NEW YORK INSTITUTE
OF TECHNOLOGY

INCS 775
Data Center Security

SDN Security

Dr. Zakaria Alomari
zalomari@nyit.edu

Objectives

- ❑ How to Secure SDN?
- ❑ How SDN secures that Data Center?

Today’s Agenda

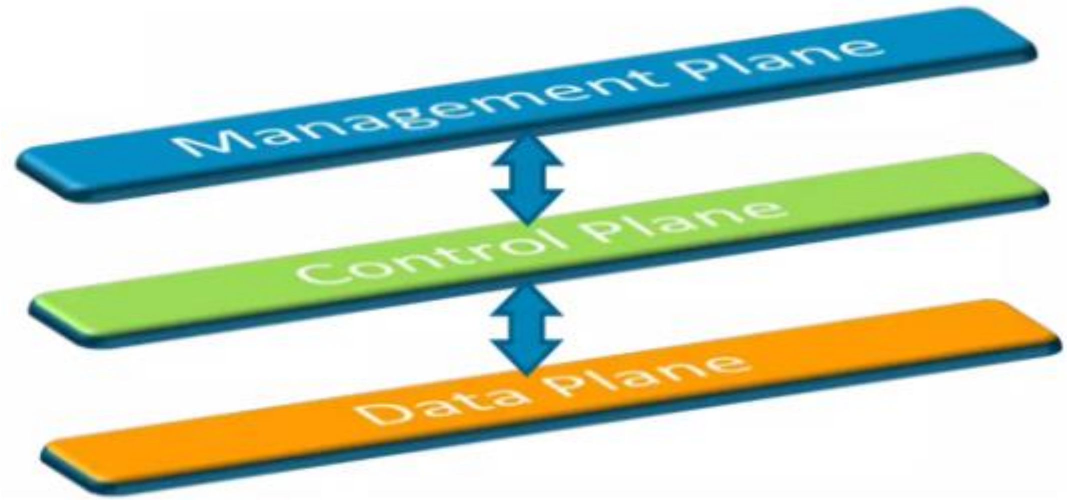
- Brief Review of Software Defined Networking (SDN)

- Heads:
  - Attack Vectors for SDN Systems
  - Securing an SDN System

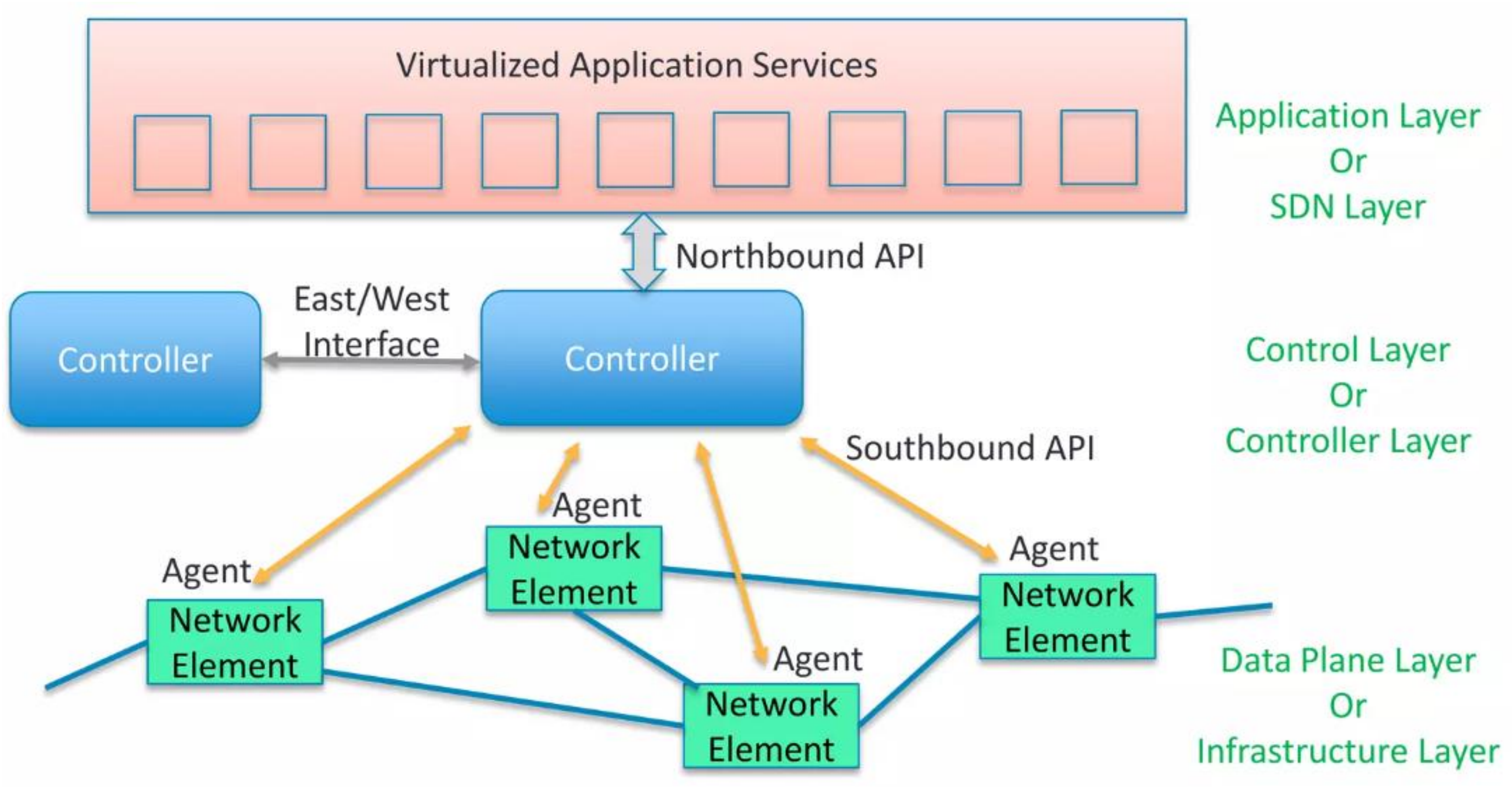
- Tails:
  - SDN Security Use Cases and Applications

Defining SDN

- Software-Defined Networking is an approach to networking that separates the control plane from the forwarding plane to support virtualization.
- SDN is a new paradigm for network virtualization



SDN High-Level Architecture

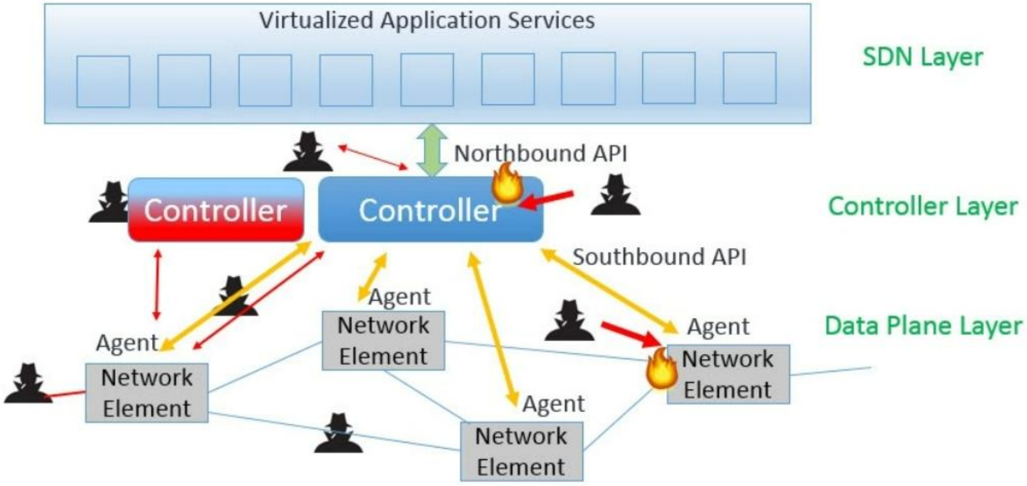


Heads: Security of SDN System

- There are several attack vectors on SDN systems. The more common SDN security concerns include:
  - **Attacks targeting the SDN controller →** either DOS or to instantiate new flows (*spoofing northbound API messages or spoofing southbound flows*)
  - **The attacker establishes its own controller and orchestrates network elements to accept data streams from this controller→** *spoofing flows from the legitimate controller*
  - **Targeting the network elements →** *DoS or to instantiate new flows*
  - **Attacking the DCI protocol (VXLAN, NVGRE, STT) →** *These protocols may lack authentication, with no encryption, this is either part of the protocol design, or the vendor has implemented the protocol.*

