Hewlett Packard
Enterprise

Dynamic Network Perimeterization

Isolating Tenant Workloads w/ VLANs, VNIs, and ACLs

Dennis Walker, Siri Vias Khalsa, Amit Jain, Stephen Han, Nikhil Mukundan, Atif Ali, Vinay Karanth

May 4, 2025

Agenda

- Use Case Review: Why Multitenancy? Why Network Partitioning?

- Mechanisms of Isolation: VLANs, VNIs, and ACLs

- Software Features for Network Partitioning: Slingshot, HPCM, and CSM
  - Code/Implementation Example

- Case Study:
  - Partitions
  - Mechanisms of partitioning
  - Configuration Management
  - End Result

- Recap, Further Reading, Q&A

Why Multi-tenancy? Why Partition Networks?

Not everyone has the same **security clearance**

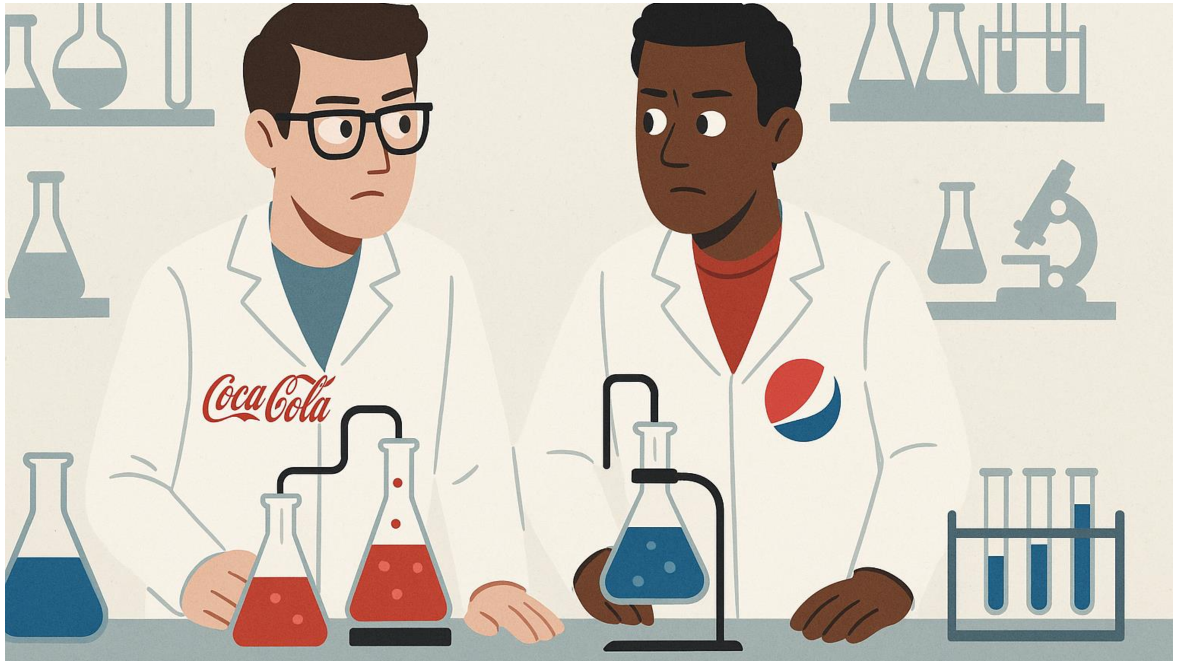
Not everyone has the same **priority**

Not everyone knows how to **run jobs safely**

To render all compute nodes for use for every possible minute, we must make them available to more people.

“Good fences make good neighbors.”

Network Partitioning – Use Cases



Separate Tenants: Coke and Pepsi

Some groups needs more rigorous privacy guarantees.

Secure Administrative Functions

Most critical exploit remediation plans involve completely reinstalling all software from scratch, potentially requiring weeks of downtime.

Network Partitioning – Use Cases (cont.)

- Grouping Jobs, Isolating Jobs, and Job Steps

- Compliance controls demand more rigor when accessing private data. Securing data means limiting the scope of access.

Network Partitioning Isolation Mechanisms

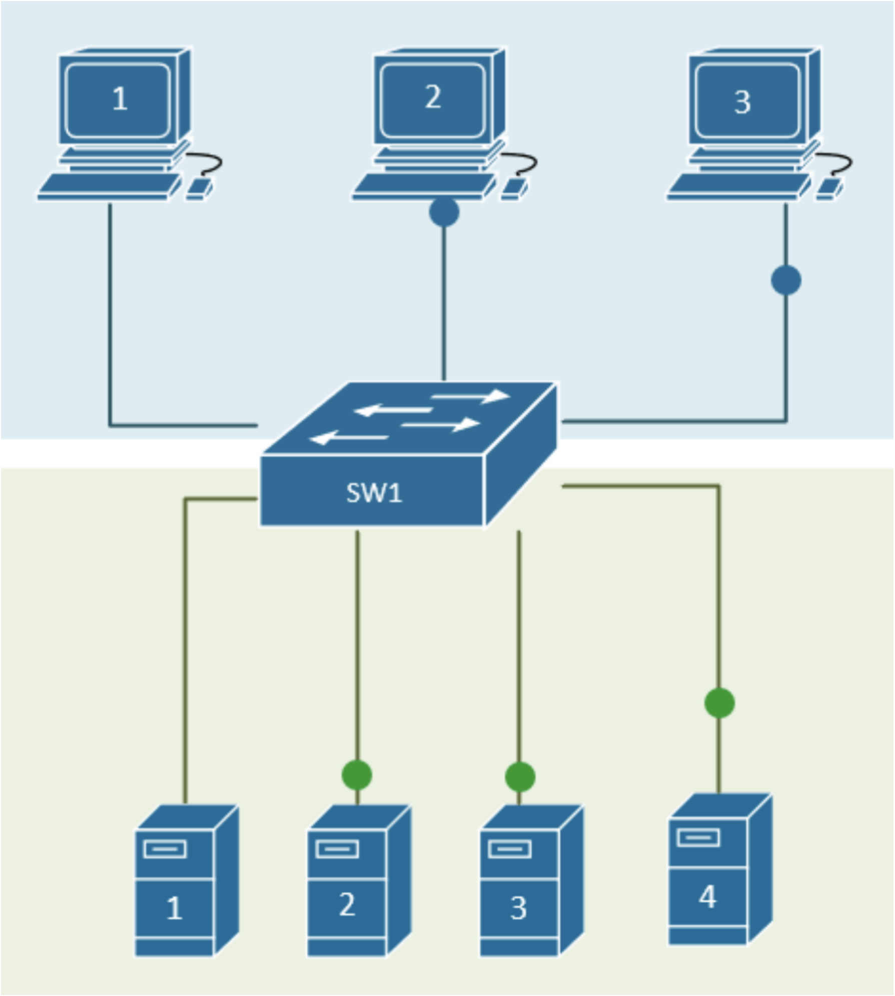
VLAN (Virtual LAN): A logical segmentation of IP traffic at Layer 2 to isolate traffic within defined broadcast domains.

