I·C·S
The Institute for Cyber Security

C·SPECC
Center for Security and Privacy Enhanced Cloud Computing

Software Defined Networks: Overview

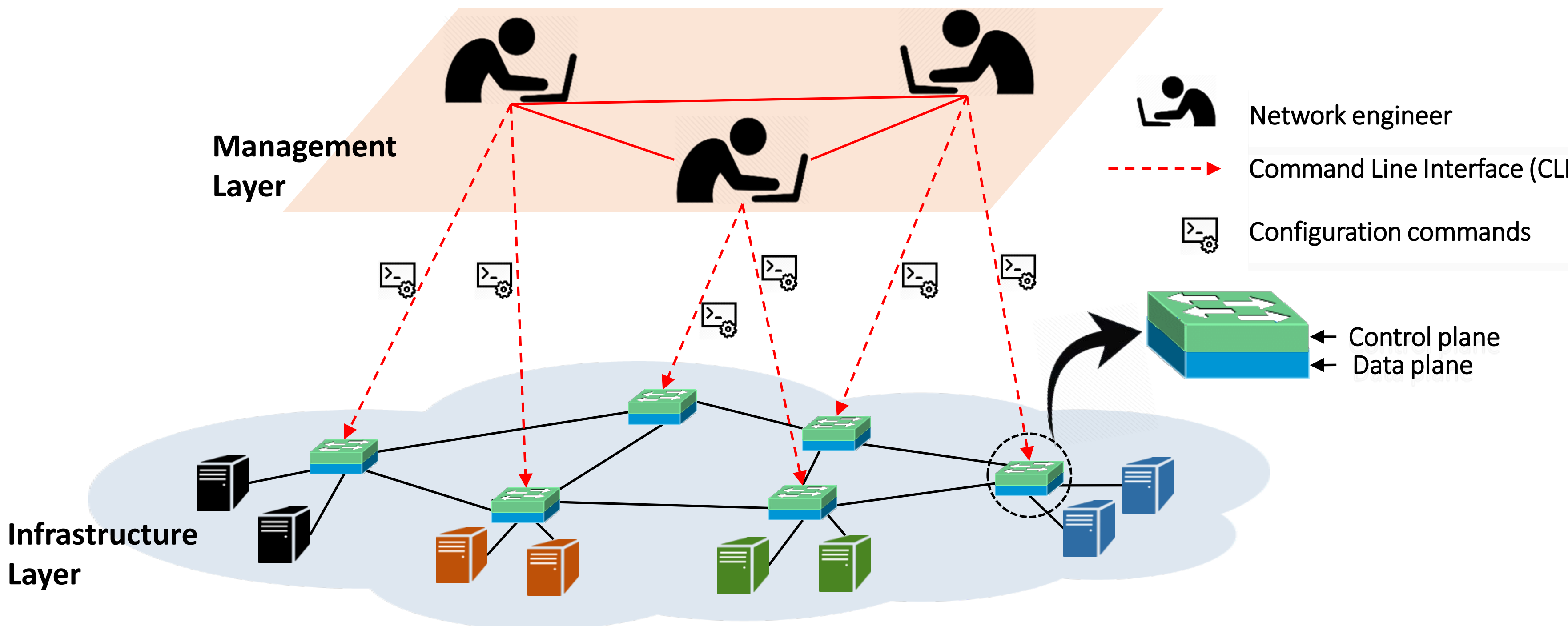
CS6393

Lecture 9-1

UTSA Computer Science

Traditional Networks

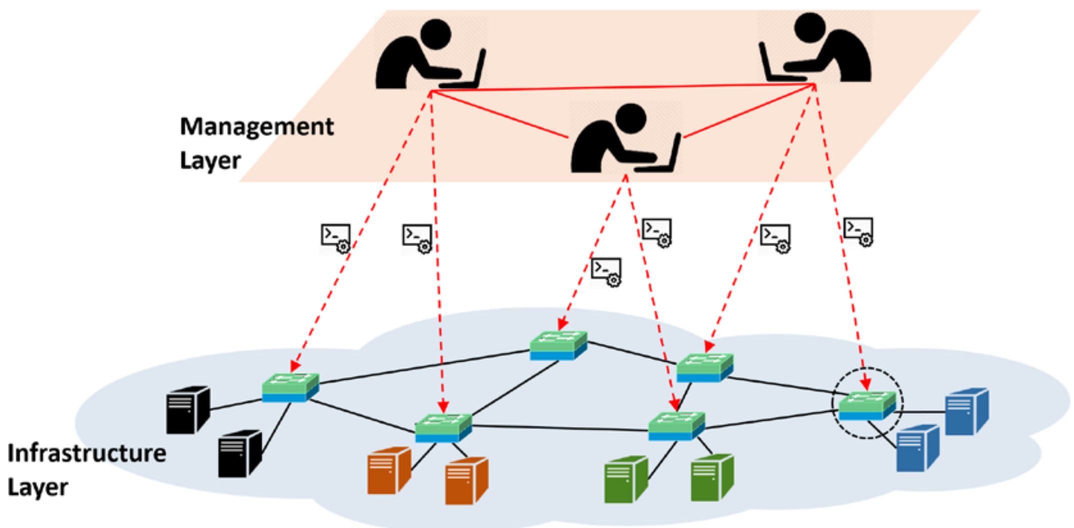
Overview of Traditional Networks Architecture