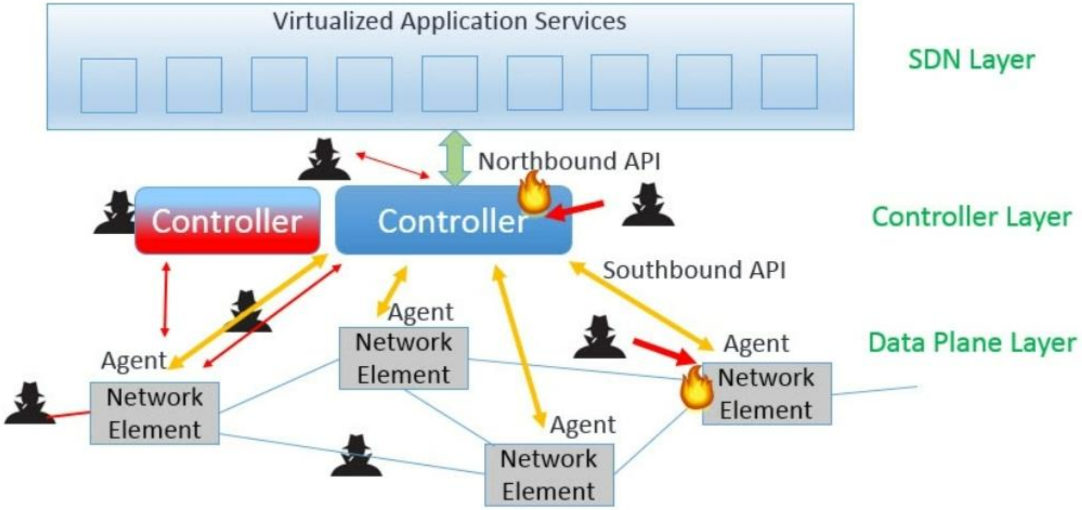
Heads: Security of SDN System

- Firstly, the attacker would want to instantiate new flows by either spoofing northbound API messages or spoofing southbound messages toward the network devices.
- If an attacker can successfully spoof flows from the legitimate controller then the attacker would have the ability to allow traffic to flow across the SDN at **their will** and **possibly bypass policies that may be relied on for security**.



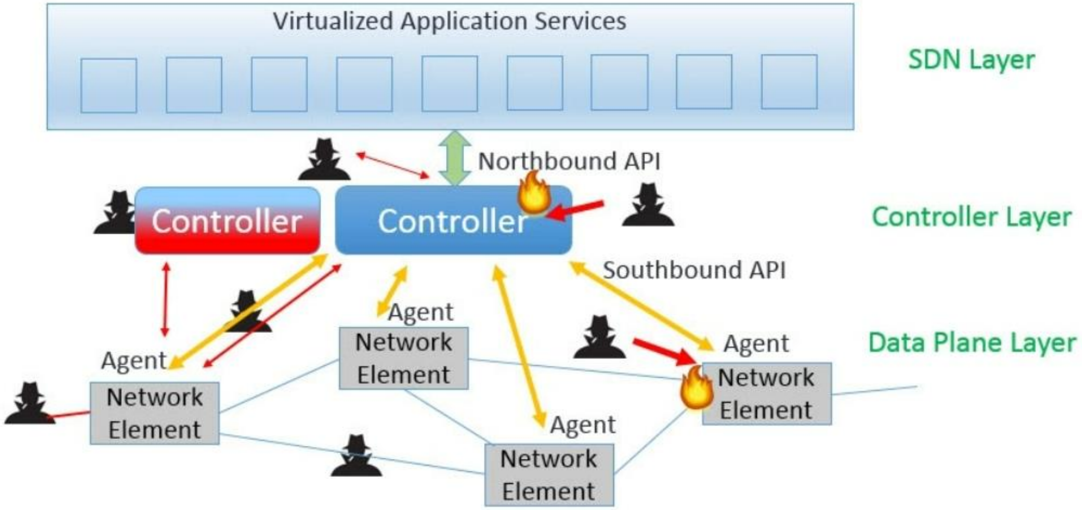
Heads: Security of SDN System

- An attacker might try to perform a DoS of the controller or use another method to cause the controller to fail.
- The attacker might try to attempt some form of resource consumption attack on the controller to bog it down and cause it to respond extremely slowly to Packet_In events and make it slow to send Packet_Out messages.



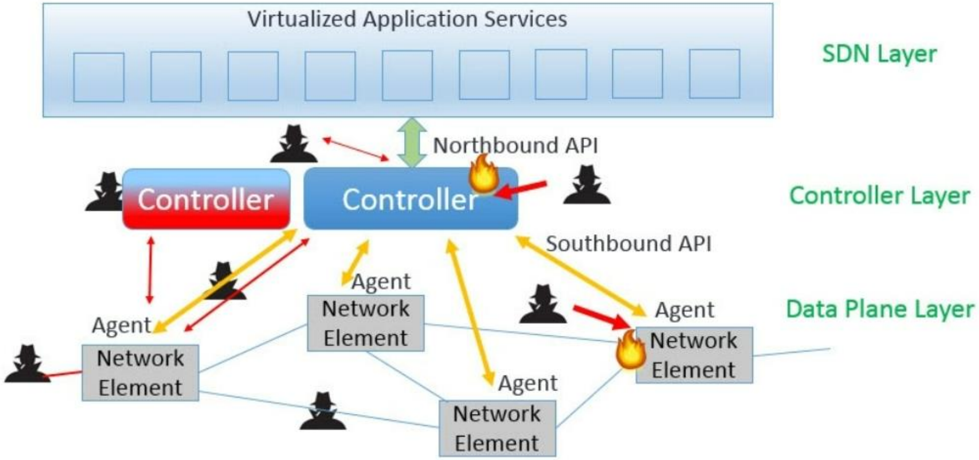
Heads: Security of SDN System

- Secondly, it would be bad if an attacker created its own controller and got network elements to believe flows from the “rogue” controller.
- The attacker could then create entries in the flow tables of the network elements and the SDN engineers would not have visibility to those flows from the perspective of the production controller.
- In this case, the ***attacker would have complete control of the network***.



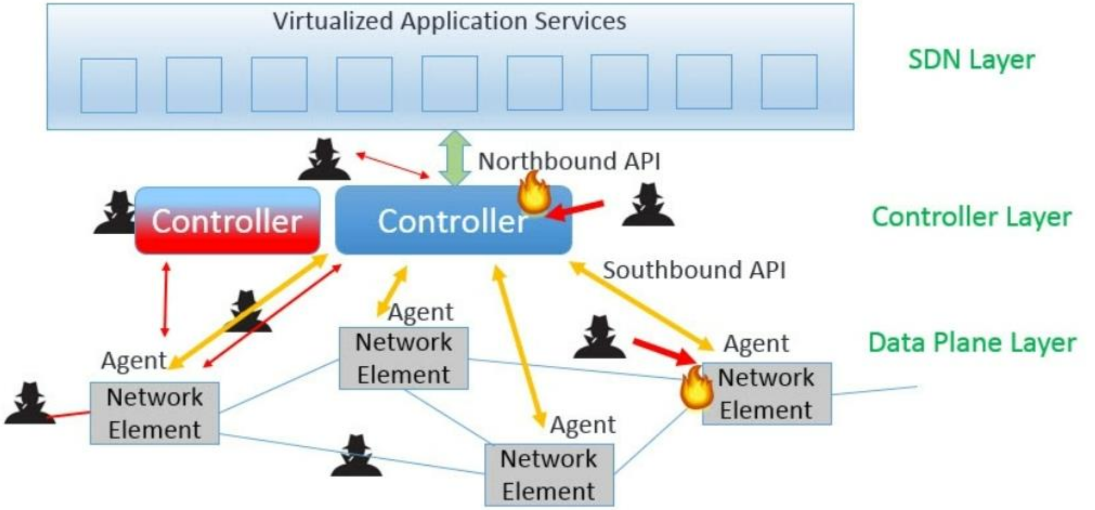
Heads: Security of SDN System

- Thirdly, attackers could target the network elements from within the network itself.
- An **attacker** could theoretically gain unauthorized physical or virtual access to the network or compromise a host that is already connected to the SDN and then try to perform attacks to destabilize the network elements.
- This could be a type of Denial of Service (DoS) attack or it could be a type of fuzzing attack to try to attack the network elements.



Heads: Security of SDN System

- An attacker would like to eavesdrop on flows to see what flows are in use and what traffic is being permitted across the network.
- The attacker would want to try to eavesdrop on southbound communication between the **network element** and the **controller** → This information could be useful for a replay attack or simply for exploration purposes.



SDN Vulnerability Genome Project

- The **SDN Vulnerability Genome Project** is an initiative aimed at identifying, cataloging, and analyzing vulnerabilities within Software-Defined Networking (SDN) environments.

- The **SDN Vulnerability Genome Project** seeks to systematically document these vulnerabilities to enhance understanding and awareness within the cybersecurity community. By creating a comprehensive database of SDN vulnerabilities, researchers and practitioners can better assess risks, develop mitigation strategies, and ultimately strengthen the security posture of SDN deployments.