VNI (Virtual Network Identifier): A packet label embedded in Slingshot Transport RDMA enabling fine-grained isolation policies.

ACL (Access Control List): A rule-based filter applied to network traffic to permit or deny packets based on IP, port, or protocol.

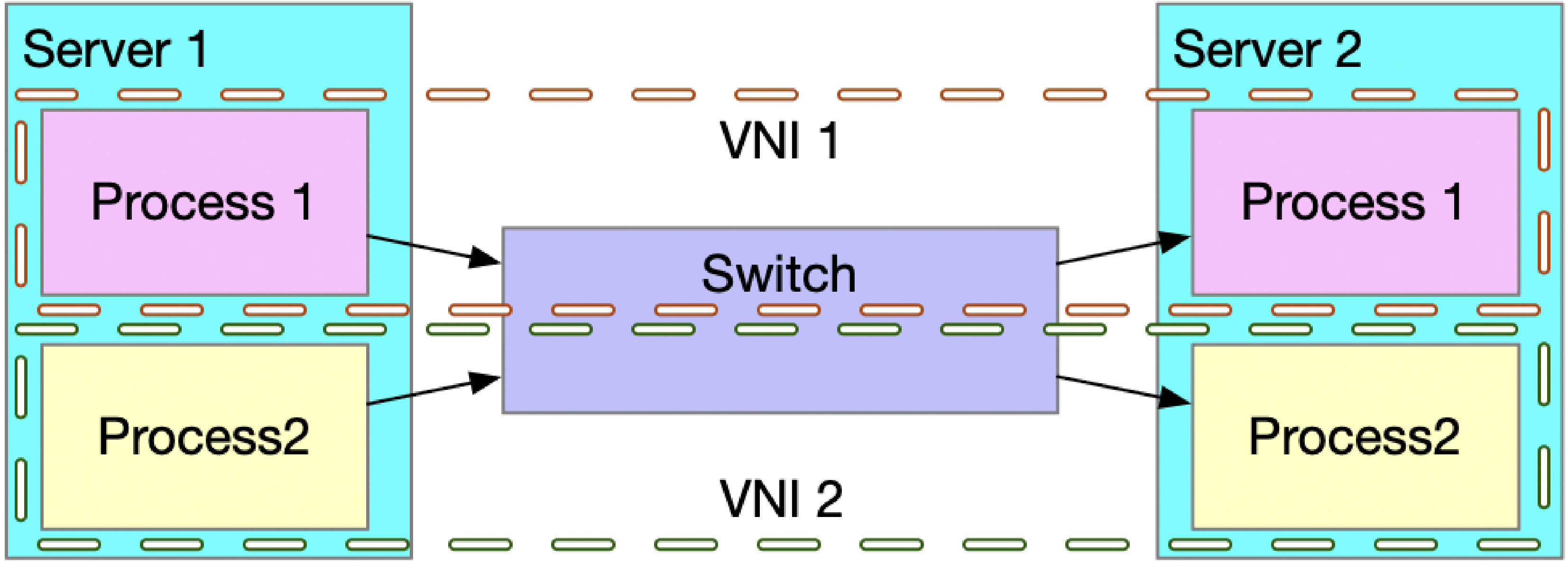
VLAN Partitioning

- Layer 2 tagging (802.1q) of Ethernet
- Available in management networks and high-speed networks
- Can be trunked to enable specific routes
- Can be assigned at either/both the switch port or the OS NIC configuration



