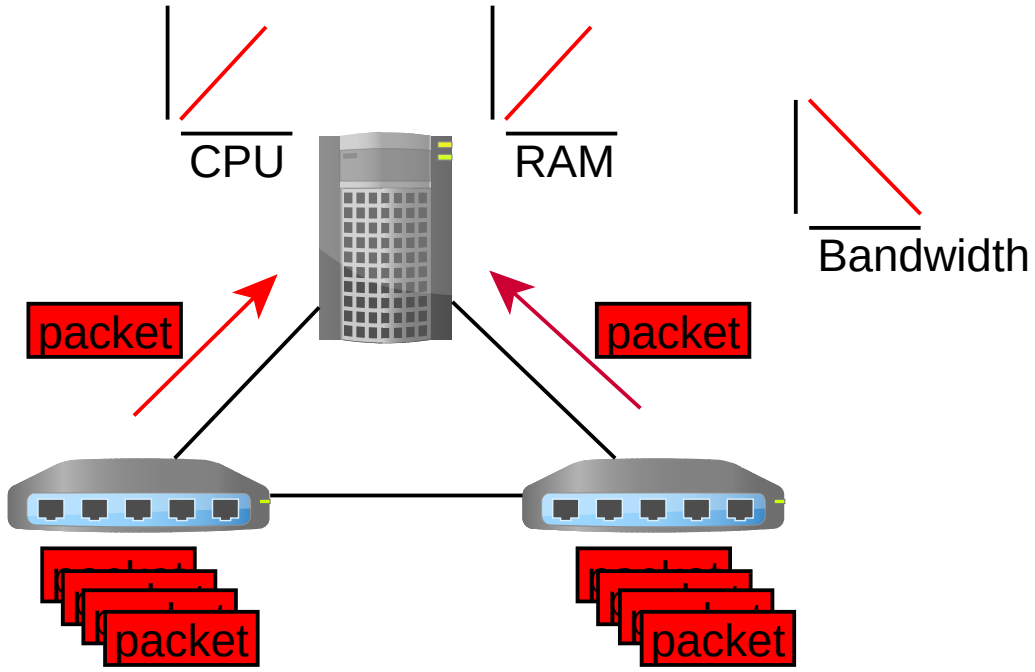
		Layer 3 Access Control Bypass		Layer 2 Access Control Bypass	
Controller (2018 version)	DPACP	Ingress Rule	Egress Rule	Ingress Rule	Egress Rule
Floodlight	Yes	Yes	Yes	No	Yes
ONOS	Yes*	No	No	No	Yes*
ODL	Yes	Yes	Yes	No	Yes
Ryu	Yes	Yes	Yes	No	Yes
Pox	Yes*	No	No	No	Yes*

Non-obvious Security Issues

- Controllers are vulnerable to Denial of Service (DoS)

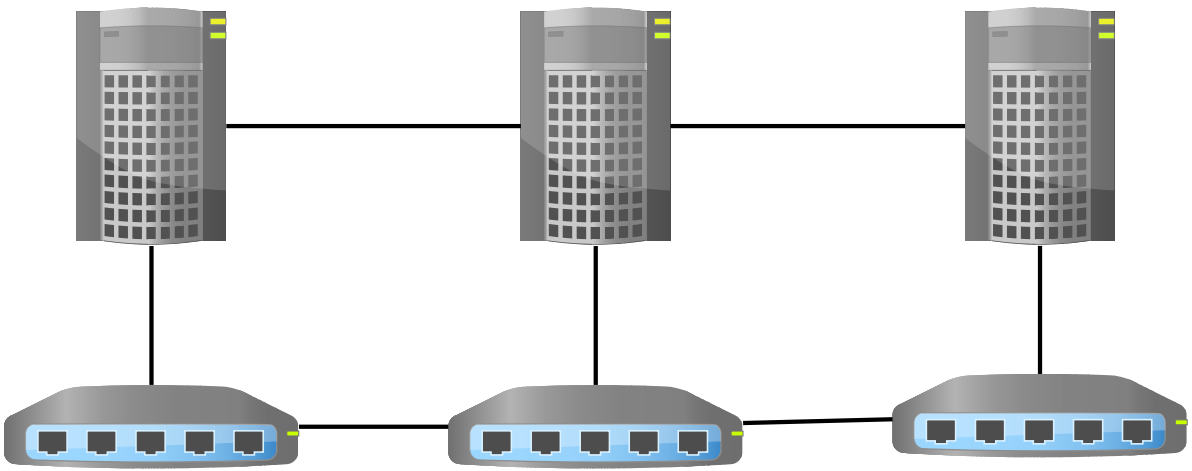
Non-obvious Security Issues

- A surge of packets coming from the switches can overwhelm a controller reducing it's ability to manage the network