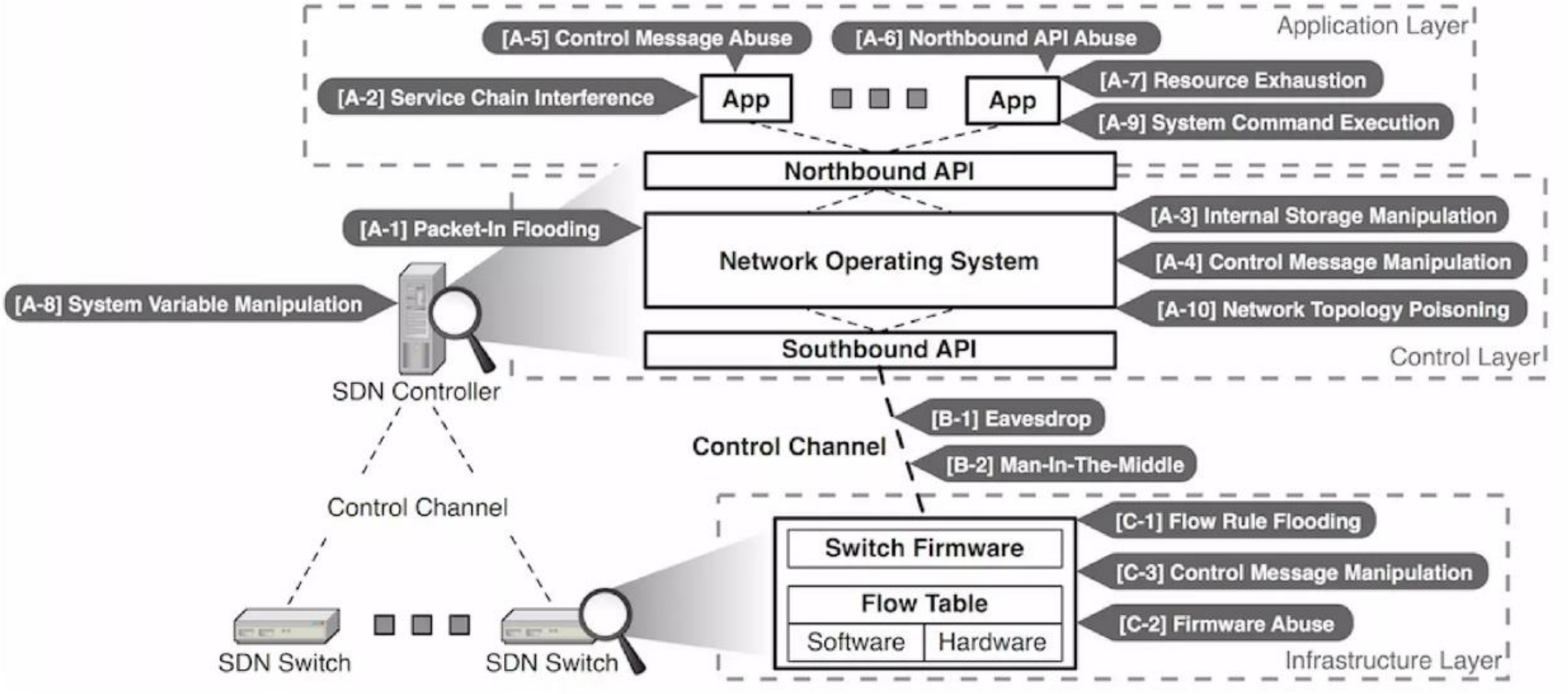
SDN Vulnerability Genome Project

- The project typically involves collecting data on vulnerabilities affecting SDN frameworks, controllers, protocols, and related components.

- This data is then analyzed to identify common patterns, trends, and potential areas of improvement in SDN security.

- Ultimately, the goal is to foster collaboration among stakeholders, promote best practices, and contribute to the ongoing evolution of secure SDN technologies.

SDN Vulnerability Genome Project



Recent SDN System Vulnerabilities

- Some version of SDN systems may contain other opensource software that is discovered to have vulnerabilities: **bash, OpenSSH**, **OpenSSL**, **ntpd.**
  - **OpenSSH** is the open-source version of the Secure Shell (SSH) tools used by administrators of Linux and other non-Windows for cross-platform management of remote systems.

- Several vulnerabilities have been reported and fixed within OpenDaylight
  - https://wiki.opendaylight.org/display/ODL/Security
  - **Mailing Lists and Forums:** Subscribe to OpenDaylight mailing lists and forums. These channels are often used to announce security updates and discuss related issues.

Recent SDN System Vulnerabilities

- The **Netdump vulnerability in OpenDaylight**,
  - Discovered by Gregory Pickett of Hellfire Security, involved a flaw in the protocol used for transporting kernel core memory images during a crash.
  - The issue went unnoticed for four months because there was no direct security contact in OpenDaylight, and the initial report was sent to a generic inbox.
  - It only gained attention in December after being discussed on the project's public mailing list. This delay highlighted the need for better security reporting mechanisms within OpenDaylight.
  - https://seclists.org/bugtraq/2014/Aug/75
  - **Now OpenDaylight project has security team in place**

Recent SDN System Vulnerabilities

- **ONIE** vulnerabilities identified in BigSwitch's Switch Light controller, Cumulus Linux, Mellanox-OS (August 2015)
  - **Open networks install environment (ONIE) is an open source "install environment" and network operating system installer.** It is widely used by white-box switch vendors to load a network OS onto its switches.

- CVE-2015-5699-Cumulus Linux's Switch Configuration Tools Backend, clcmd_server, Vulnerable to Local Privilege Escalation (August 11, 2015)
  - This vulnerability allowed local privilege escalation, meaning a local user could gain higher privileges than intended

Recent SDN System Vulnerabilities

- Augest 3, 2015 – Cisco APIC (Cisco's Application Policy Infrastructure Controller) root access vulnerability
  - The Cisco APIC is a central component of Cisco's Application Centric Infrastructure (ACI), which is a software-defined networking (SDN) solution used for managing and automating network infrastructure.
  - This vulnerability allowed unauthorized users to gain root access to the Cisco APIC device, potentially giving them full control over the system.
  - https://sec.cloudapps.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20150722-apic

Hardening an SDN System

- Use TLS 1.2 (or UDP/DTLS) to authenticate and encrypt traffic between network device agent and controller, authenticate controller and network devices / SDN agent using certificate.
- High-Availability controller architecture
- Prevent unauthorized access to SDN control network
- Use out-of-Band (OOB) network for control traffic, OOB and secure protocols for controller management and northbound communications
- Harden the controller and the network elements (typical host hardening)

Hardening an SDN System

- Closely monitor controllers for suspicious activity
- Secure coding practices for all northbound applications requesting SDN resources
- Ability to validate flows in network devices table against controller policy
- Use data center interconnect (DCI) protocols that can authenticate tunnel endpoints and secure tunneled traffic

Hardening an SDN System

- Security Technical implementation Guides (STIGs) document the hardening procedures

- The **DISA Draft SDN STIG** (Security Technical Implementation Guide) version 1 for the Information Assurance Support Environment (IASE) is a set of guidelines and requirements developed by the Defense Information Systems Agency (DISA) to enhance the security of Software-Defined Networking (SDN) within the Information Assurance Support Environment
  - https://public.cyber.mil/stigs/downloads/

Hardening an SDN System

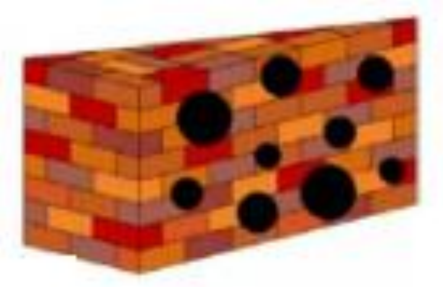
- Vmware NSX meets STIG for DOD FOUO
  - implies that VMware's NSX product has been assessed and found to with the Security Technical Implementation Guide (STIG) requirements set by the Department of Defense (DoD) for information classified as "For Official Use Only" (FOUO).
  - https://news.vmware.com/releases/newly-released-stig-validates-vmware-nsx-meets-the-security-hardening-guidance-required-for-installment-on-department-of-defense-dod-networks

Tails: SDN Security –Specific Use Case

- SDN allows for creative new approaches to security
- We will now review 6 SDN uses cases for security
  - Traffic filtering with SDN and Software Defined Perimeter
  - Network Slicing, Campus Slicing, Multi-Tenancy, Enclaves, Isolation, Network Segmentation
  - DDoS Mitigation
  - Network Access Control (NAC)
  - Security Traffic Monitoring
  - Moving Target Defense (MTD)

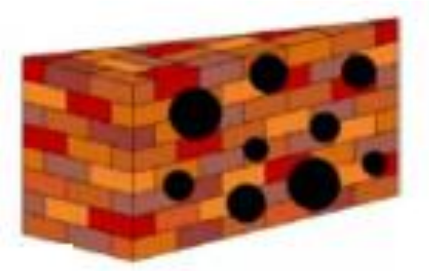
Traffic Filtering with SDN

- That which is not permitted is denied – make the SDN switches not transparent learning/forwarding
- **Cisco APIC-EM (Cisco Application Policy Infrastructure Controller Enterprise Module)→** configures the ACI (Application Centric Infrastructure) policy for traffic permitted between End Point Groups (EPGs) and for traffic steering – if not permitted, traffic is dropped
  - **APIC-EM** is an **SDN controller that was created for Enterprise hardware**. It uses a REST API for the northbound API with a decent GUI.
  - For the southbound interface, it uses common protocols like *Telnet*, *SSH* and *SNMP* to communicate with your hardware.
  - The **ACI Policy Model** is a software-defined networking (SDN) approach that enables network administrators to manage their networks in a more efficient and flexible way.



