State migration

Yingjie Gu
Melinda Shore
Senthil Sivakumar

AGENDA

- Problem definition
- Related efforts
- Way forward

Problem definition

Flow-associated State and Policy

- Flow-associated State (State, for short)
  - State is dynamic and learnt
  - State is created by traffic flows
  - No standardized definition for state
  - State is different for different middle boxes, vendors and applications
- Policy is provisioned and mostly static
- Policy is applied prior to creating state

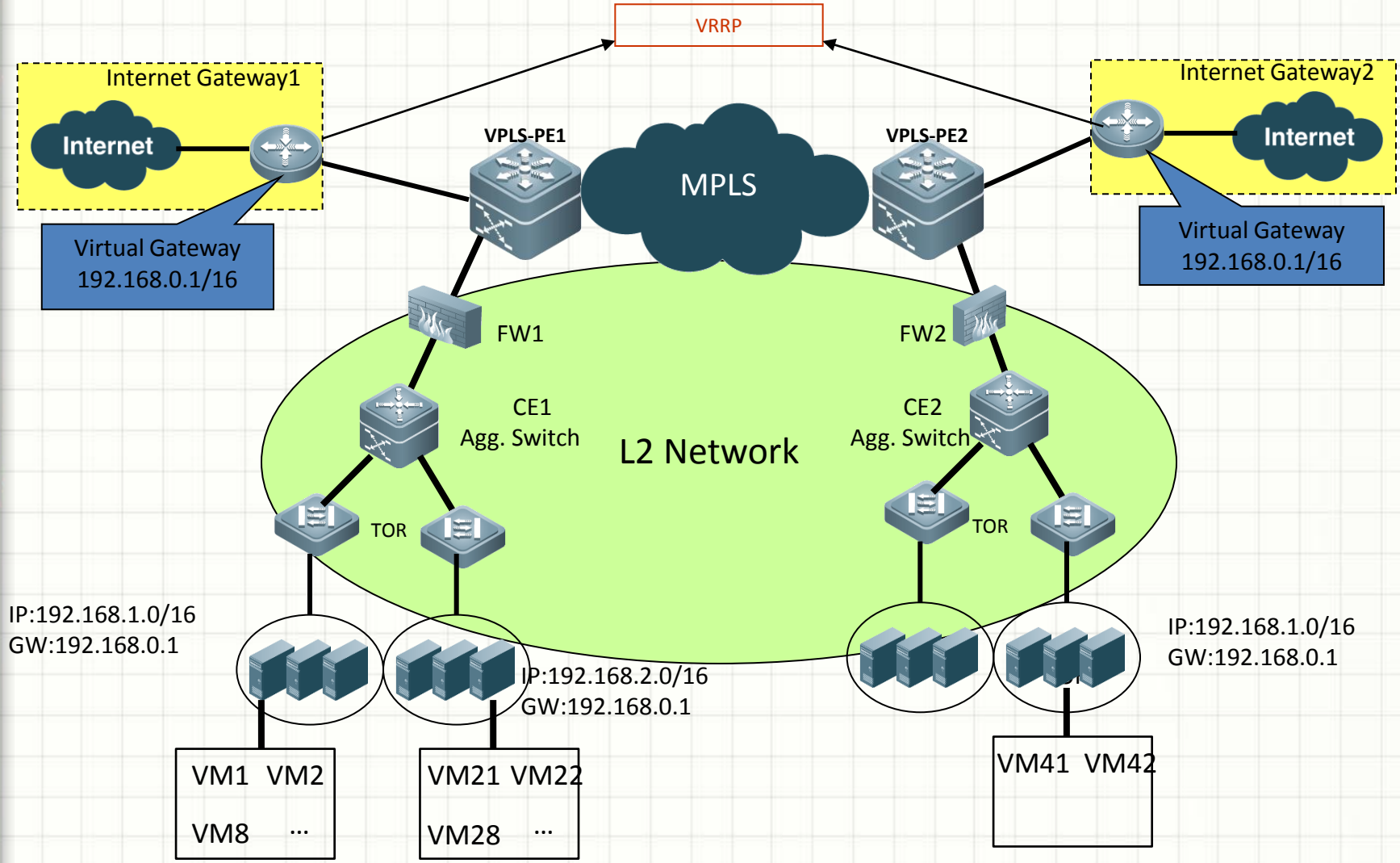
Why do we need to move state?

- End-to-end network flow typically traverses one or more "middlebox," which may retain state about the flow.
- When a point of attachment changes for an end-point, if the state is lost, the applications fail.