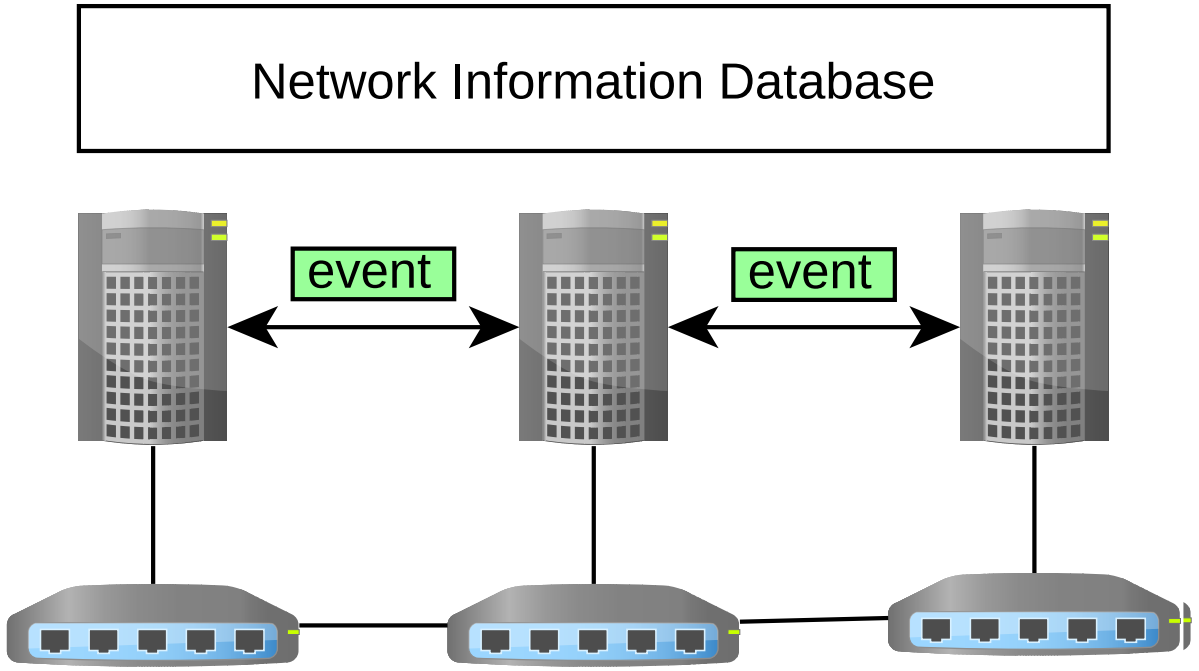
Non-obvious Security Issues

- A distributed controller architecture removes a single point of failure and can help manage such DoS attacks