Traditional Network

Network Devices in Traditional Networks

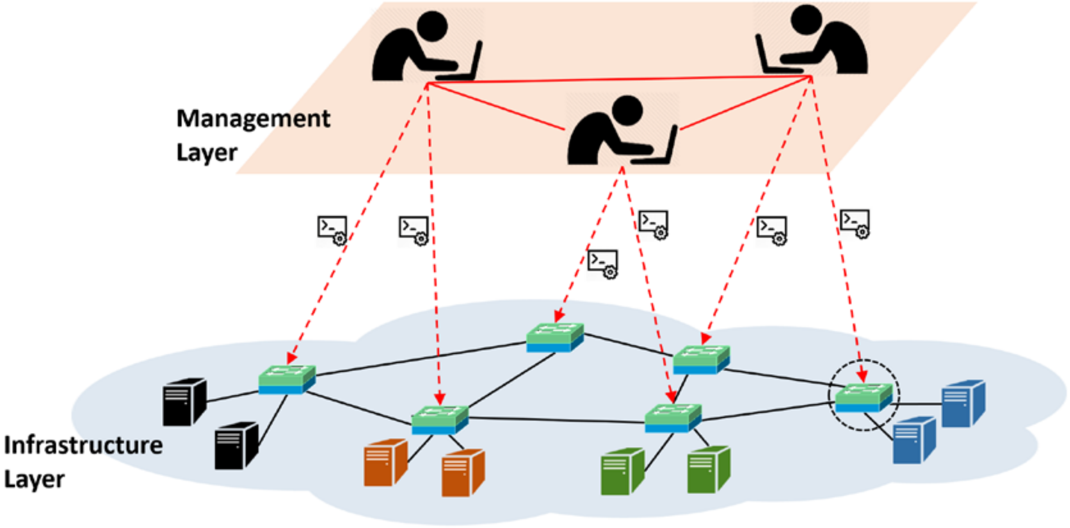
- Vendor-specific and heterogeneous.
- Requires manually configuration.
  - Costly.
  - High-rate of configuration errors.
  - Serious security breaches even for well-known security guidelines.
- Hardwired with specific algorithms and protocols to route, control and monitor data flow.



Traditional Network

Problems in Traditional Networks

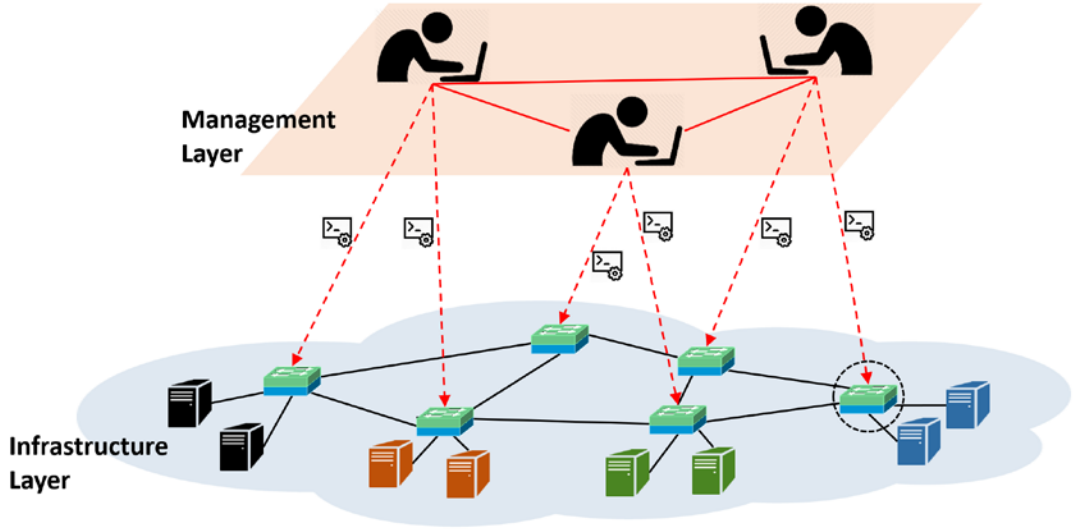
- Lack of global visibility of the network state.
- Difficulties in deploying and maintaining coherent network-wide policies.
- Innovation in networking functionalities and control is difficult.
- Unmanageable and high operational cost by network engineers.
- Complex and weak integration of decentralized networking devices.
- Hard to maintain stable and robust network security.



Traditional Network

Security Policies in Traditional Networks

- Traditional networks lack security automation and run-time deployment of security policies.
- Lack for runtime update of security policies in response to traffic behavior or intrusions.
- To implement security policies, network operators need to
  - Translate high-level security policies into low-level configuration commands.
  - Implement low-level commands manually into large sets of vendor-specific devices.
- Updating security policies might require changing the hardware or updating its firmware.