Real life use case scenarios:

- Triggered migration, including planned and unplanned:
  - Features:
    - Undefined destination
    - Trigger from devices other than Middlebox
  - An example: Virtual Machine migration

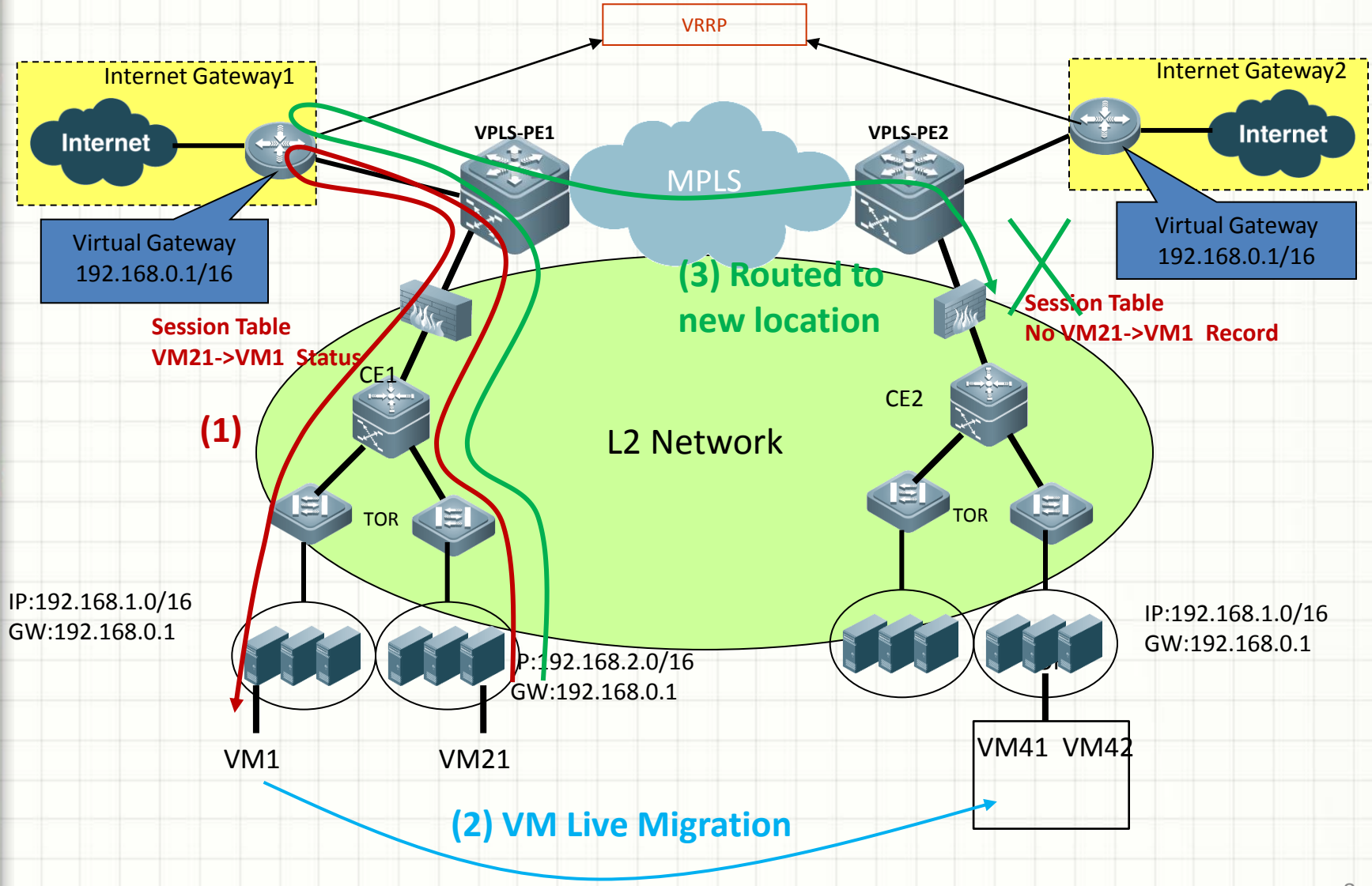
Network Architecture Example



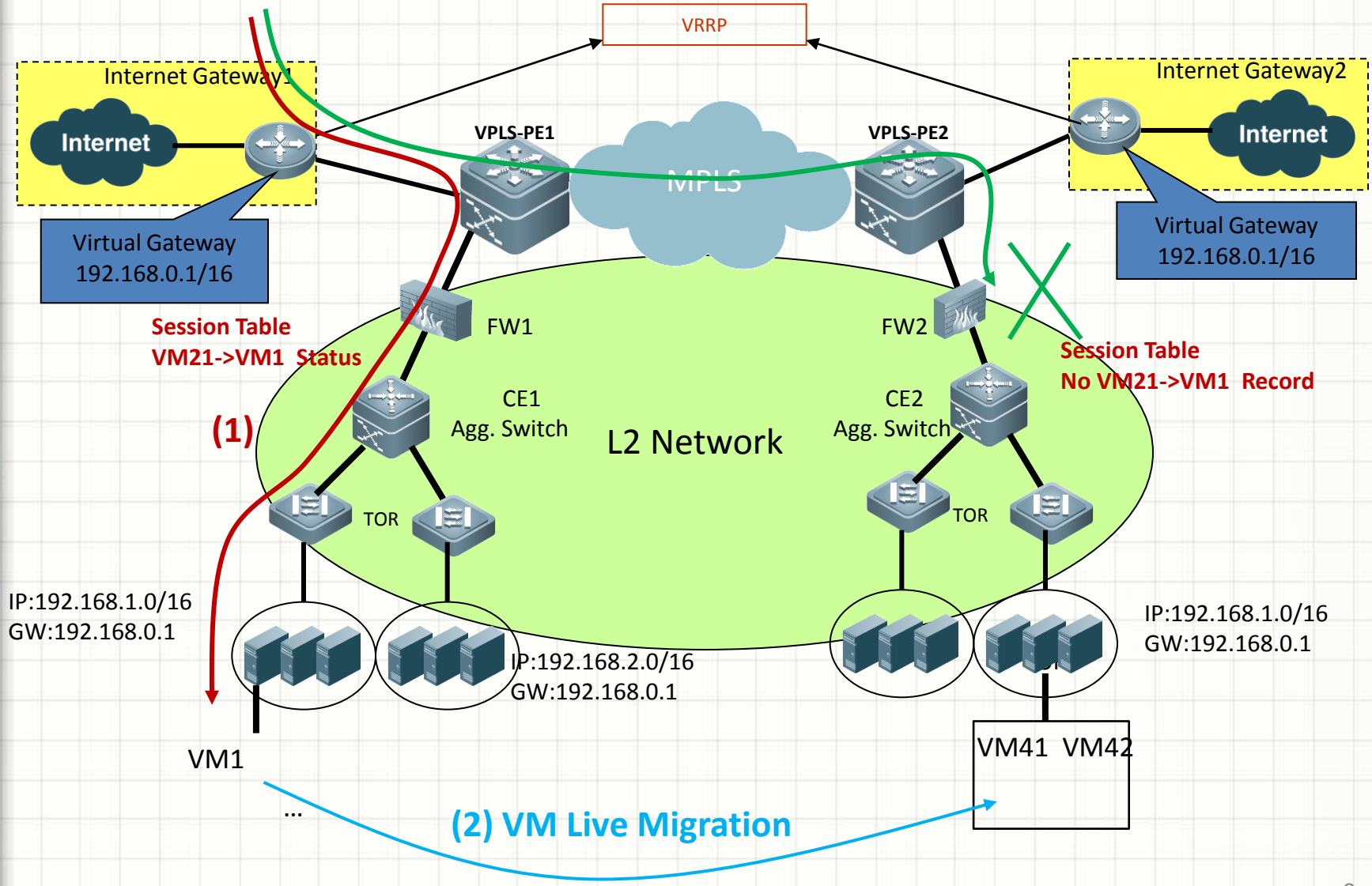
VM1 : 192.168.1.1

VM21 : 192.168.2.21

Use Case 1: Intra-communication



Use Case 2: Extra-communication



Virtualization and state migration

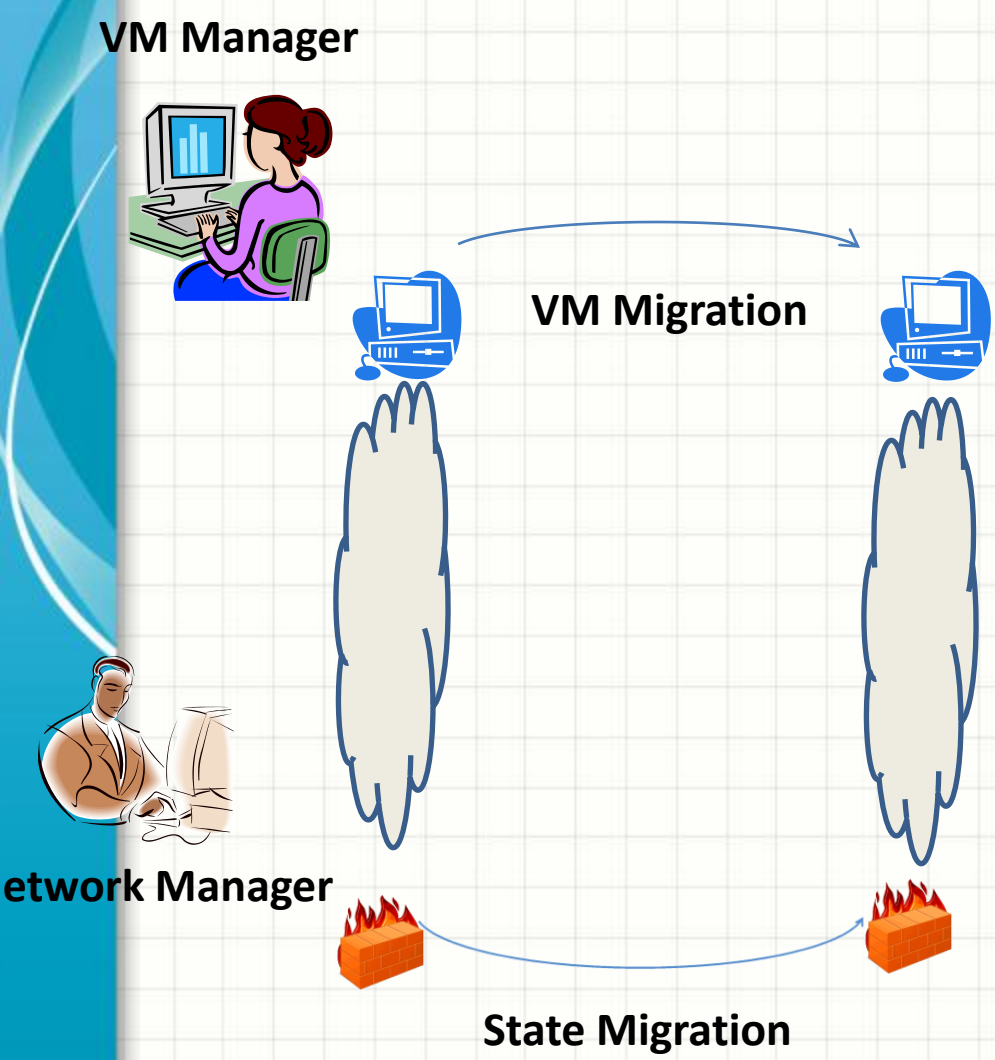
- When a virtual machine moves, the endpoint's attachment to network changes
- To make the virtual machine migration seamless in live networks, the state associated with VM should migrate.
- Middlebox flow state must be migrated when the VM migrates.

In a nutshell..

- Real life use case scenarios exist that requires triggered state migration
  - Virtual machine migration and Middlebox