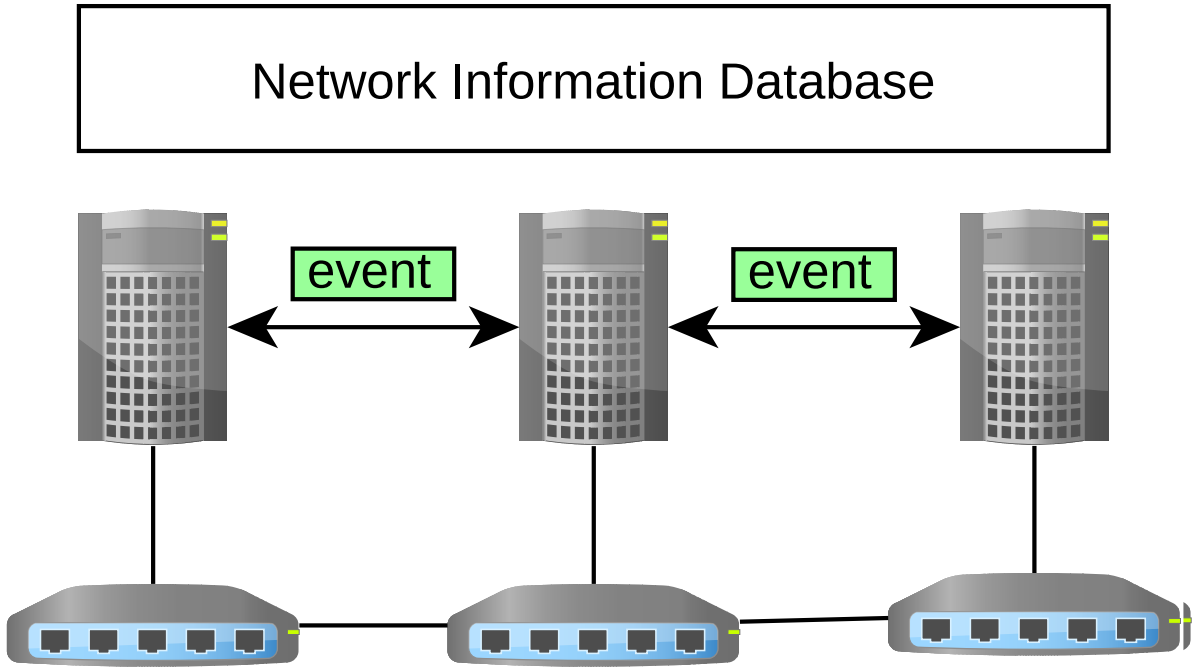
Non-obvious Security Issues

- The controllers maintain a consistent view of the network by sharing **network events** with each other