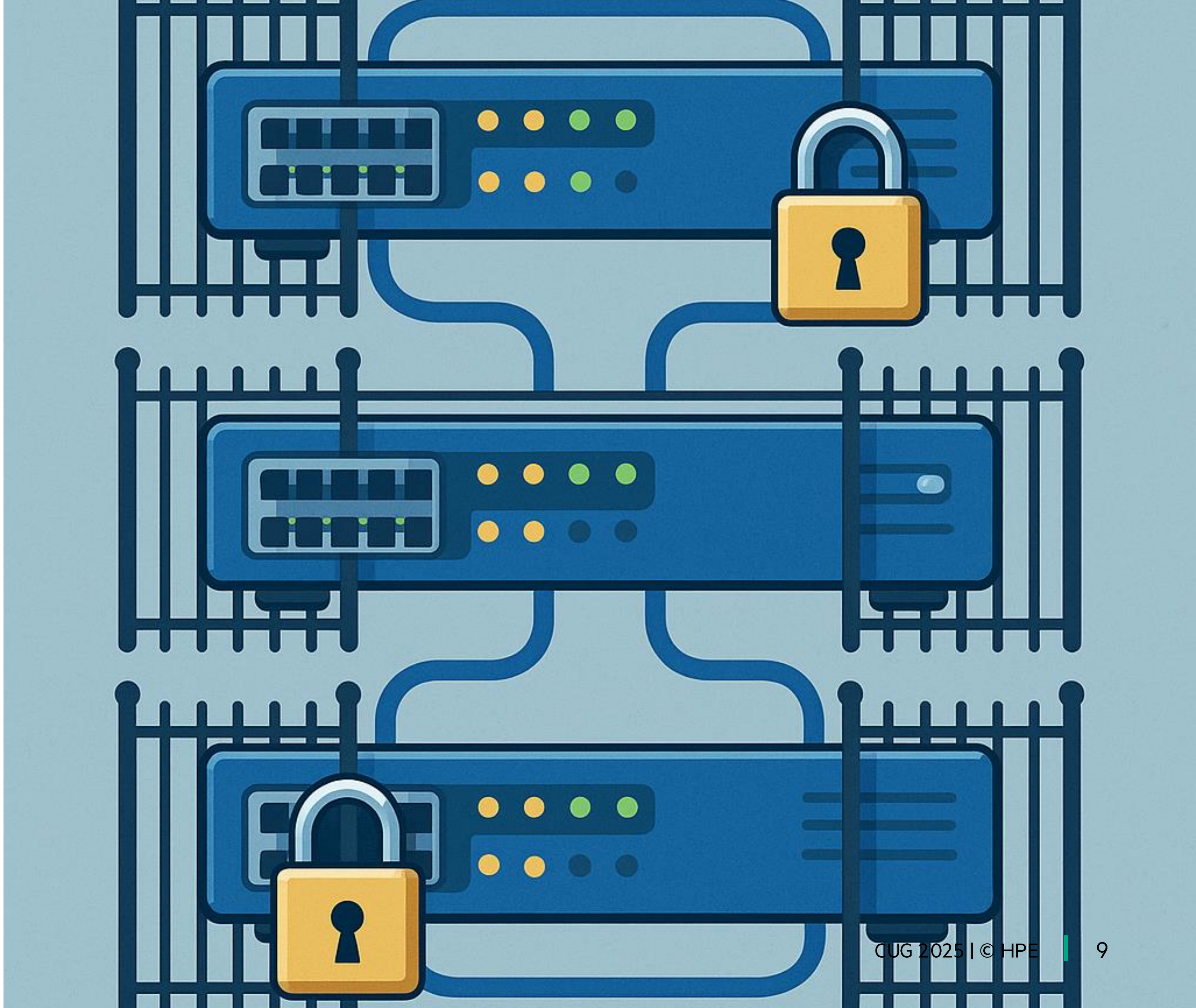
VNI (Virtual Network Identifier

- Packet label of ST RDMA used to enforce isolation
- Enables scalable multi-tenant partitioning
- Can be instrumented at the switch, the NIC, and into the requesting **process**, e.g. service or job



ACLs

- Used in the management network
- Control traffic flow at the switch port level
- Filter packets by IP, protocol, or port

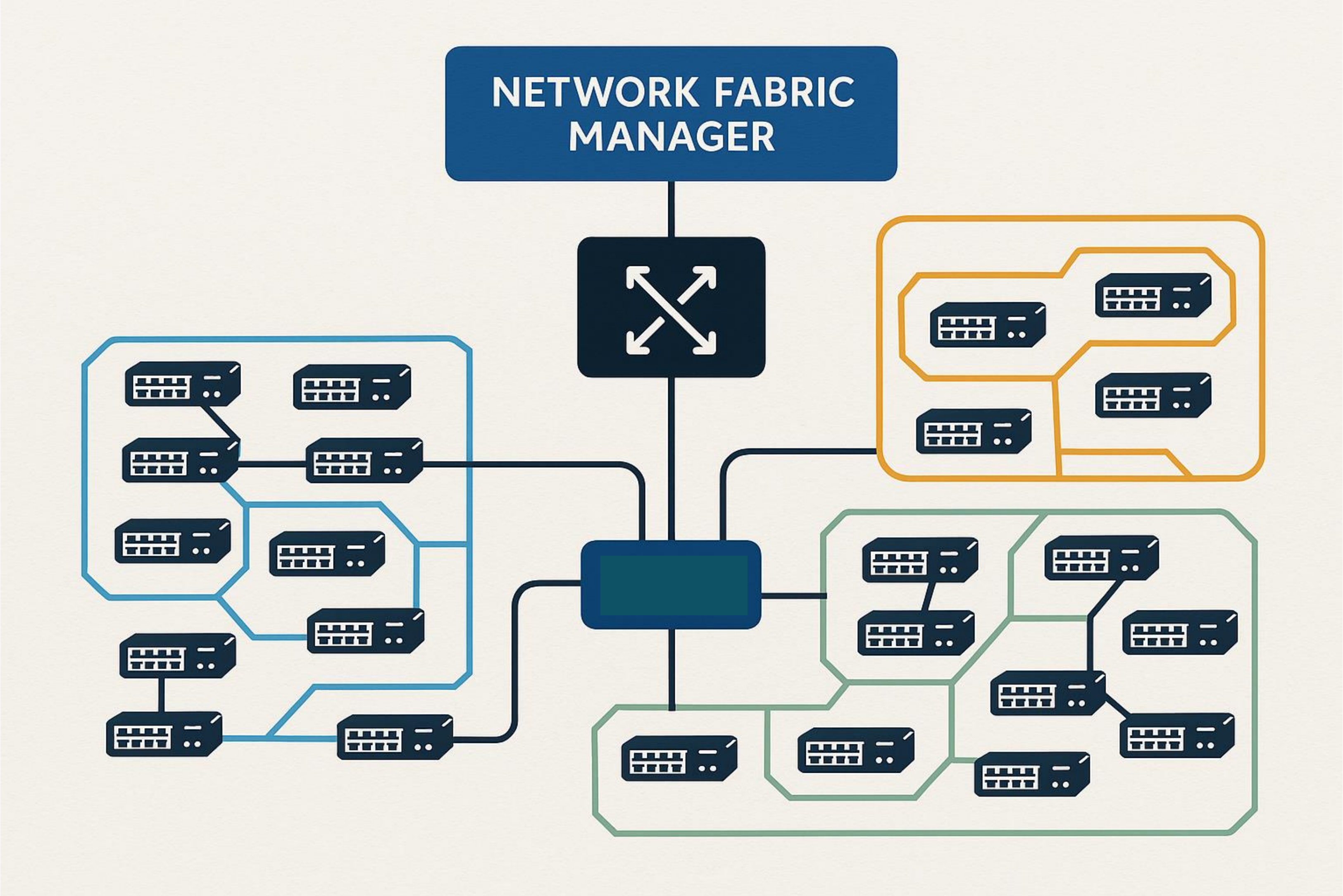


Architecture Features to Partition Networks

- Slingshot
- HPCM
- CSM

Slingshot Fabric Partitioning Features

- Fabric management (centralized orchestration of HSN switches)
- VNIs provisioning via REST API
- VLAN provisioning via REST API
- Slingshot Network Operator (Open source integration pattern based on CSM)