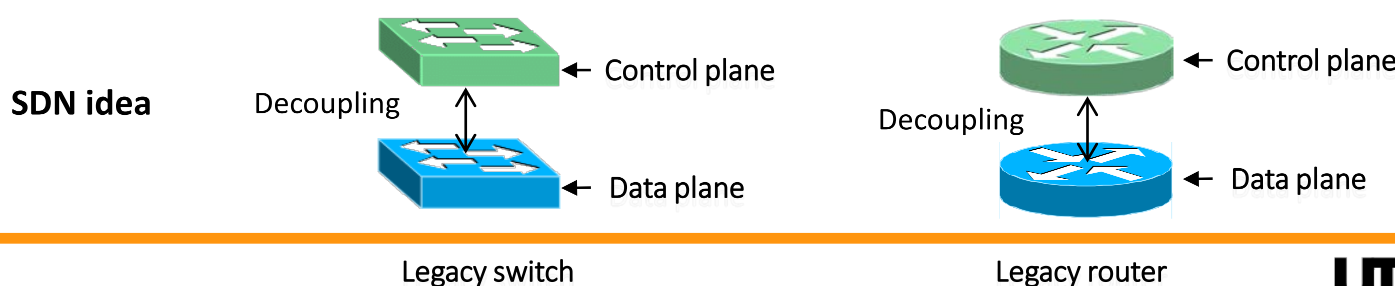


Traditional Network

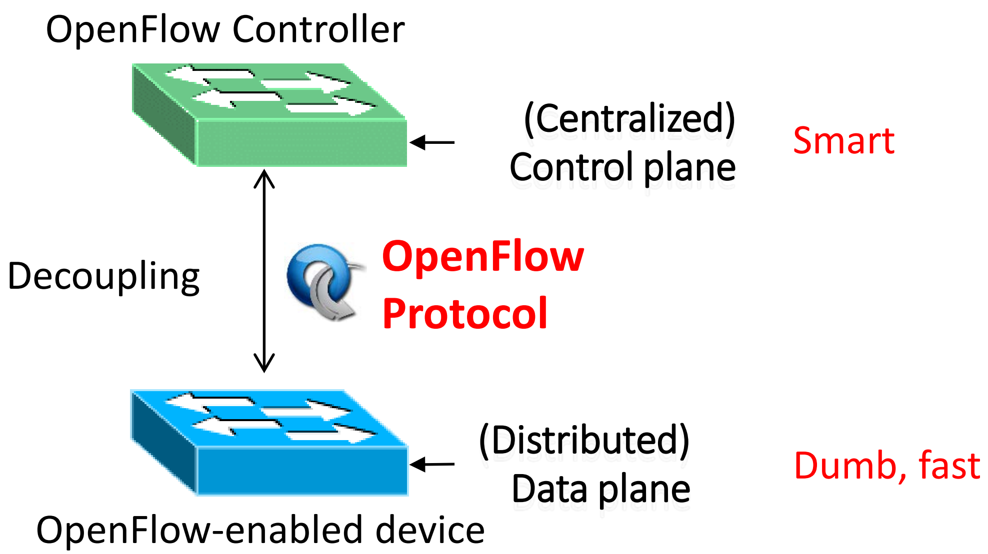
Software Defined Networks

Software Defined Networks (SDN)

- SDN decouples the network control from the data forwarding plane in routers and switches.
- The result:
  - Control plane (logically centralized).
  - Forwarding plane takes decisions from the control plane.
  - Applications and services are implemented on top of the control Plane.
    - Control plan maps the entire network to the application layer.
  - Programmability of network functions and control.