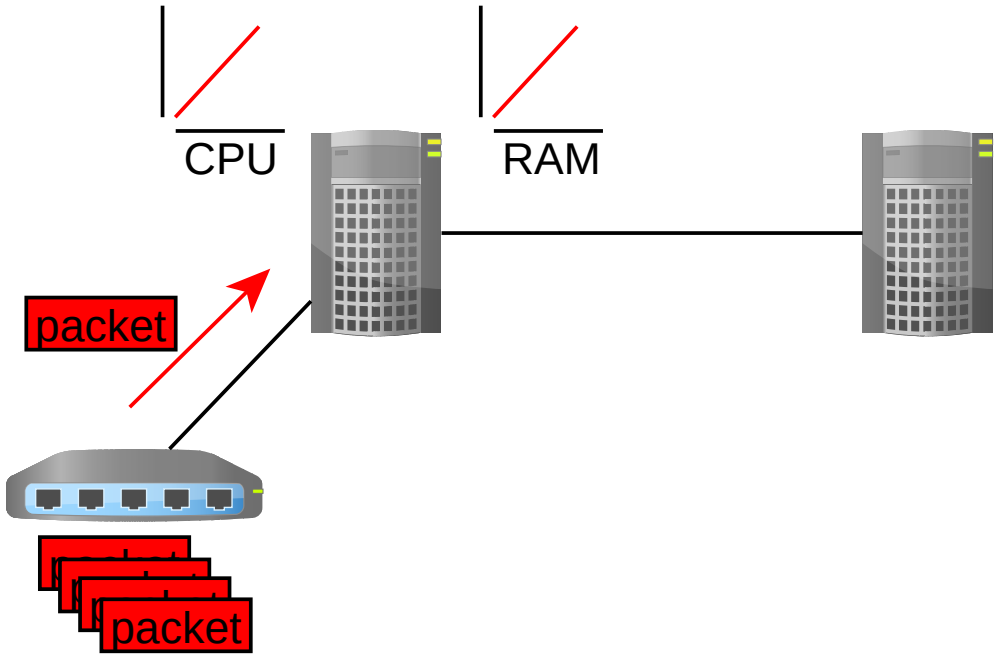
Non-obvious Security Issues

- A new host or link appearing in the network would be an **event**



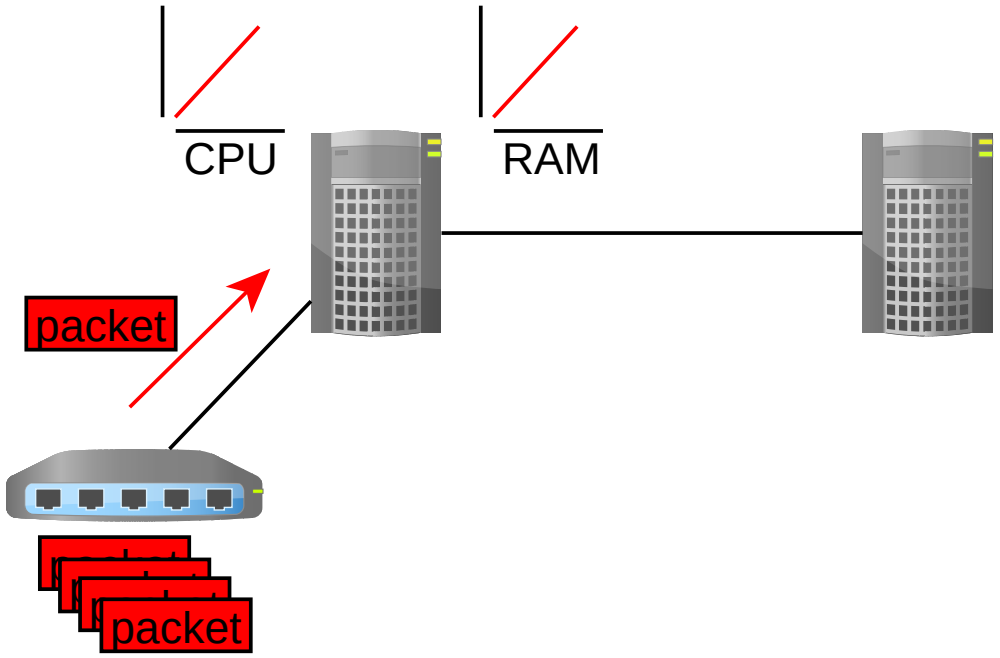
Non-obvious Security Issues

- An attacker performing a DoS attack through flooding would have the same outcome as before



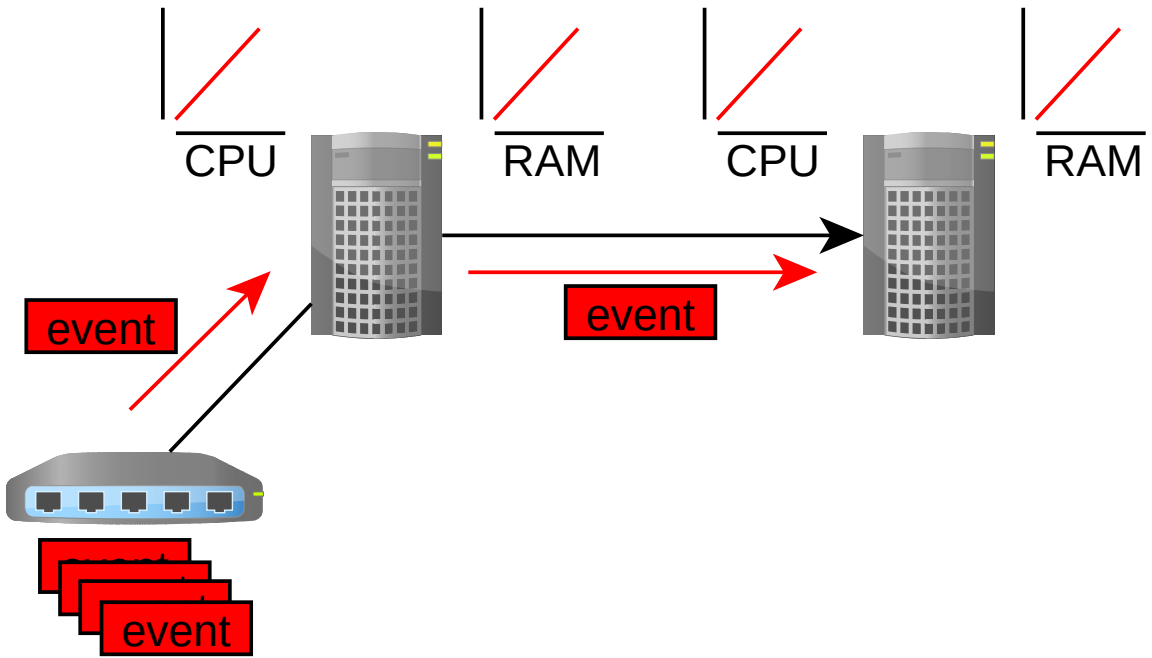
Non-obvious Security Issues

- However, if each new packet is a network event...