Traffic Filtering with SDN

- Integrate SDN system with Cisco Identity Services Engine (ISE) for **device profiling**, **user authentication**, **SGT**, **TrustSec tagging.**
  - **ISE** provides secure access to network resources by enforces policies for user and device authentication, authorization, and accounting (AAA).
  - **SGT (Security Group Tag)** specifies the privileges of traffic source within a trusted network.

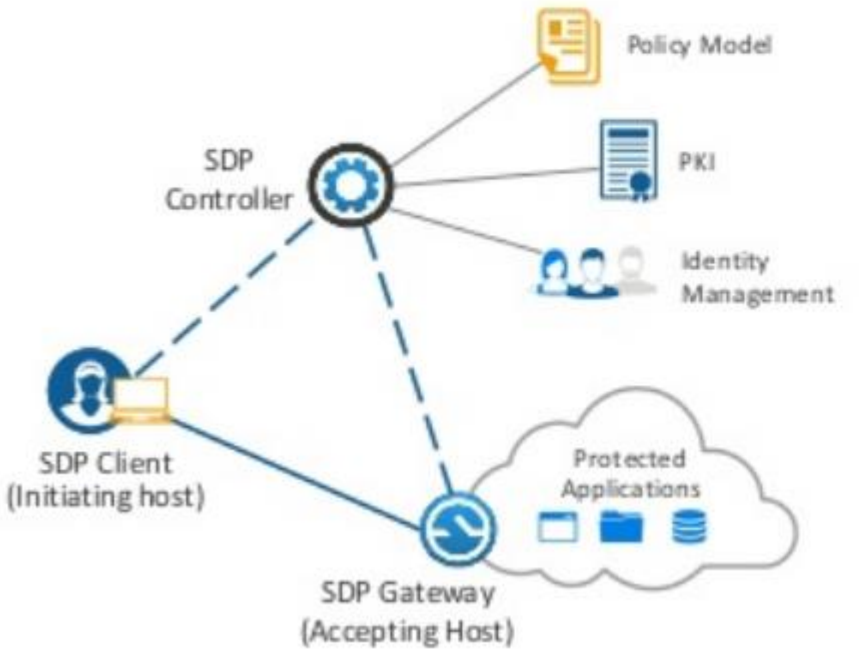


- Traffic steering toward firewall or content filter, security service insertion between client and server

Software Defined Perimeter (SDP)

- **Software-Defined Perimeter (SDP)** is a security framework designed to provide secure, dynamic, and scalable access control to network resources.

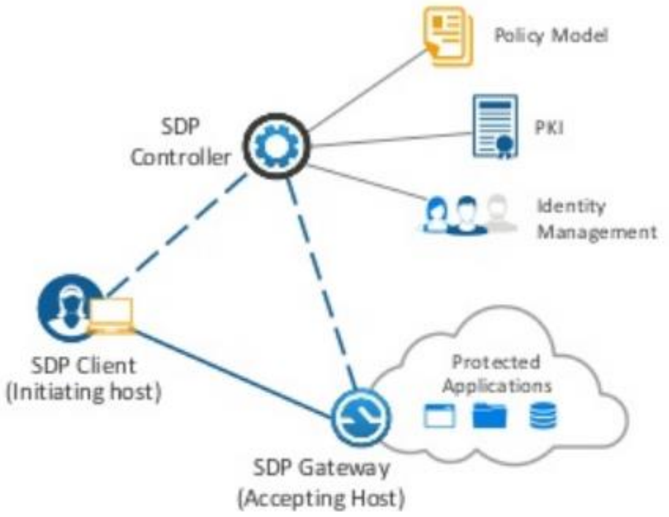
- It shifts the **traditional network security model** from perimeter-based defenses to a model where access is granted based on identity and context.



Software Defined Perimeter (SDP)

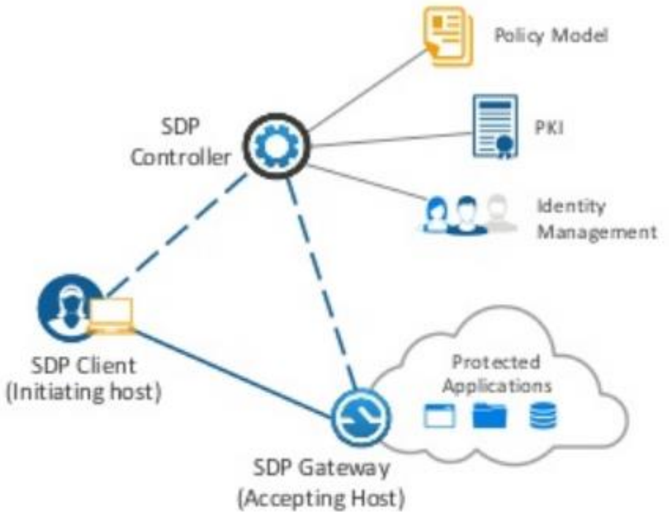
❑ Key Components of SDP

1. Controller:
   - Acts as the central management point.
   - Authenticates and authorizes users and devices.
   - Enforces security policies.
   - Establishes secure connections between clients and resources.



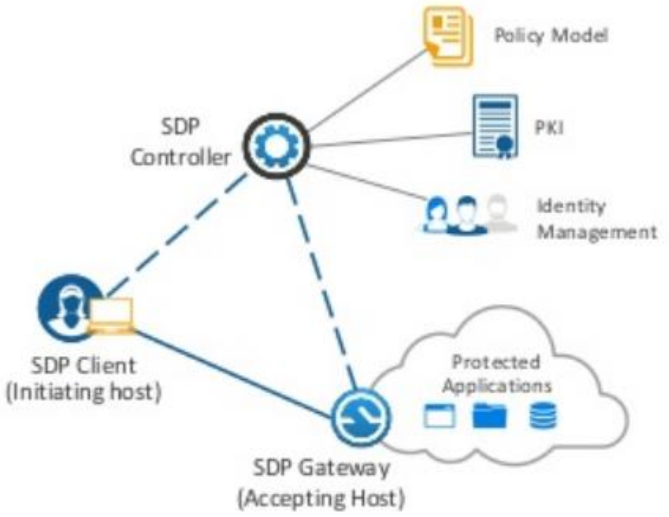
Software Defined Perimeter (SDP)

2. Client:
   - Installed on user devices (laptops, smartphones, etc.).
   - Requests access to resources.
   - Communicates with the controller for authentication and authorization.



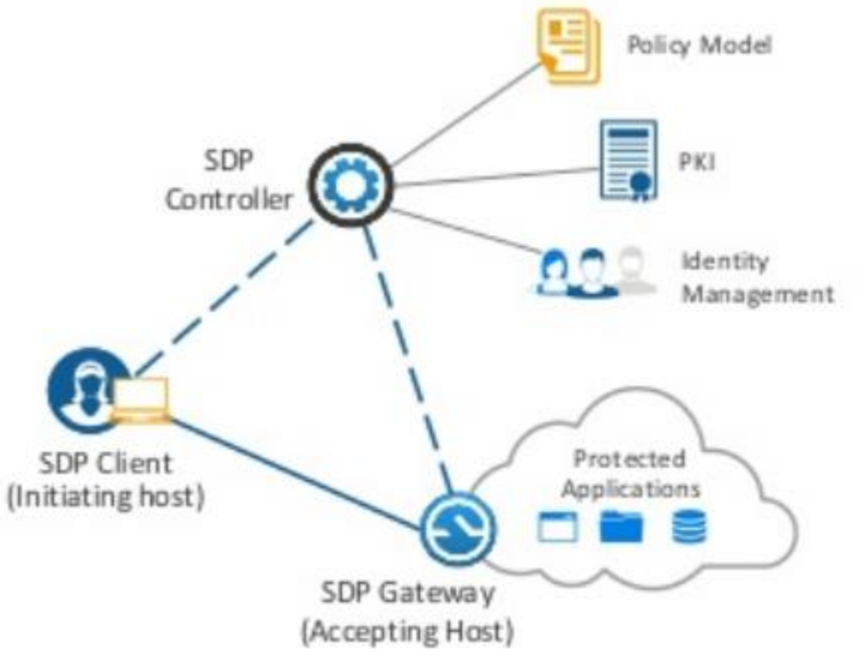
Software Defined Perimeter (SDP)

3. Gateway:
   - Protects the resources within the network.
   - Manages and enforces access policies.
   - Establishes secure connections with authenticated clients.
   - Can be deployed as virtual or physical appliances.