Example: Provisioning a VLAN in Slingshot

In this example:

- 3 VLANs are created: Red, Green, and Blue
- A port policy is created, defaulting traffic to Red, but also allowing Green
- The vlan policy is applied to port x3000c0r31j14p0

```
# Create 3 VLANs
fmctl create vlans name=RedNetwork status=ONLINE id=1
fmctl create vlans name=GreenNetwork status=ONLINE id=2
fmctl create vlans name=BlueNetwork status=ONLINE id=5
```

```
# Create a port policy file
{ "state": "ONLINE", "autoneg": true, "speed": "BJ_100G", "precode": "AUTO",
  "flowControl": {"rx": true, "tx": true },
  "mac": "02:00:00:00:00:00", "loopback": "NONE",
  "isUntaggedAllowed":false,
  "allowedVlans": ["/fabric/vlans/1","/fabric/vlans/2"],
  "nativeVlanId":"/fabric/vlans/1",
  "documentKind": "com:services:fabric:models:PortPolicyState", "documentSelfLink":
"/fabric/port-policies/vlan-policy-vlan1-2 " }
fmctl create port-policies --file <port-policy-file>
```

```
# Assign VLANs to a port
fmctl update x3000c0r31j14p0 –name vlan-policy-vlan1-2
```

Example: Provisioning a VNI in Slingshot

In this example:

- A partition is created having 200 VNIs. The response provides the provisioned range.

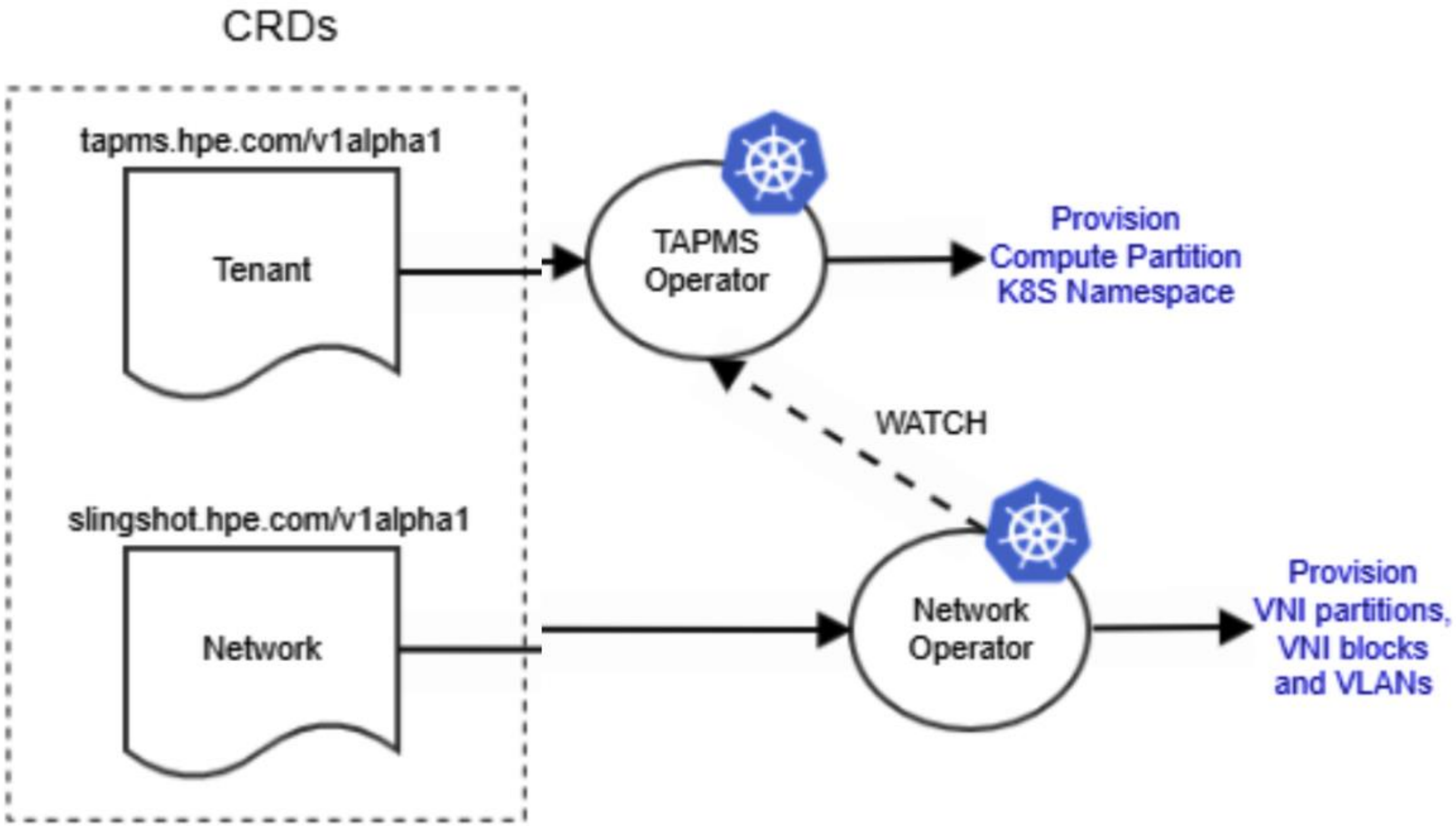
- Enforcement of VNIs is applied on the switch ports.

```
# Creating a partition of VNIs
curl -X POST "http://127.0.0.1:8000/fabric/vni/partition" \
  -u admin:YourPasswordHere \
  -H "Content-Type: application/json" \
  -d ‘{
  "partitionName": "WLM1",
  "description": "VNIs managed by WLM1",
  "vniCount”: 200,
  "edgePorts": [0x84a000, 0x8ac000, 0x1163000, 0x1532000, 0x3278a000]
}’

# Enforcing VNIs on the list of ports.
curl -X POST "http://127.0.0.1:8000/fabric/vni/block" \
  -u admin:YourPasswordHere \
  -H "Content-Type: application/json" \
  -d ‘{
  "blockName" : "WLM1-BLK1",
  "vniRange": ["2000-2100"],  # Informed by the response of the prior curl call
  "partitionName" : "WLM1",
  "portDFAs" : [0x84a000, 0x8ac000, 0x1163000, 0x1532000, 0x3278a000]
}‘
```