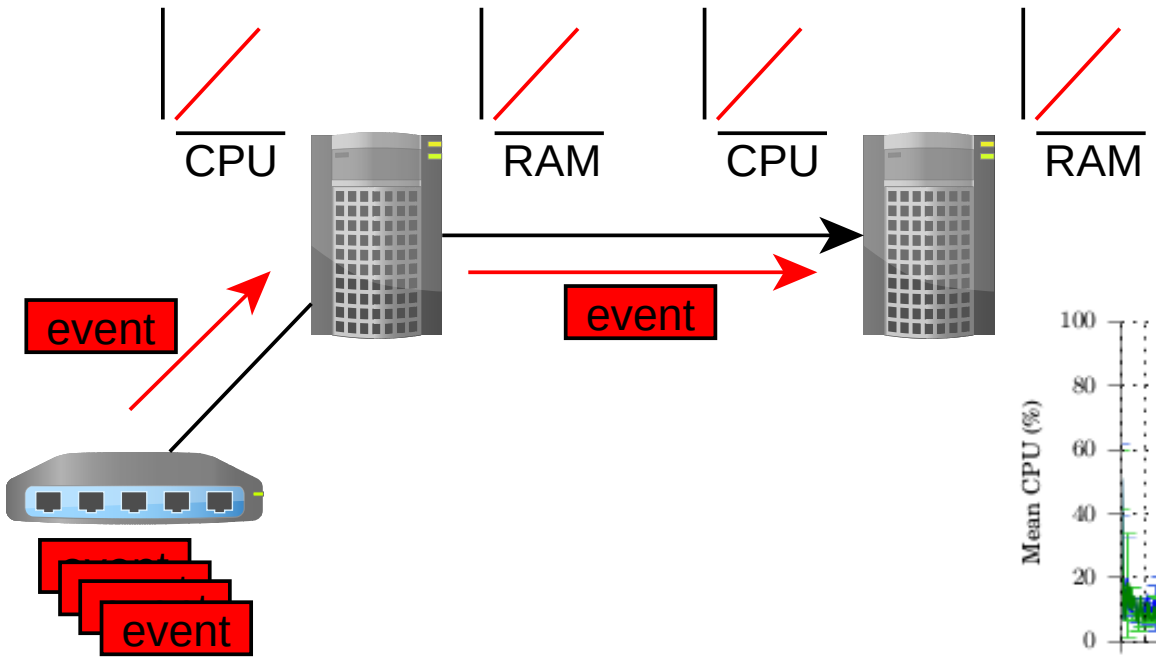
Non-obvious Security Issues

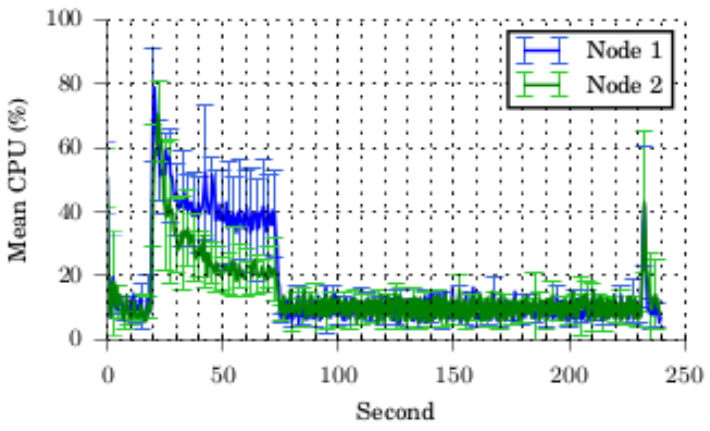
- Every event generated by the attacker will be shared between controllers