Software Defined Perimeter (SDP)

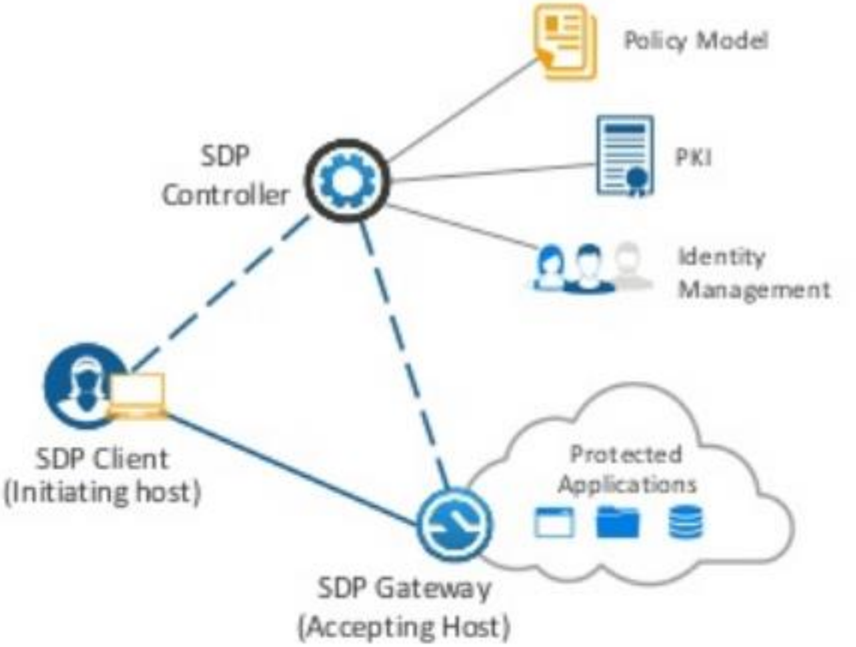
- Controller is the authentication point, containing user access policies.
- Clients are securely onboarded.
- All connections based on mutual TLS connectivity.
- Traffic is securely tunneled from Client through Gateway



Software Defined Perimeter (SDP)

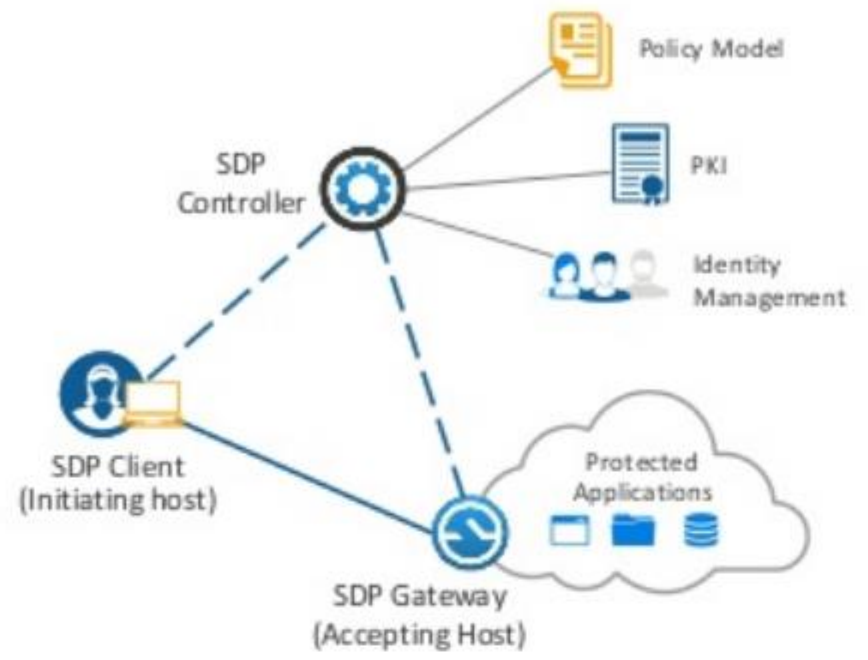
❑ **How SDP Works**

1. Authentication:
   - When a client wants to access a resource, it first communicates with the SDP controller.
   - The client authenticates itself using credentials (username/password, certificates, multi-factor authentication, etc.).



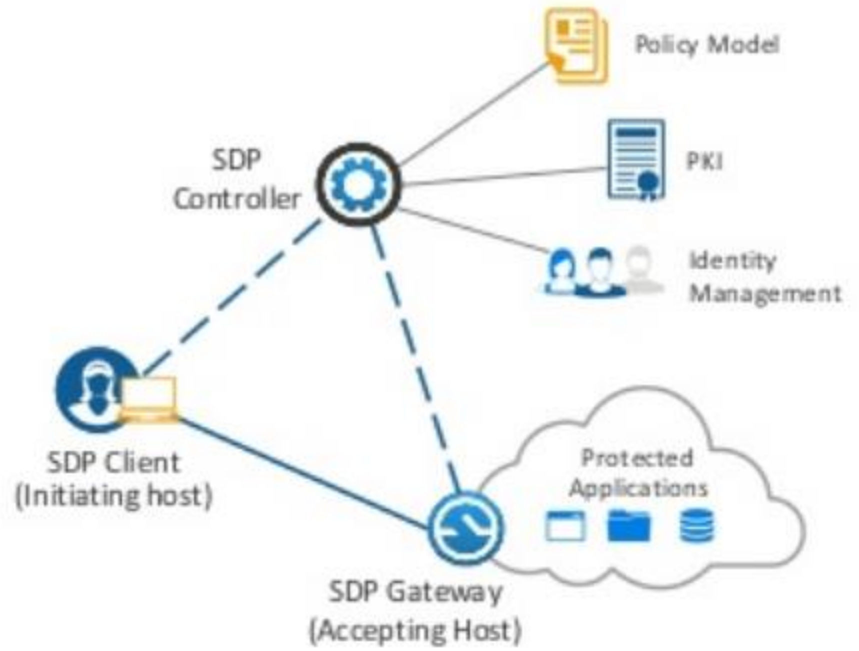
Software Defined Perimeter (SDP)

2. Authorization:
   - After successful authentication, the controller evaluates whether the client is authorized to access the requested resource.
   - Authorization is based on predefined policies, which can include factors like user role, device type, location, and time of access.



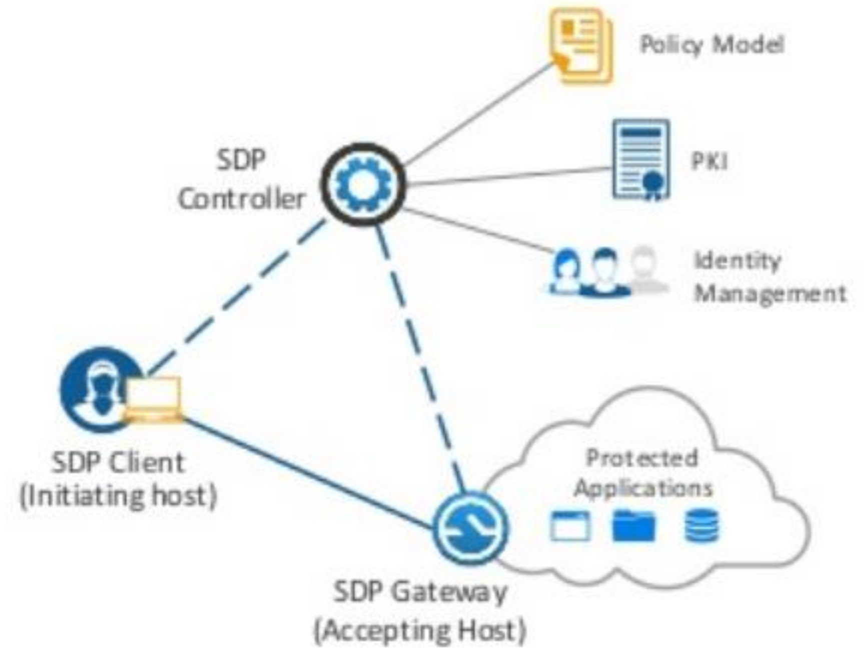
Software Defined Perimeter (SDP)

3. Secure Connection Establishment:
   - If the client is authorized, the controller instructs the gateway to establish a secure connection with the client.
   - This connection is typically encrypted, ensuring data confidentiality and integrity.



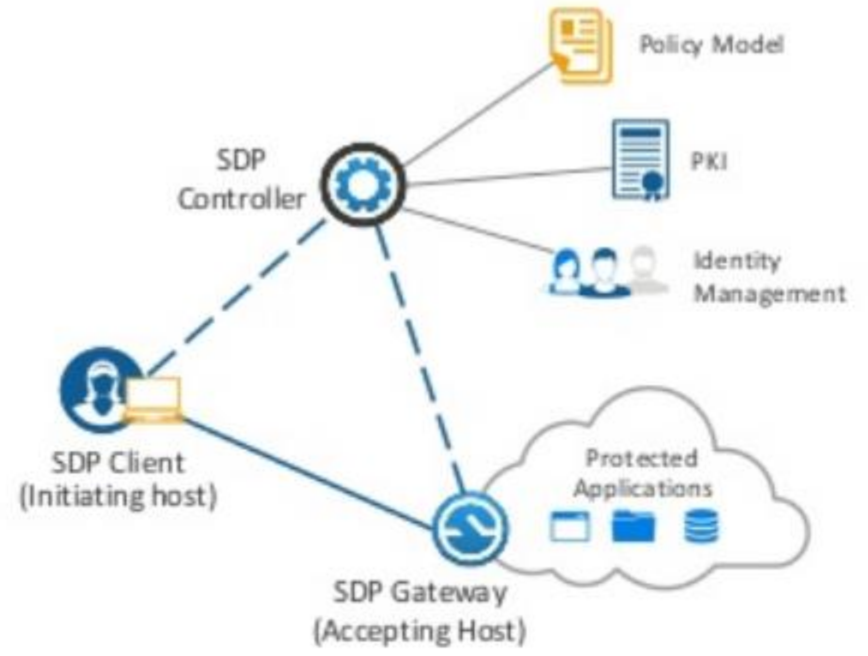
Software Defined Perimeter (SDP)

4. Access Control and Monitoring:
   - The gateway enforces access control policies, ensuring that the client can only interact with the specific resources they are authorized to access.
   - The controller and gateway continuously monitor the connection for any signs of malicious activity or policy violations.