CSM Partitioning Features

- Tenant & Partition Mgmt (TAPMS)
- Slingshot Network Operator
- Tenant-Aware Node Management (secrets, jobs, OS images & configurations)



Example: Provisioning a Tenant in CSM

In this example:

- A tenant “tyrten02” is specified
- Two compute nodes are specified in the allocation
- One user node is specified in the allocation
- The tenant and partition is created

```
# Create Tenant Definition File – tyrten02.yml
apiVersion: tapms.hpe.com/v1alpha3
kind: Tenant
metadata:
name: tyrten02
spec:
childnamespaces:
- slurm
- user
tenantname: ExampleTenant1
tenanthooks: []
tenantresources:
- enforceexclusivehsmgroups: true
   hsmgrouplabel: tyrten02
   type: compute
   xnames:
   - x9000c1s0b1n0
   - x9000c1s0b1n1
- enforceexclusivehsmgroups: true
   hsmgrouplabel: tyrten02
   type: application
   xnames:
     - x3000c0s29b0n0
```

kubectl apply -n tenants –f ./tyrten02.yml

Example: Assigning VNIs to a Tenant in CSM

In this example:

- A block of VNIs is specified for tenant tyrten02
- The partition is applied

```
# Define a Slingshot VNI partition for the tenant – sshot_tenant.yml
apiVersion: slingshot.hpe.com/v1alpha1
kind: SlingshotTenant
metadata:
  labels:
  name: tyrten02-slingshot-parition
  namespace: tenants
spec:
  tenantname: tyrten02
  vniBlockName: Block1
  vnipartition:
    vniRanges: ["1-3000"]

kubectl –n tenants apply –f ./sshot_tenant.yml
```

Example: Provisioning a VLAN in CSM+Aruba w/ Ansible

In this example:

- A VLAN is specified
- The VLAN port assignment is specified
- Ansible applies change. Inventory is provided by CANU.

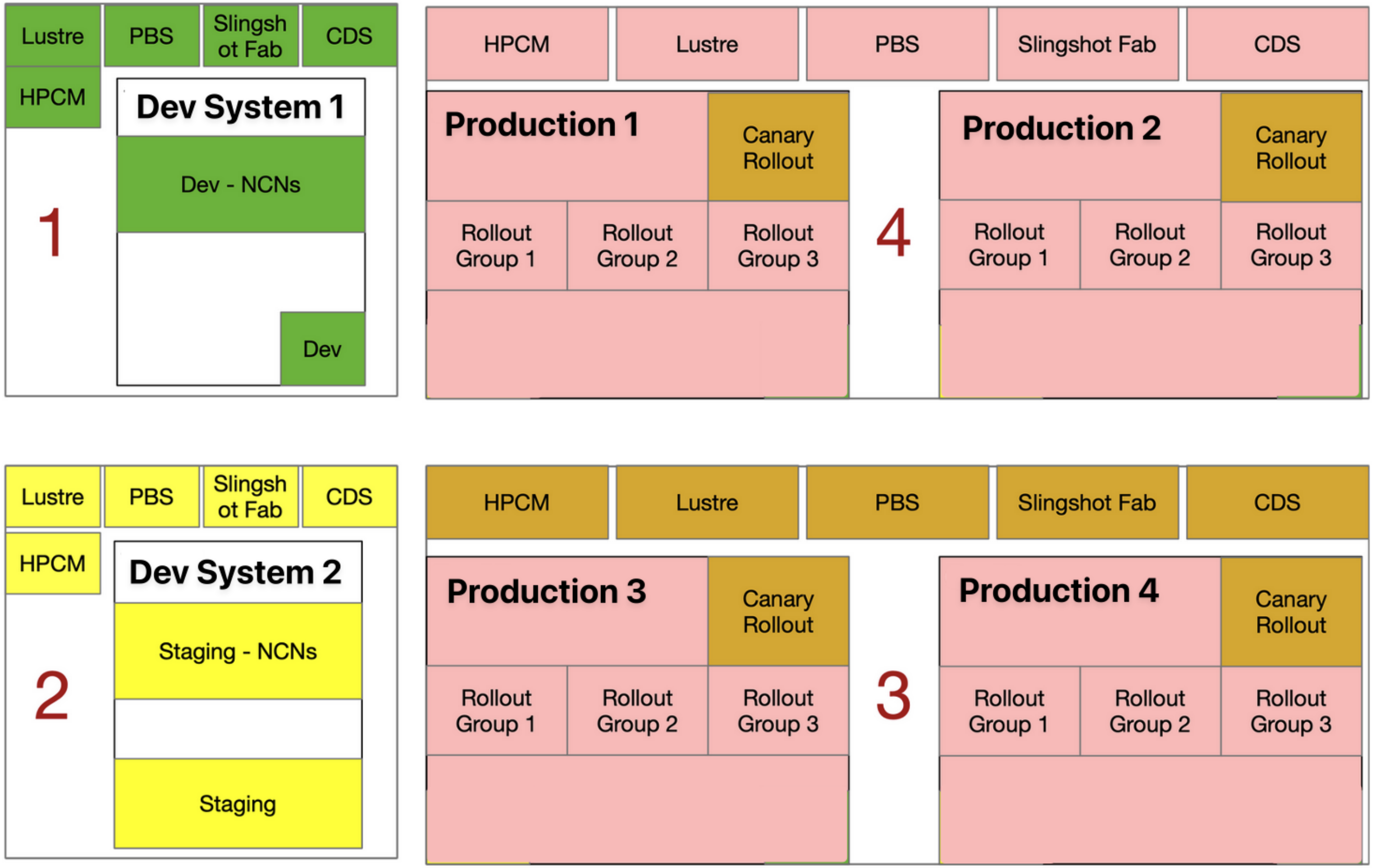
```
# Ansible playbook – vlan_300_on_port_21_leaf_1.yml
-  hosts: leafswitch_1
   collections:
    - arubanetworks.aos_switch
   tasks:
    - name: create vlan
      arubaoss_vlan:
       vlan_id: 300
       name: "vlan300"
       status: "VS_PORT_BASED"
       vlantype: "VT_STATIC"
       config: "create"
       command: config_vlan

    - name: assign vlan to port 21
      arubaoss_vlan:
       vlan_id: 300
       port_id: 21
       command: config_vlan_port
```