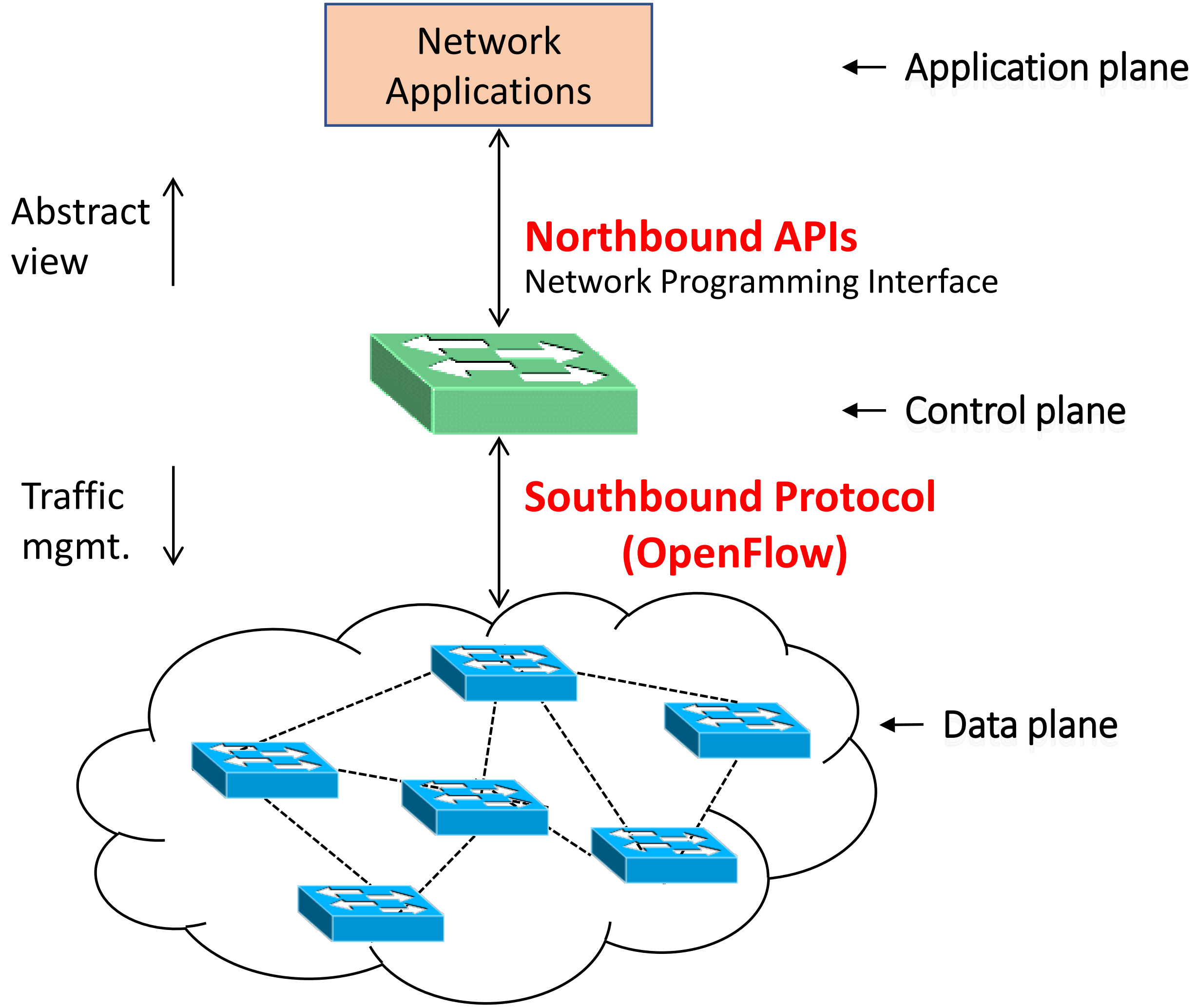
Decoupling : The Idea behind SDNs



Software Defined Networks

- ***OpenFlow***
  - A Protocol between the control plan and data plane.
  - Describes how controller and a network forwarding device should communicate.

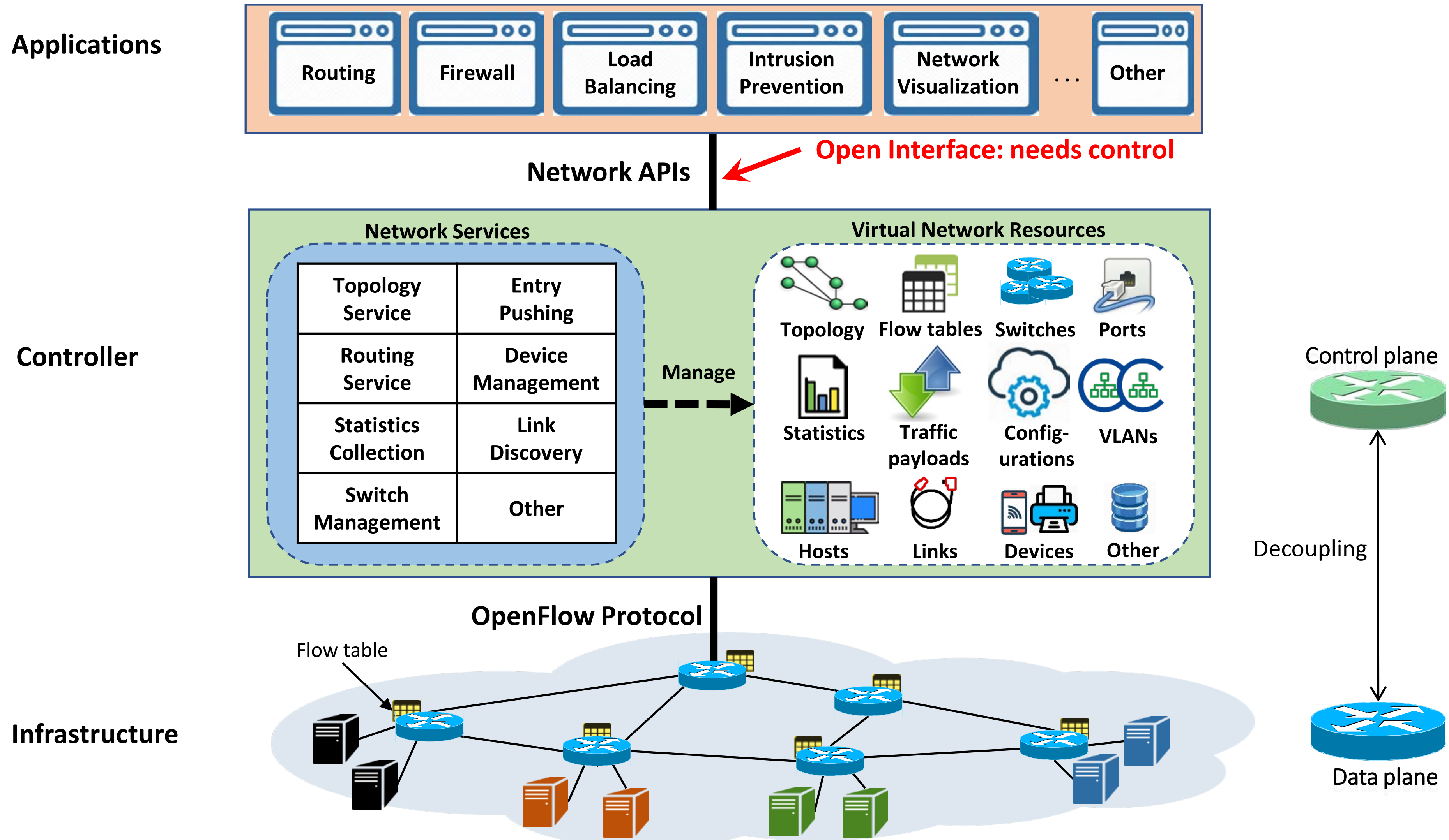
Network Programmability



The three Planes of SDN Architecture

- ***Application Plane:*** *SDN applications for various* functionalities, such as
  - network management
  - traffic automation
  - network Monitoring
  - security services.
  - etc.
- ***Control Plane:*** *Logically centralized control framework that:*
  - runs the Network Operating System (NOS)
  - maintains global view of the network.
  - provides hardware abstractions to SDN applications.
- ***Data Plane:*** *Forwarding elements* that:
  - forward traffic flows based on instructions from the control plane.