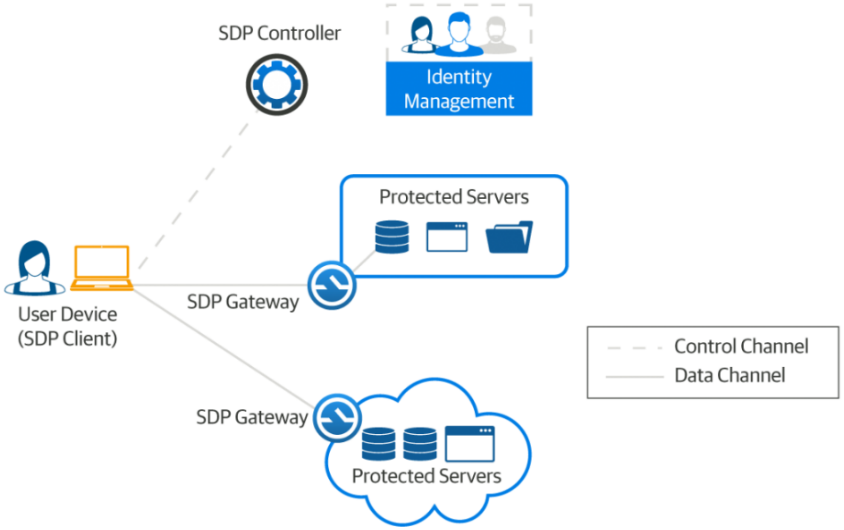
Software Defined Perimeter (SDP)

4. Dynamic Adjustment:
   - SDP can dynamically adjust access permissions based on changing conditions (e.g., the user's context changes, such as moving to a different network or location).
   - Policies can be updated in real-time to adapt to new security threats or changes in organizational requirements.



Software Defined Perimeter (SDP)

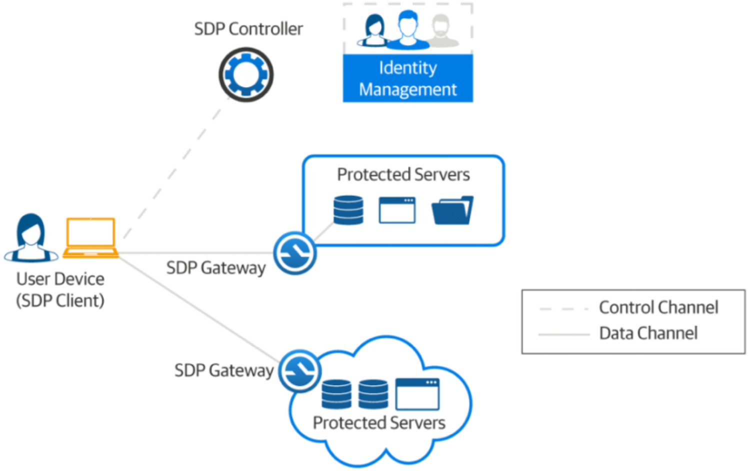
- The basic premise of a **Software-Defined Perimeter (SDP)** is built on an "authenticate first, connect second" approach.
- Unlike a traditional network that connects various roles or groups to a network segment and then relies on application level permissions for authorization, a Software-Defined Perimeter creates individualized perimeters for each user, *allowing for much more fine grained access control*.



Software Defined Perimeter (SDP)

❑ **Using your own home as an example**

- Imagine someone is knocking on your front door.
- The Software-Defined Perimeter zero-trust approach takes the person that is knocking at your front door, confirms who the person is and what it is that they want/ need, and then opens the door to let them inside of the house.
- Once inside the house, they can only access those rooms that they need, and nothing else.

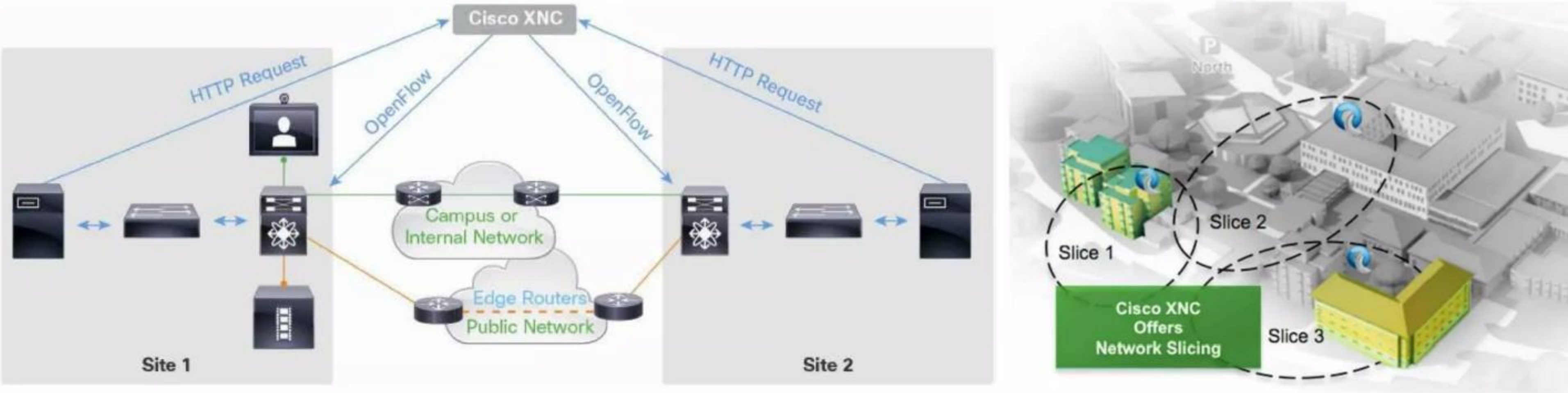


Network Segmentation with SDN

- Separating the network into **logically separated networks**
- *Network Slicing*, *Campus Slicing*, *Secured Enclaves*, *Micro-Segmentation*, *Virtual Routing and Forwarding*, etc.
- Done by adding a slicing layer between the control plane and the data plane, *policies are slice-specific*
- Enforce strong isolation between slices – actions in one slice do not affect another (Flowspace)
- Examples: Cisco XNC with Networking Slicing application, FlowVisor is a special purpose OpenFlow controller that acts as a transparent proxy between OpenFlow switches and multiple OpenFlow controllers.

Network Segmentation with SDN

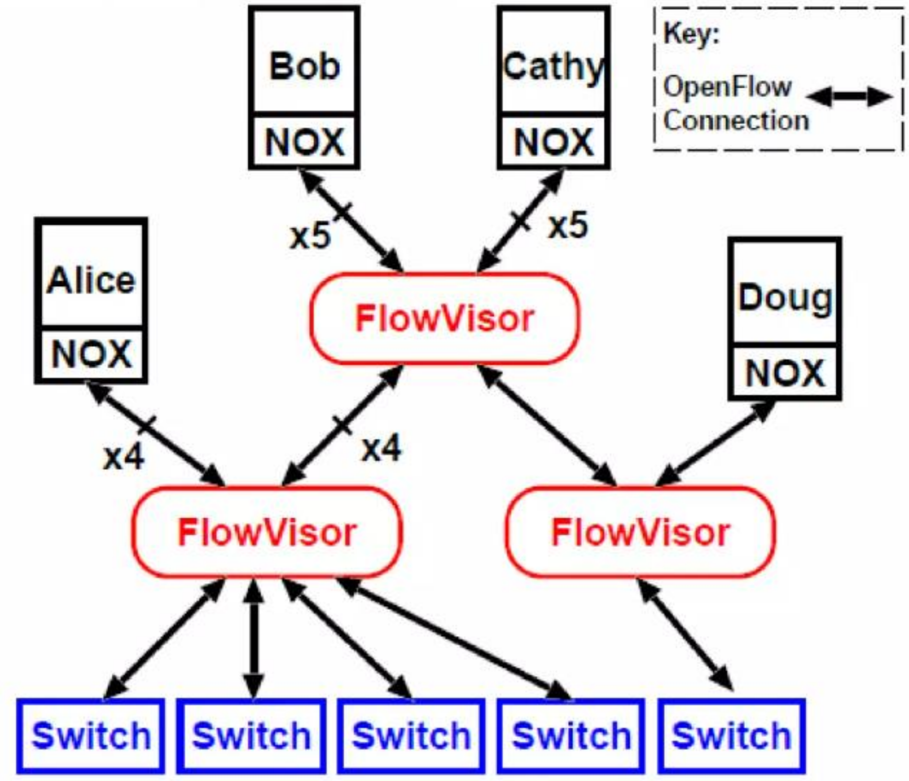
- Network Slicing Use case



Source: Cisco Extensible Network Controller Topology-Independent Forwarding and Network Slicing Applications
http://www.cisco.com/en/US/prod/collateral/netmgtsw/ps13397/ps13400/data_sheet_c78-729458.pdf

Network Segmentation with SDN

- FlowVisor performs policy checks across flowspace and enforces isolation between each slcie,.