ansible-playbook -i canu-inventory vlan_300_on_port_21_leaf_1.yml

Case Study: Physical Topology

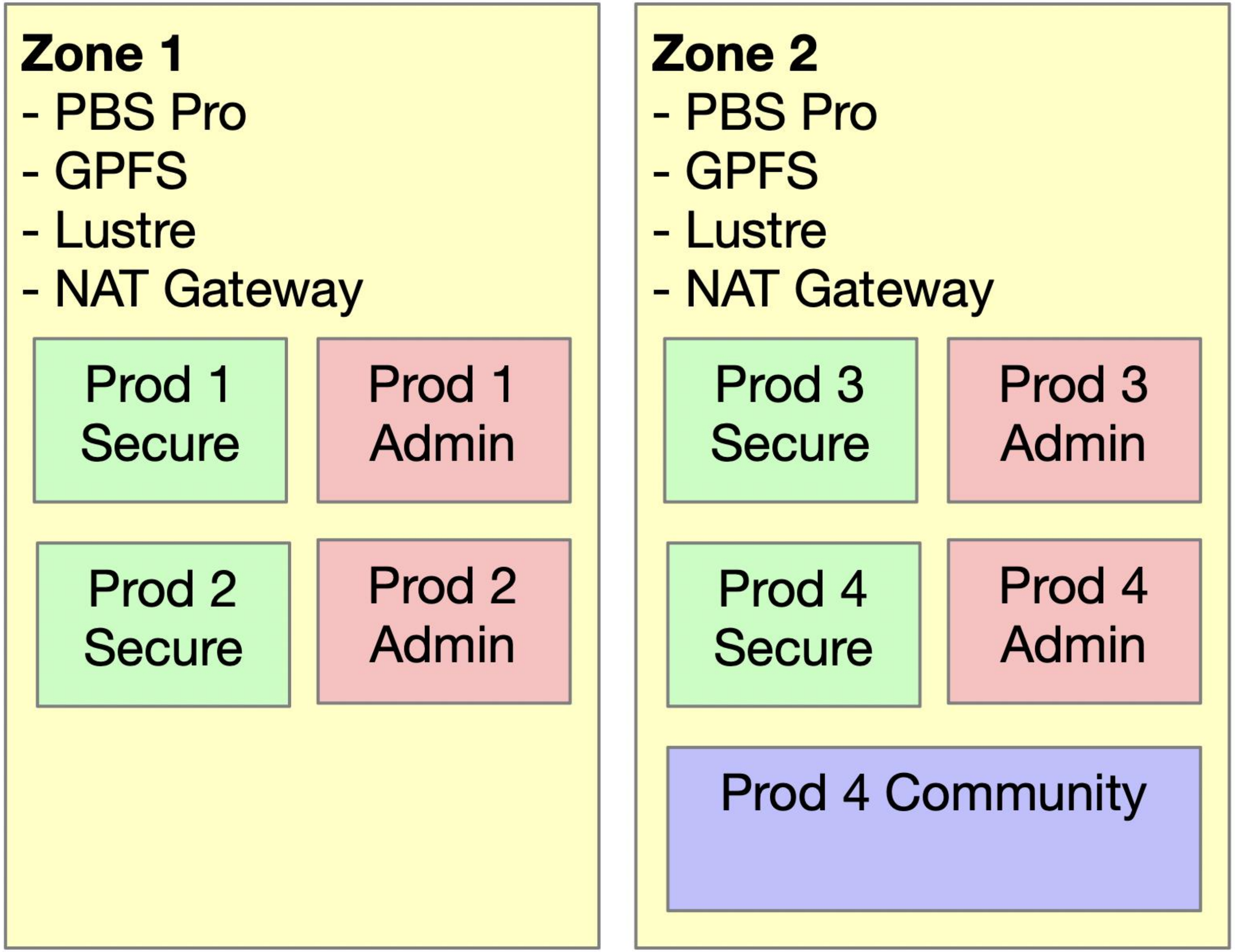
- Almost 11k nodes
- In Two data centers
- Two HPCM-managed Infrastructure "zones"
- Four CSM-managed compute node environments
- Two TDS systems with HPCM/CSM
- Hybrid-cloud infrastructure management



Case Study: Partitioning Scheme

- 2 Infrastructure Partition (HPCM)
- 4 Admin Partition (CSM)
- 4 Secure Usage Partition (CSM)
- 1 Community Partition (CSM)
- Every Partition Needs HSN and NMN isolation

Every partition has unique IP subnets, NAT gateways, and VLANs.