- No standardized models exist

Problem Decomposing

Simplified Model



Steps to migrate state

- Recognizing when an endpoint has moved
- Locating middle boxes along the original path
- Locating middle boxes along the new path
- Getting a copy of state from middle boxes along the old path
- Installing that state in middle boxes along the new

Potential work

- Trigger protocol/interface:
  - Recognizing when an endpoint has moved
- Path discovery:
  - Locating middle boxes along the original or new path
- State copy:
  - Getting a copy of state from middle boxes along the old path
- Feedback protocol/interface:
  - Getting a copy of state from middle boxes along the old path

Potential work

- Trigger protocol/interface:
  - Recognizing when an endpoint has moved
- Path discovery:
  - Locating middle boxes along the original or new path
- State copy:
  - Getting a copy of state from middle boxes along the old path
- Feedback protocol/interface:
  - Getting a copy of state from middle boxes along the old path

Not all are necessary, it depends on what model we choose to solve the problem

```
        -----------              -----------
        | Upper   |----------->| Upper   |
        | Devices | 3.Transfer | Devices |
        -----------              -----------
4.        |  ^  2.
Feedback |  | Distribute
         |  | Migration
         V  | Notification
        -----------              -----------
        | Source  |----------->|  Dest.  |
        | Bridge  | 3.Transfer | Bridge  |
        -----------              -----------
         |  ^   1.
4.       |  | Migration
Feedback |  | Notification
         |  | VDP Associate
         V  |  (S-Bit=1)
        ----------               ----------
        |   VM1  |               |new VM1 |
        ----------               ----------
```