Source: Can the Production Network Be the Testbed?
By Rob Sherwood, Glen Gibb, Kok-Kiong Yap, Guido Appenzeller ,Martin Casado, Nick McKeown, Guru Parulkar

DDoS Mitigation with SDN

- SDN can be used to create a DDoS mitigation system
- SDN network sends DDoS telemetry data to the DDoS detection system (*volumetric*, *app attacks*, *protocol DDoS*)
- DDoS detection system communicates with northbound API which configures the policy on the controller for the destination of the attack
- SDN controller sends flows to network devices to drop suspicious inbound traffic toward victim
- Cleaned traffic is allowed to pass toward the destination
- Examples: Radware Defense Flow, Radware Defense4All in ODL Helium, A10 Networks Thunder Threat Protection System, others …

DDoS Mitigation with SDN

- ❑ **Radware's DefenseFlow and DefensePro** are integral components in network security, particularly when integrated with Software-Defined Networking (SDN).

- ❑ **Understanding the Components**

1. DefenseFlow:
   - **Purpose:** DefenseFlow is a network-wide attack detection and mitigation solution designed to operate within SDN environments.
   - **Functionality:** It identifies attacks in real-time and orchestrates mitigation actions.
   - **Integration with SDN:** It leverages SDN controllers to dynamically adapt network policies and reroute traffic.

DDoS Mitigation with SDN

1. DefensePro:
   - **Purpose:** DefensePro is an intrusion prevention system (IPS) that protects against network and application-level attacks.
   - **Functionality:** It provides DDoS protection, behavioral analysis, and real-time mitigation.

DDoS Mitigation with SDN

❑ **How DefenseFlow and DefensePro Work with SDN**

Detection and Mitigation Workflow:

1. Traffic Monitoring:
   - DefenseFlow continuously monitors network traffic for anomalies using data provided by SDN controllers and network elements (switches and routers).
   - It uses techniques such as flow analysis, anomaly detection, and pattern recognition.

DDoS Mitigation with SDN

2. Attack Detection:
   - Upon detecting a potential threat, DefenseFlow analyzes the traffic to identify attack characteristics.
   - It can detect various attack types, including volumetric DDoS, application-layer attacks, and low-and-slow attacks.

DDoS Mitigation with SDN

3. Policy Update and Traffic Rerouting:
   - DefenseFlow communicates with the SDN controller to update network policies dynamically.
   - The SDN controller enforces these policies by rerouting malicious traffic to DefensePro appliances for inspection and mitigation.
   - This ensures that legitimate traffic remains unaffected while the threat is contained.

DDoS Mitigation with SDN

4. Traffic Mitigation:
   - DefensePro inspects the redirected traffic in real-time.
   - It employs multiple mitigation techniques, such as rate limiting, traffic filtering, and signature-based detection, to neutralize the threat.
   - Clean traffic is then forwarded back to the network, ensuring minimal disruption.

DDoS Mitigation with SDN

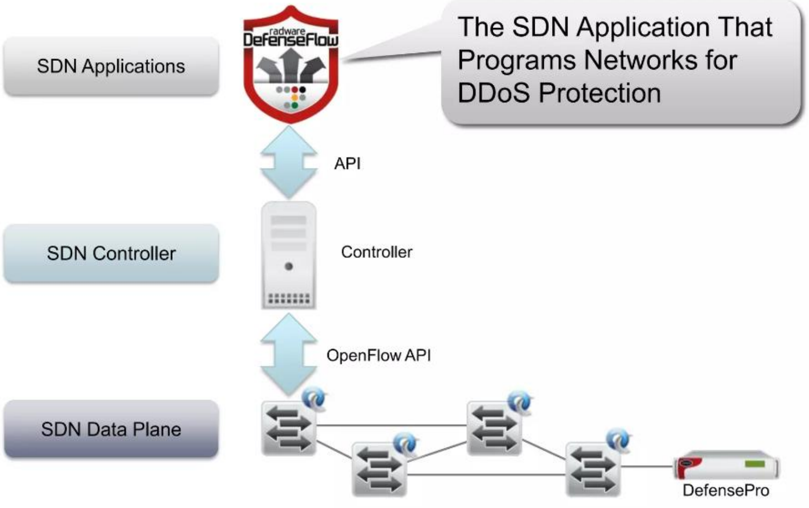
5. Feedback Loop:
   - DefensePro provides feedback to DefenseFlow on the nature and characteristics of the mitigated attack.
   - This information is used to refine detection algorithms and update mitigation strategies.

DDoS Mitigation with SDN

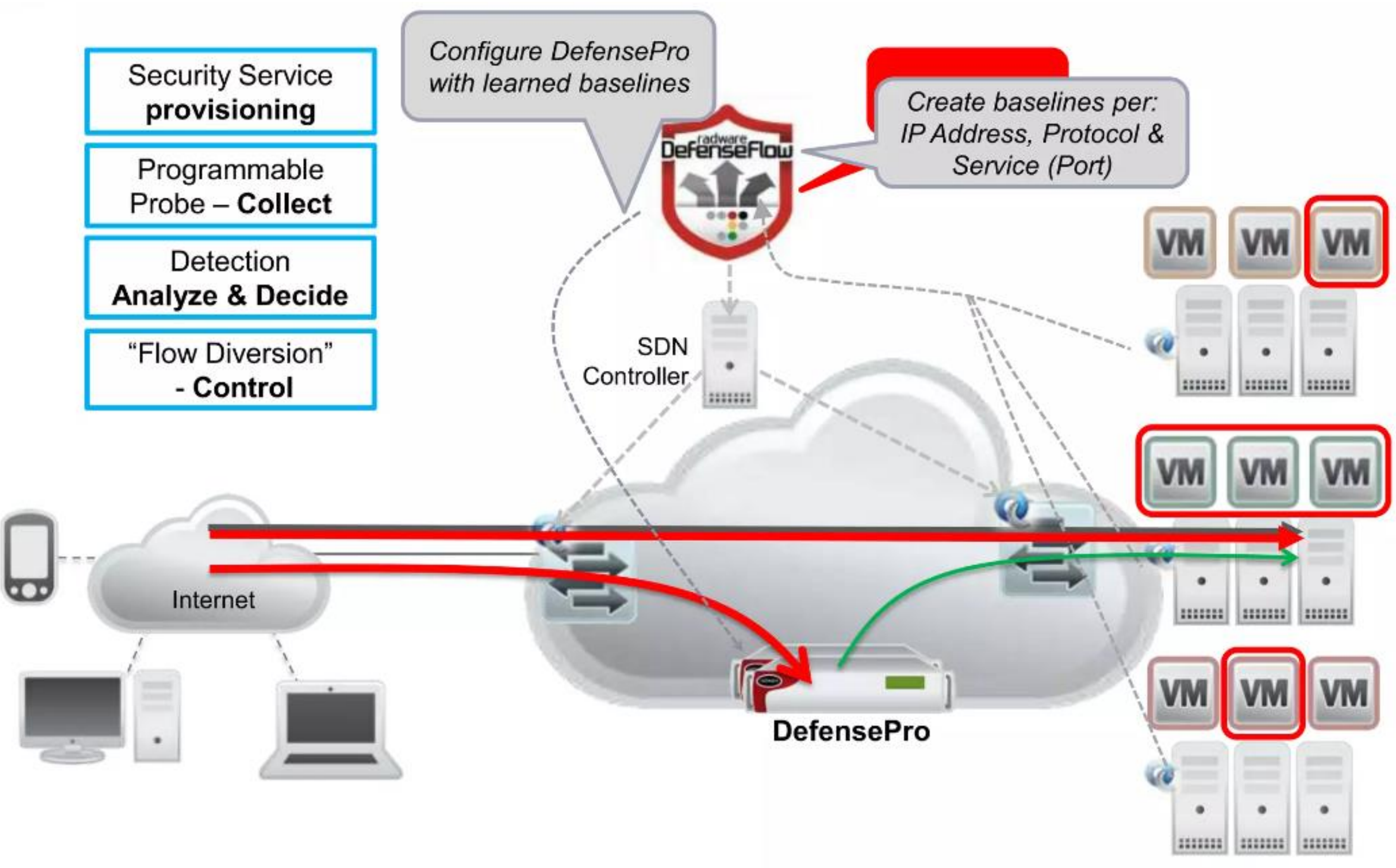
- Radware Defenseflow integrates with Cisco's XNC, OpenDaylight, BigSwitch Floodlight, and NEC's Programmable Flow OpenFlow-based switches and controller.

DefensePro: mitigation devices can be placed in any location.

DefenseFlow: diverts the traffic to the nearest mitigation device.