Non-obvious Security Issues

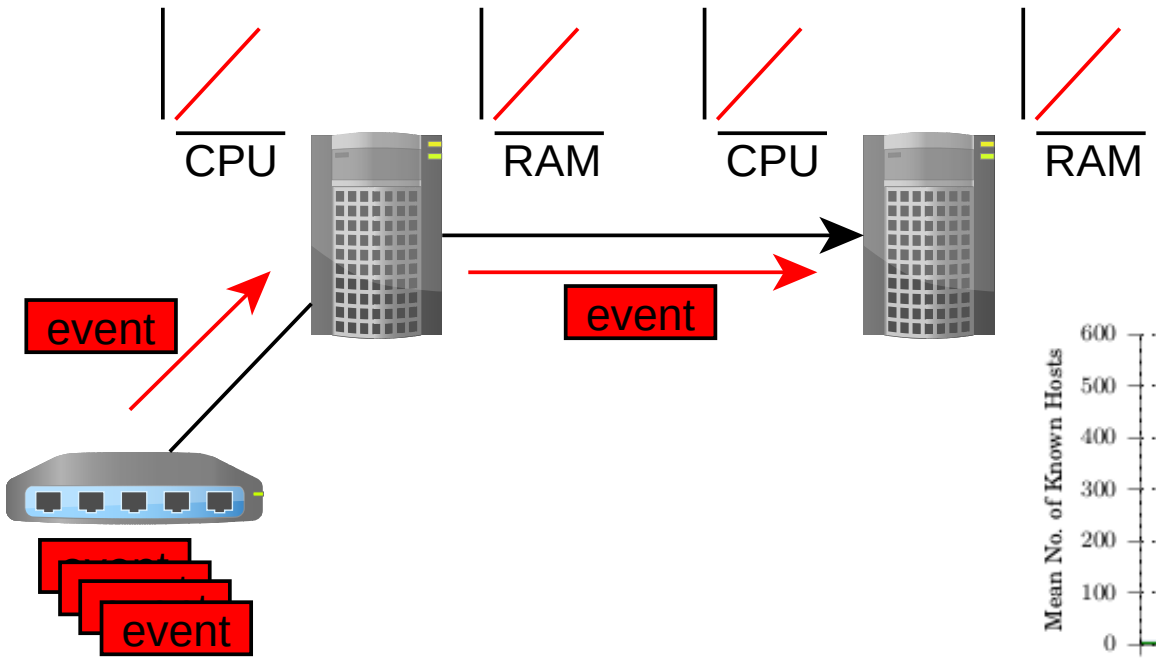
- Previous work has looked at the impact of this [4]

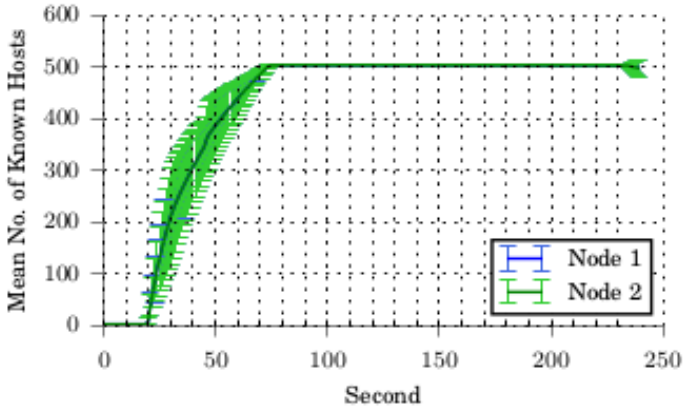




Non-obvious Security Issues

- Previous work has looked at the impact of this [4]





Conclusion

- SDN can be beneficial to network security
- Inexact forwarding rules can be exploited
  - When deploying an SDN it’s important to consider how forwarding rules are structured
- Distributed controller architectures are vulnerable to DoS
  - Rate limiting messages between controllers is needed

References

[1] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S. and Turner, J., 2008. OpenFlow: enabling innovation in campus networks. ACM SIGCOMM Computer Communication Review, 38(2), pp.69-74.

[2] Sinha, Y. and Haribabu, K., 2017. A survey: Hybrid sdn. Journal of Network and Computer Applications, 100, pp.35-55.

[3] Smyth, D., Cionca, V., McSweeney, S. and O'Shea, D., 2016, June. Exploiting pitfalls in software-defined networking implementation. In 2016 International Conference On Cyber Security And Protection Of Digital Services (Cyber Security) (pp. 1-8). IEEE.

[4] Smyth, D., O'Shea, D., Cionca, V. and McSweeney, S., 2019. Attacking distributed software-defined networks by leveraging network state consistency. Computer Networks, 156, pp.9-19.

Thank you!

Questions?