Overview of SDN Architecture



Enhanced Network Security in SDN

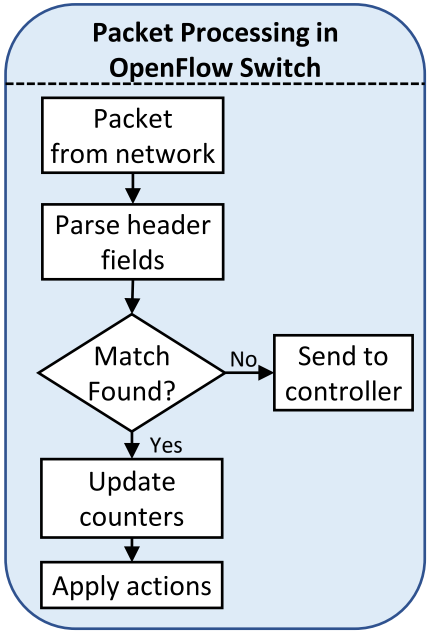
- Network security is enhanced in SDN via:
  - Network programmability.
  - Centralized control of network behavior.
  - Global visibility of the network state.
    - Easier to spot network vulnerabilities and intrusions.
    - Easier to implement security policies.
    - Easier to mitigate the risks of policy collision.
  - Run-time implementation of security policies.
  - Run-time manipulation of traffic forwarding rules.
  - Software based implementation of security policy.
    - No hardware change or firmware update needed.

13

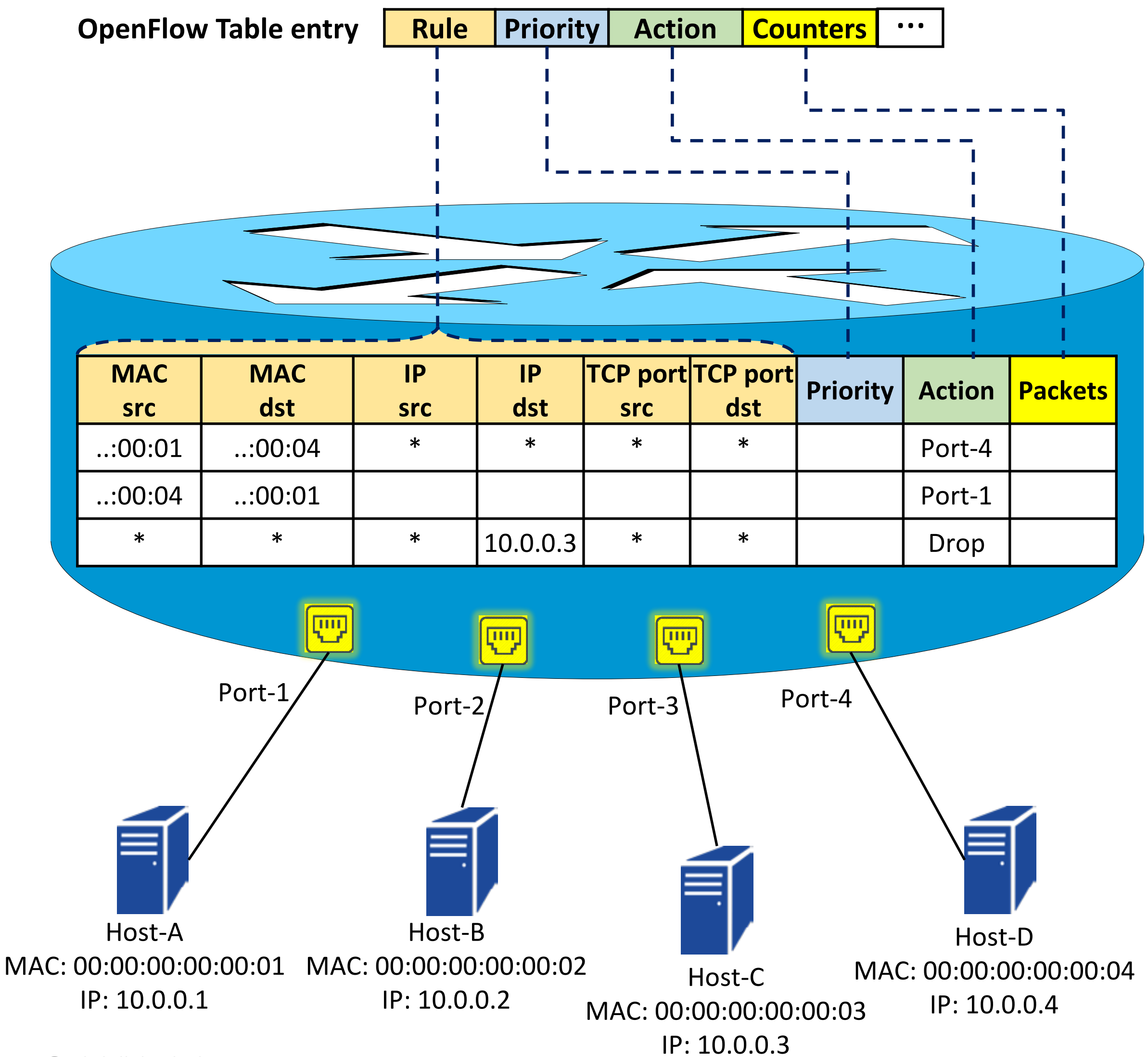
Flow Table Structure

Match Fields	actions	priority	Counters	Timeouts	...