Straight Way

```
                                  1. Migration
          --------------------   Notification ------------
          |   COORDINATOR    |<-------------> |VM Manager|
          --------------------  4.Feedback    ------------
          /\  /\        \     \                     /\
  2. Get  /  /           \     \ 3. Set             |
Policies /  /             \     \   Policies        |
        /  /               V     \                  |
       / -----------   ----------- |                |
      |  | Source  |   | Dest.   | |                |
      |  | Upper   |   | Upper   | |                | bi-directional
      |  | Devices |   | Devices | |                |   API
      |  -----------   ----------- |                |
      |                            V                |
    ----------                 ----------          |
    | Source  |                |  Dest.  |          |
    | Bridge  |                | Bridge  |          |
    ----------                 ----------          |
        |                          |                |
     ---------                  ---------          |
     |  VM1  |                  |new VM1 | <----------
     ---------                  ---------
```

Out of band

```
                                  1. Migration
          --------------------------   Notification  ---------------
          |       COORDINATOR       |-----------------|  VM Manager |
          --------------------------- 6. Feedback     ---------------
           /\  /\          /\     /\                          /\
3. Upload DI/   /  3.       5.\      \ 5.Download DI           |
           /   /               \      \                        |
          /   /                 \      \                        |
         /   V 2.               2. V     \                      |
        / -----------          ----------- \                   |
       |  | Source  |          | Dest.   |  |                  |
       |  | Upper   |          | Upper   |  |                  |
       |  | Devices |          | Devices |  |                  |
       |  -----------          -----------  |                  |API
       |    |                          |    |                  |
       |    /                           \   |                  |
     2.V   /                           \  V 4. Migration       |
     ----------                   ----------  Notification     |
     | Source  |                  |  Dest.  |                  |
     | Bridge  |                  | Bridge  |                  |
     ----------                   ----------                   |
          |                          |                         |
          |                          |                         |
      ---------                  ---------                     |
      |  VM1  |                  |new VM1 |---------------------
      ---------                  ---------
```

Hybrid Way

Gap Analysis Summary

- There are a number of protocols (IETF and otherwise) for communicating with middlebox.
- Some are generalized to support multiple middlebox types; most are not
- Discovery remains largely unaddressed, and where it is addressed it is either too narrowly scoped (UPnP IGD) or unreliable (STUN)

Way forward

Way forward

- We would like the unaddressed problems be solved.
  - We believe State Migration (SAMI) is a typical IETF topic;

Find a place to do the potential work

- Trigger protocol/interface:
  - Recognizing when an endpoint has moved
- Path discovery:
  - Locating middle boxes along the original or new path
- State copy:
  - Getting a copy of state from middle boxes along the old path
- Feedback protocol/interface:
  - Getting a copy of state from middle boxes along the old path

Backup Slides- related work

SOCKS

- The IETF's first firewall traversal protocol
- RFC 1928
- firewall only
- set up tunnels between an endpoint and middlebox
- no discovery

RSIP

- RFC 3101
- NAT only
- set up tunnels between endpoint and NAT
- no discovery

midcom

- RFC 3303
- firewall *and* NAT
- specified signaling between an endpoint or its proxy and middlebox to request firewall pinholes and NAT mappings
- SNMP transport
  - everybody hates SNMP
- no discovery

nsis NAT/Firewall Signaling Layer

- RFC 5971
- end-to-end signaling messages (next-gen RSVP)
- no discovery of middleboxes not already on the path between endpoints

STUN

- RFC 5389
- NAT only
- used to discover existing NAT table mapping or create one via side-effect
- NAT discovery is a by-product of discovering endpoint's NATted address
  - note that it is possible that this would not be a control address (esp. since it's the external-facing address)

TURN

- RFC 5766
- NAT only
- establishes relay at external server
- bypasses NAT completely; no discovery

ICE

- RFC 5245
- NAT only
- not really a protocol
  - procedure describing use of STUN and TURN to discover set of candidate addresses for endpoint and then choose “best”

PCP

- Newish IETF working group
- firewall and NAT
- same basic communication model as midcom
- doesn’t use SNMP
  - (everybody hates SNMP)
- deployment context is carrier-grade NAT
- no discovery mechanism

UPnP IGD

- UPnP Forum specification
- NAT/firewall
  - (consumer-grade “router”)
- direct communication between endpoint and device
- discovery is limited to local link