Zone 1
- PBS Pro
- GPFS
- Lustre
- NAT Gateway

Prod 1 Secure

Prod 1 Admin

Prod 2 Secure

Prod 2 Admin

Zone 2
- PBS Pro
- GPFS
- Lustre
- NAT Gateway

Prod 3 Secure

Prod 3 Admin

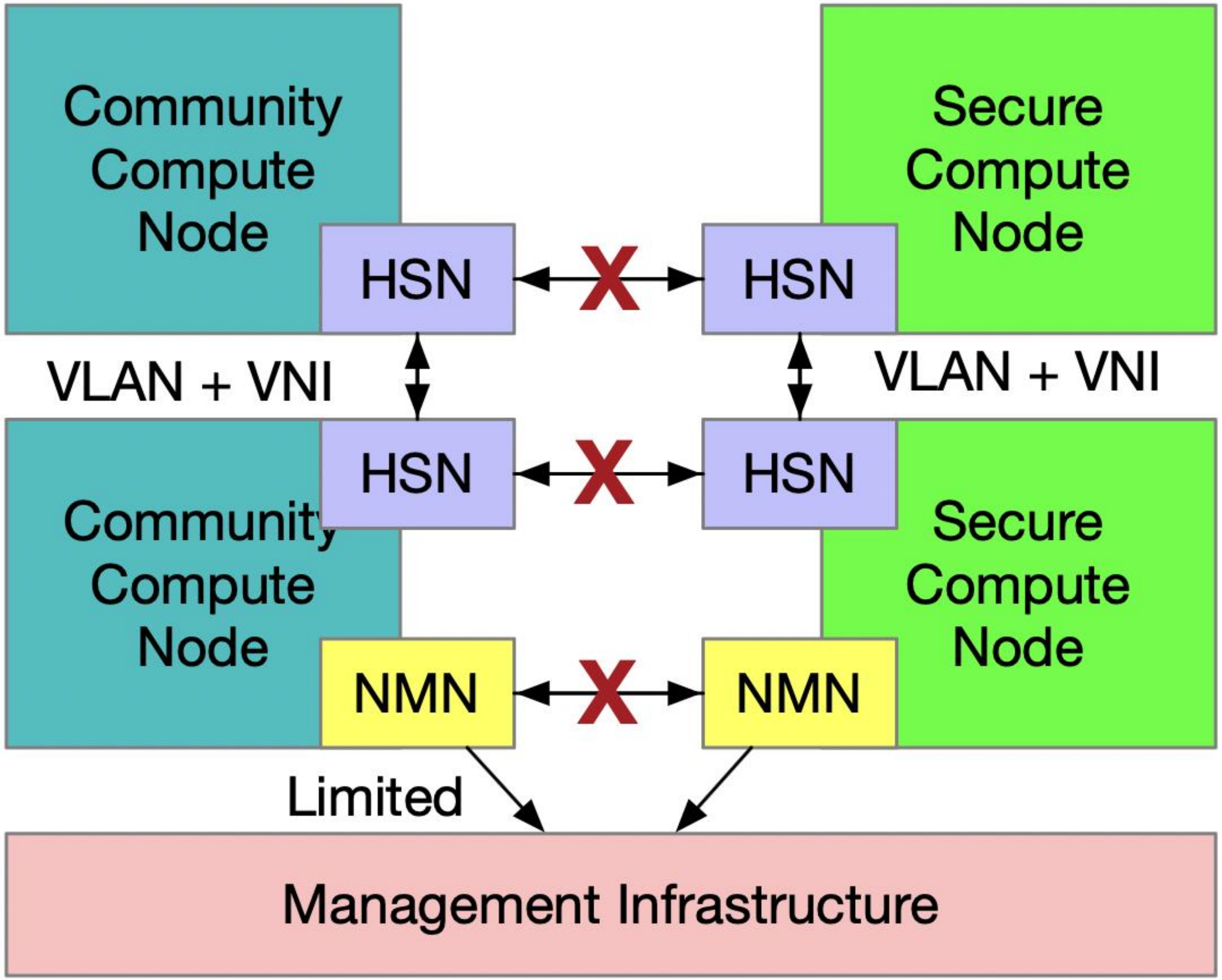
Prod 4 Secure

Prod 4 Admin

Prod 4 Community

Case Study:
Community Partition

- Least Security Clearance, Most Secured
- IPTables block all inter-compute traffic over NMN
- HSN Port Policies assign IP and VLAN
- HSN VNIs used in jobs