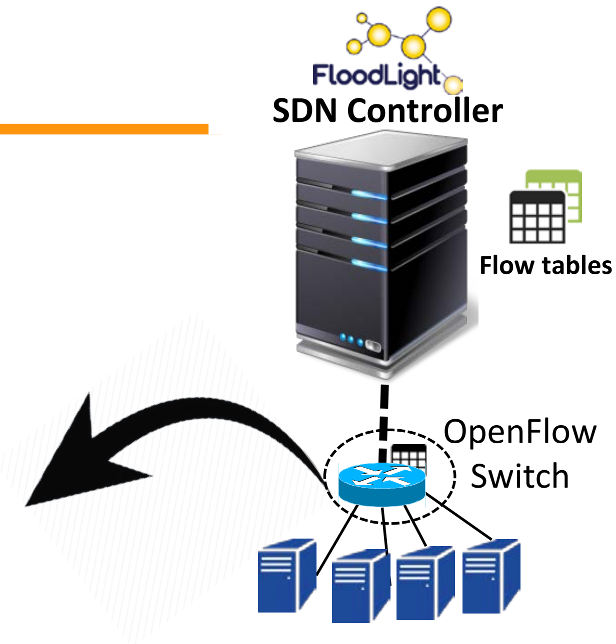
- **match fields:** to match against packets.
- **actions**: to be applied for the matching packet/flow.
- **priority**: precedence of the flow entry.
- **counters**: updated when packets are matched.
- **timeouts**: maximum time/idle time before flow is expired by the switch.