DDoS Mitigation with SDN


Security Service **provisioning**
Programmable Probe – **Collect**
Detection **Analyze & Decide**
"Flow Diversion" - **Control**
Configure DefensePro with learned baselines
radware DefenseFlow
Create baselines per: IP Address, Protocol & Service (Port)
VM
SDN Controller
Internet
DefensePro

Network Access Control (NAC) with SDN

- SDN system can prevent unauthorized access or isolate compromised hosts to a quarantine network, Automated Malware Quarantine (AMQ).
- SDN systems can intervene in assigning addresses to nodes joining network based on their security posture
- Authenticated end nodes are able to send/receive if they pass security checks (Antivirus (AV) running/updated/ patched, …)
- End nodes can only send/receive with their assigned IP/MAC addresses
  - Source Address Validation Improvement (SAVI) and First Hop Security (FHS)
- Or direct end-node traffic to Cisco Cloud Threat Defense system, detect the issue, check with ISE, set SGT=BAD, to contain the traffic
- Examples: Cisco Cloud Threat Defense, HP VAN Sentinel Security Application

Network Access Control (NAC) with SDN

❑ **Network Access Control (NAC) and Software-Defined Networking (SDN)** can work together to provide enhanced network security and management capabilities. Here's how:

1. Centralized Policy Management:
   - SDN Controller: The SDN controller acts as the central point for enforcing network policies. NAC policies are defined and communicated to the SDN controller.
   - Dynamic Policy Enforcement: Policies can be dynamically updated and enforced across the network without manually reconfiguring each device

Network Access Control (NAC) with SDN

2. Network Visibility and Monitoring:
   - Real-time Monitoring: SDN provides a holistic view of the entire network, allowing NAC solutions to continuously monitor and assess the state of the network.
   - Anomaly Detection: By analyzing traffic patterns centrally, SDN can help NAC detect and respond to anomalies more quickly.

3. Automated Response:
   - Dynamic Access Control: When a device or user tries to access the network, NAC can dynamically enforce access policies using the SDN controller.
   - Quarantine and Remediation: If a device is non-compliant or compromised, NAC can instruct the SDN controller to isolate the device or reroute its traffic to a remediation network.

Network Access Control (NAC) with SDN

4. Scalability and Flexibility:
   - Scalable Network Management: SDN enables scalable and flexible management of large, complex networks, enhancing the scalability of NAC solutions.
   - Policy Flexibility: Policies can be easily adjusted and deployed across the entire network without physical changes to the infrastructure.

Network Access Control (NAC) with SDN

5. Enhanced Security Posture:
   - Unified Security Policies: Combining NAC with SDN allows for unified security policies that are consistently applied across all network segments.
   - Reduced Attack Surface: SDN can segment the network logically, reducing the attack surface and preventing lateral movement of threats.

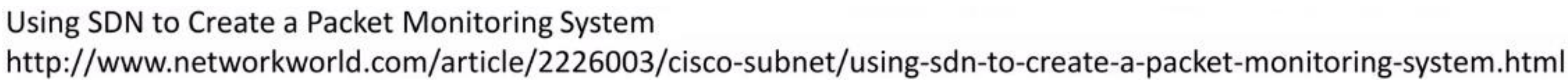
Network Access Control (NAC) with SDN

❑ **Example Scenario**

1. **Access Request:** A user tries to connect to the network with their device.
2. **Policy Check**: The NAC solution checks the user's credentials and the device's compliance status.
3. **SDN Integration**: Based on the policy check, the NAC solution communicates with the SDN controller to enforce the appropriate policy.
   - If **compliant**, the SDN controller grants access and configures network paths accordingly.
   - If **non-compliant**, the SDN controller isolates the device or redirects it to a remediation network.

Security Monitoring with SDN

- Switches often lack sufficient resources to perform packet/port mirroring/taps
  - Every IT silo/team wants their own tap/SPAN session (Network Packet Broker (NPB))
- Bi-directional packet capture is much better than NetFlow
- Dedicated copper/optical packet monitoring switches can be very expansive, many taps are required –no blocking ability
- Tap Aggregation is an application that is simple for a SDN controller and uses low-cost SDN-capable network devices
- Examples: Cisco XNC with monitor Manager and Nexus 3000Tap Aggregation Switch, BigSwitch Big Tap Monitoring Fabric, Microsoft Distributed Ethernet Monitoring (DEMon)



Using SDN to Create a Packet Monitoring System
http://www.networkworld.com/article/2226003/cisco-subnet/using-sdn-to-create-a-packet-monitoring-system.html

Security Monitoring with SDN

❑ Security monitoring in Software-Defined Networking (SDN) environments, particularly through tap aggregation, leverages the programmability and flexibility of SDN to efficiently monitor and manage network traffic.

❑ **Understanding Tap Aggregation**

- Tap aggregation involves capturing traffic from various points in the network and aggregating it for monitoring and analysis. In traditional networks, this is done using physical taps or port mirroring, which can be costly and complex to manage.

Security Monitoring with SDN

❑ **Components and Workflow**

1. SDN Controller
   – Acts as the brain of the network.
   – Manages the flow rules and policies for traffic monitoring.
   – Provides a centralized interface to configure tap aggregation points.

2. SDN-Capable Network Devices
   – Switches and routers that support SDN protocols like OpenFlow.
   – These devices can be programmed to mirror traffic to specific ports based on the controller's instructions.

Security Monitoring with SDN

3. Monitoring Tools
   - Tools that receive mirrored traffic for analysis.
   - These can include intrusion detection systems (IDS), performance monitoring systems, and logging systems.

Security Monitoring with SDN

❑ **Tap Aggregation Process with SDN (Step-by-Step Process):**

1. Define Monitoring Policies:
   - Network administrators define policies for what traffic needs to be monitored.
   - These policies can include specific flows, ports, IP addresses, or types of traffic.
2. Controller Configuration:
   - The SDN controller receives the policies and translates them into flow rules.
   - These rules specify which traffic to mirror and where to send it.

Security Monitoring with SDN

3. Programming the Network Devices:
   - The controller programs SDN-capable devices with the necessary flow rules.
   - Devices start mirroring the specified traffic to designated ports or paths.
4. Traffic Aggregation:
   - Mirrored traffic is aggregated from multiple sources.
   - This aggregation can occur at specific aggregation points or devices optimized for handling high volumes of mirrored traffic.
5. Delivery to Monitoring Tools:
   - Aggregated traffic is sent to the security monitoring tools.
   - These tools analyze the traffic for anomalies, performance issues, or security threats.

Security Monitoring with SDN

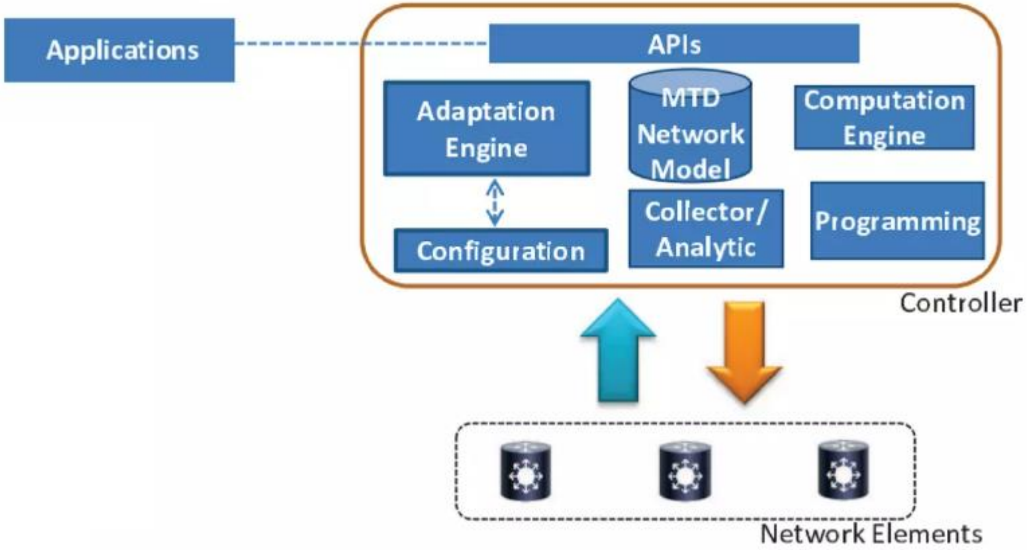
❑ Example Scenario

Imagine an enterprise network with multiple branches:

- **Policy Definition**: The security team defines a policy to monitor all HTTP traffic across the network.
- **Configuration**: The SDN controller configures switches at each branch to mirror HTTP traffic to a central aggregation point.
- **Aggregation**: The mirrored traffic is aggregated at a core switch and forwarded to a centralized IDS.
- **Analysis**: The IDS analyzes the aggregated traffic for potential threats, providing insights to the security team.

Moving Target IPv6 Defense (MT6D)

- MT6D is a system created by graduate student in the information Technology Security Laboratory at Virginia Tech to obscure IPv6 addresses
- Periodically hiding/changing characteristics of victim to make it more difficult to find/ attack



SDN Security Summary

- SDN has the potential to provide many new creative ways to network and secure systems.
- SDN represents a new way of thinking, we all need to be cognizant about this technology shift
- Heads: SDN systems are vulnerable to threats, but SDN implementations can be hardened against security attacks
- Tails: SDN systems can provide innovative security applications that are not possible with traditional methods