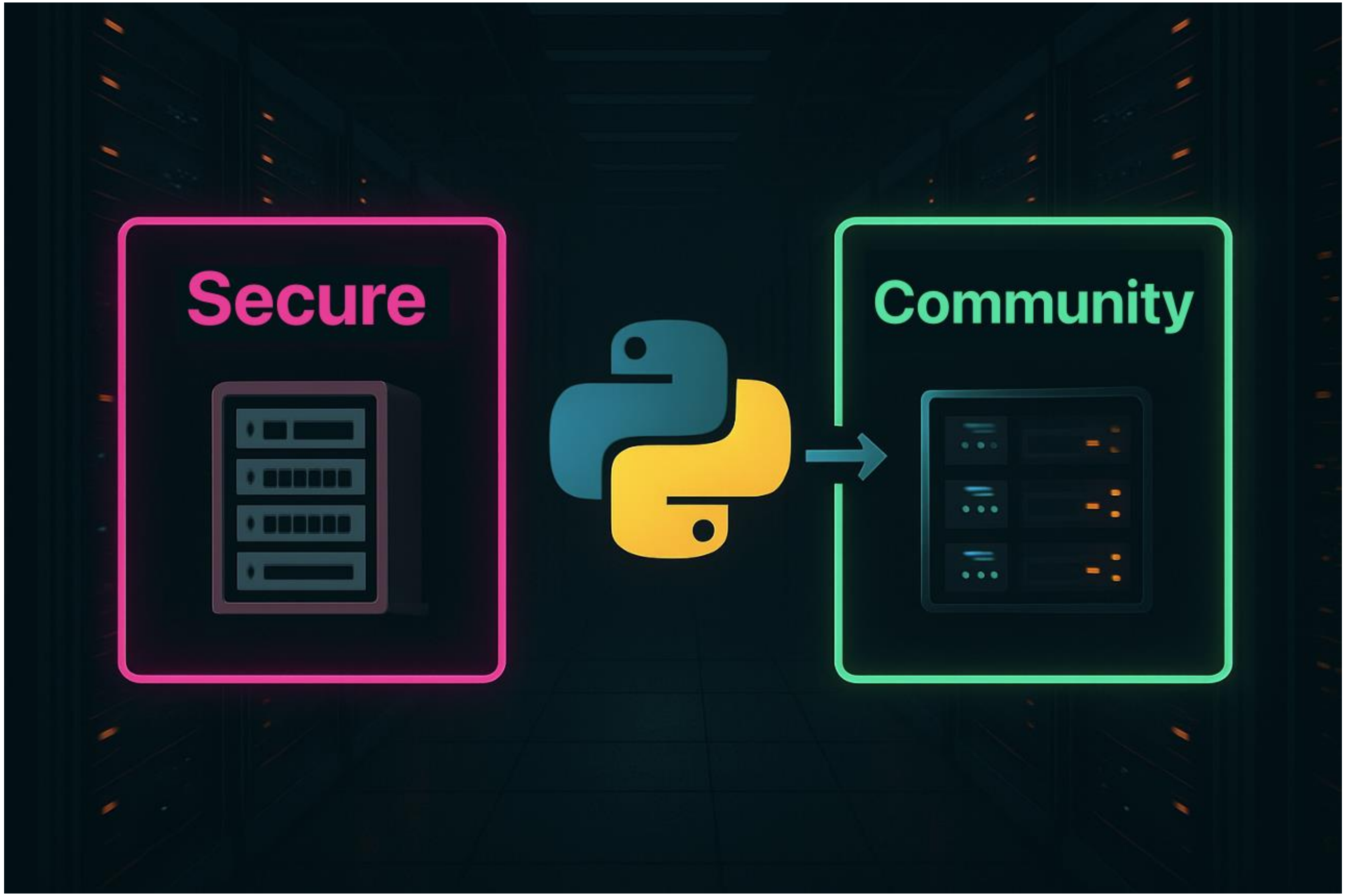
Case Study:
Configuration Management

A Python script moves nodes between Secure and Community partitions.

It:

- Powers down the node
- Updates HSM labels
- Updates IP address in SLS / Unbound DNS
- Updates HSN port policies to apply port VLANs and NIC IP
- Powers on the node

The script only needs two parameters, the zone and the xname(s). All subnets and vlans are stored in a repo for reference during execution.



Case Study: Results

Users and Workloads with varying clearance coexist

Resulting in

High Utilization / Higher ROI

Recap

- Slingshot offers network isolation via
  - VNIs (RDMA Tagging enforced at switch, NIC, and application),
    - Slingshot Network Operator (for simplified orchestration
  - VLANs (Ethernet Tagging enforced at switch and NIC)

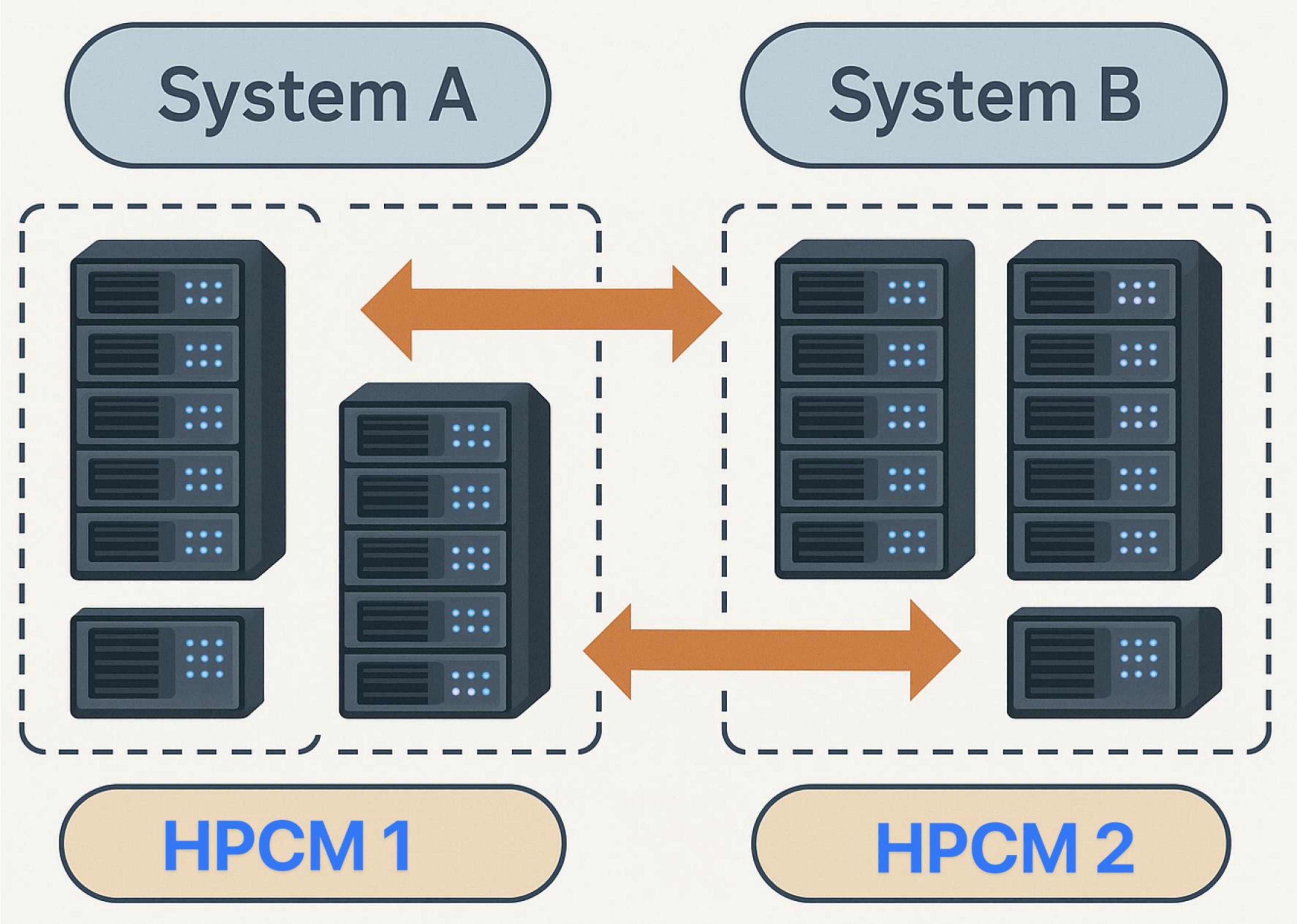
- Management Network Traffic is isolated via
  - VLANs, ACLs, Iptables + MAC

- Centralize your source-of-truth and orchestration for both high-speed and management networks even if you have many systems of different software.

Q & A

HPCM Partitioning Features

- No additional partitioning features out-of-the-box
- Extended with 3rd party tooling, e.g. Ansible, Bash, etc
- Slingshot Fabric Manager provides APIs for HSN partitioning
- Aruba switches provide REST APIs
- Physical partitioning across more environments is an option


System A
System B
HPCM 1
HPCM 2