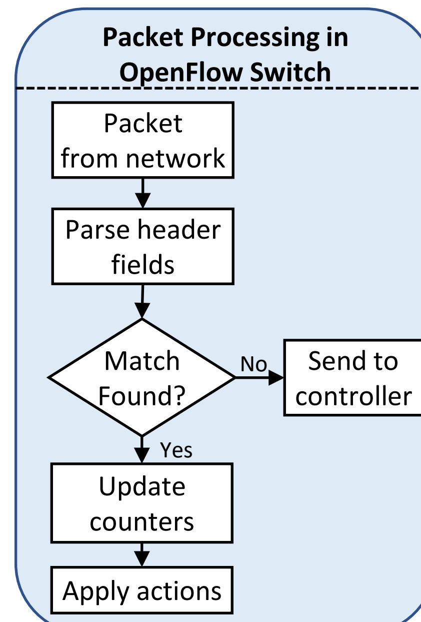


Flow Table Structure
example flow rules

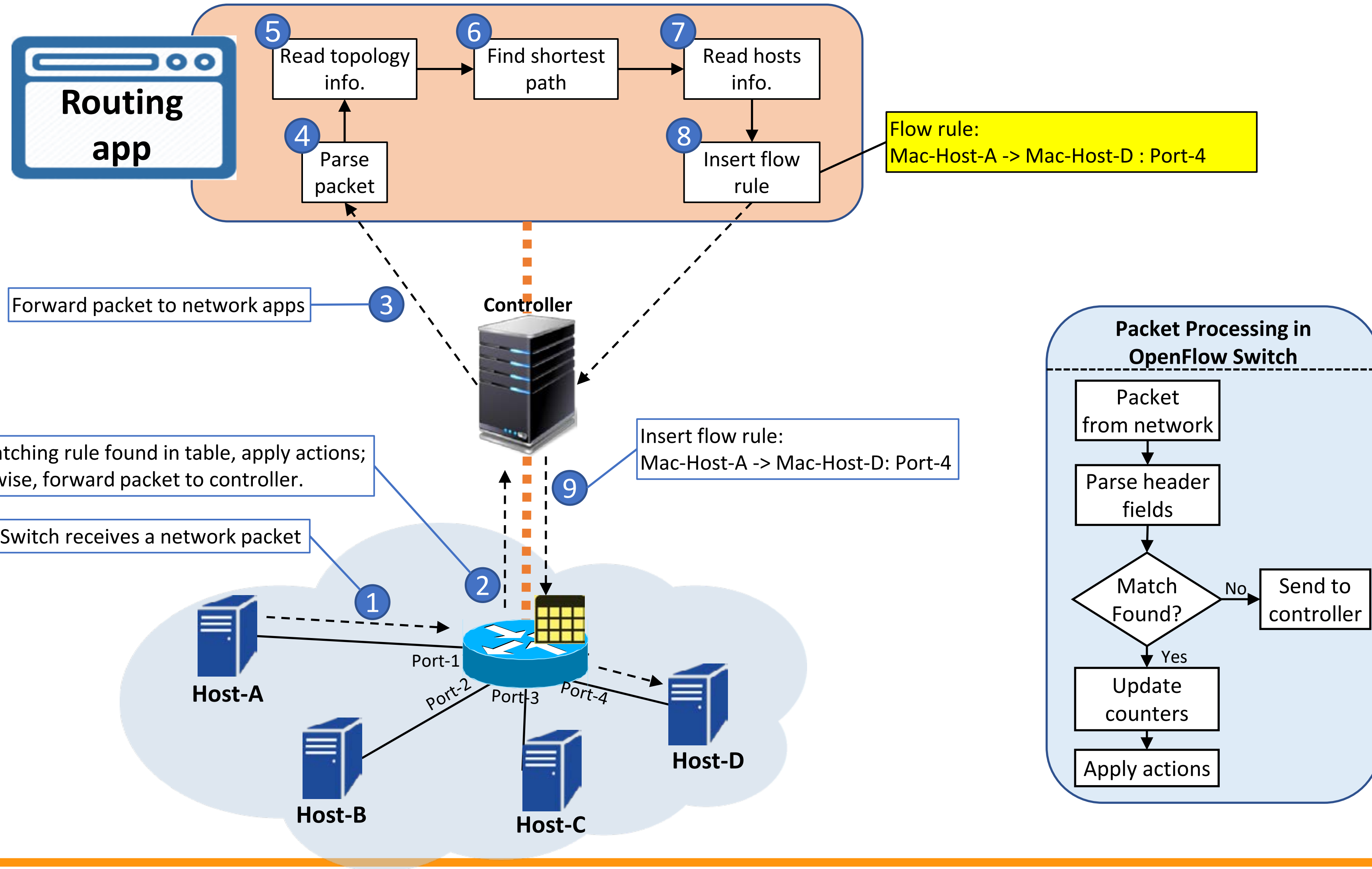


MAC src	MAC dst	IP src	IP dst	TCP port src	TCP port dst	Priority	Action	Packets
..:00:01	..:00:04	*	*	*	*		Port-4	
..:00:04	..:00:01						Port-1	
*	*	*	10.0.0.3	*	*		Drop	

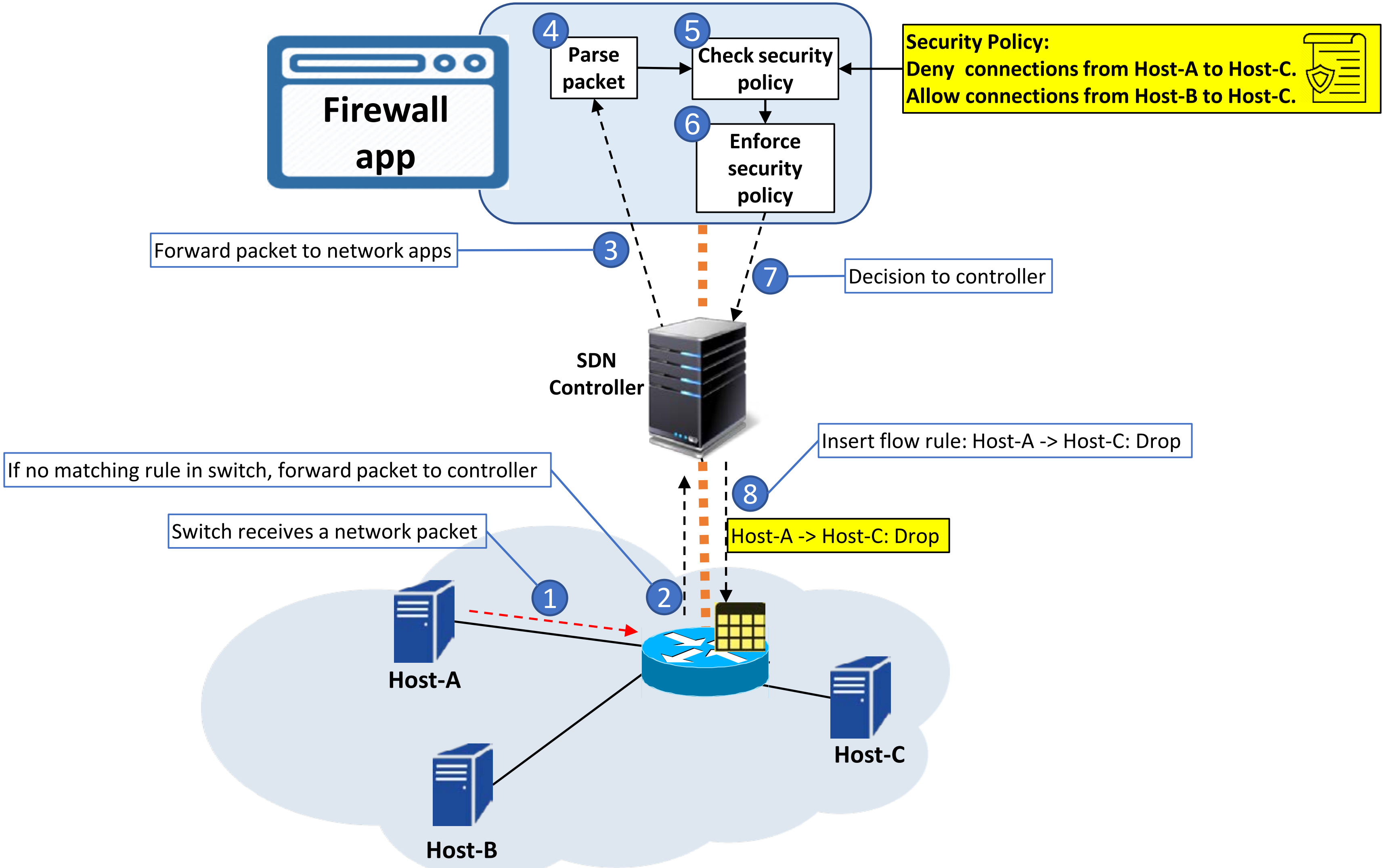




Flow Rule Insertion Example - Routing app



Flow Rule Insertion Example - Firewall app



SDN Security Challenges

Apparent Control Plane Security Challenges

Threats due to Scalability

-Huge # flow rules

DoS Attacks

-SDN response times
-IP packets with random headers

Challenges in Distributed Control Plane

- Information aggregation from multiple domains.
- different privacy rules in different domains.

Ahmad, Ijaz, Suneth Namal, Mika Ylianttila, and Andrei Gurtov. "Security in software defined networks: A survey." *IEEE Communications Surveys & Tutorials* 17, no. 4 (2015): 2317-2346.

19

Application Plane Security Challenges

Lack of authentication mechanisms for SDN apps

Requires a trust relationship between the controller and apps

Lack of access control for SDN apps

Requires a permission system to apply least privilege on apps

Fraudulent and contradicting flow rule insertion

Requires a flow rule conflict resolution

Ahmad, Ijaz, Suneth Namal, Mika Ylianttila, and Andrei Gurtov. "Security in software defined networks: A survey." *IEEE Communications Surveys & Tutorials* 17, no. 4 (2015): 2317-2346.

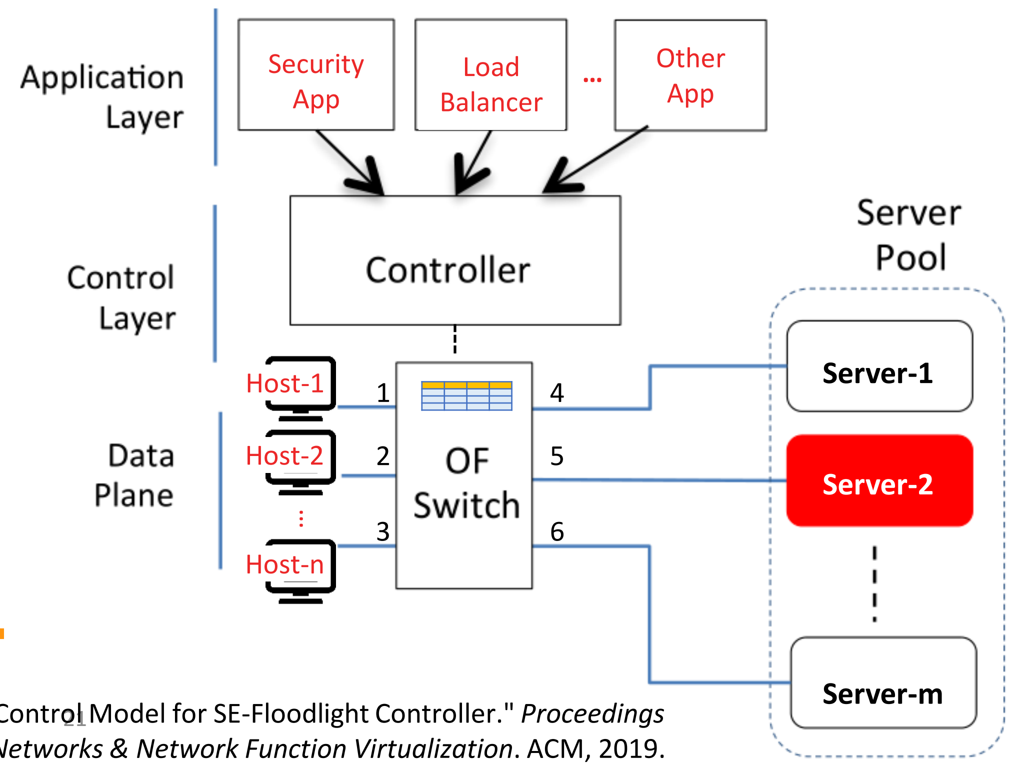
Contradicting flow rule insertion - example

	In port	MAC src	MAC dst	VLAN ID	IP src	IP dst	TCP psrc	TCP pdst	Priority	Action
Rule1	*	*	MAC Server-2	10	*	*	*	*	150	drop
Rule2	*	MAC Server-2	*	10	*	*	*	*	150	drop
Rule3	1	MAC Host-1	MAC Server-2	10	*	*	*	*	200	out port =5
Rule4	5	MAC Server-2	MAC Host-1	10	*	*	*	*	200	out port =1
	...	...	...	...	...	...	...	...	...	...

Assume this sequence of activities by SDN Apps.

1. **Security App** has identified Server-2 as Malicious server.
2. **Security App** Inserts **rule1** & **rule2** to quarantine the flows to/from Server-2.
3. Server-2 becomes least loaded server in the pool.
4. **Load Balancer App** Inserts flow Rules **rule3** & **rule4** to redirect traffic to Server-2.

Load Balancer App violates the security policy implemented by Security App.



Al-Alaj, Abdullah, Ravi Sandhu, and Ram Krishnan. "A Formal Access Control Model for SE-Floodlight Controller." *Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*. ACM, 2019.

References

- Ahmad, Ijaz, Suneth Namal, Mika Ylianttila, and Andrei Gurtov. "Security in software defined networks: A survey." *IEEE Communications Surveys & Tutorials* 17, no. 4 (2015): 2317-2346.
- Al-Alaj, Abdullah, Ravi Sandhu, and Ram Krishnan. "A Formal Access Control Model for SE-Floodlight Controller." *Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*. ACM, 2019.