SUPER CLOUD
H2020 PROJECT:

SuperCloud: The first demo is the hardest

Max Alaluna

User-centric management of security and dependability in clouds of clouds
User-centric management of security and dependability in clouds of clouds
User-centric management of security and dependability in clouds of clouds

“big switch” abstraction; software switching

flexible topology; (partially) sdn-enabled; software and hardware switching

28 January, 2016
Design requirements

Fulfilling the design requirements

Preliminary architecture

Demo: SUPERCLOUD network virtualization platform v0.1
Design requirements

Fulfilling the design requirements

Preliminary architecture

Demo: SUPERCLOUD network virtualization platform v0.1
Remote, flexible **control** of network elements

- the enabler

**Full network virtualization**

- topology abstraction; addressing abstraction; isolation

**Security and dependability**

- resilient data plane; resilient controller

**Autonomic security management**

- Data plane monitoring and proactive attack detection

**Resource snapshot and migration**

**Scalability** and performance

**Compatibility and interoperability**

- Heterogeneous clouds -> Common view
Design requirements

Fulfilling the design requirements

Preliminary architecture

Demo: SUPERCLOUD network virtualization platform v0.1
Remote, flexible control of network elements: **Open vSwitch**

- A software switch for virtualized environments
  - resides within the hypervisor or management domain
  - exports interface for fine grained control of the forwarding (via OpenFlow) and of configuration (via OVSDB: to configure queues, add/remove ports, etc.)

- **Open vSwitch support:**
  - Linux, FreeBSD, NetBSD, Windows, ESX
  - KVM, Xen, Docker, VirtualBox, Hyper-V, ...
  - OpenStack, CloudStack, OpenNebula, ...

- **Widely used:**
  - Most popular OpenStack networking backend
  - Default network stack in XenServer
Fulfilling design requirement #1: OvS
• Full network virtualization
  ✷ isolation: *logically-centralized control* allows *mapping* of virtual/physical events
  ✷ topology abstraction: *intercept* topology-related messages
  ✷ addressing abstraction: *tagging* packets (MPLS, VLAN) and/or *rewriting* L2/L3 headers, *tunneling*
Fulfilling design requirement #3: security and dependability

- **Resilient data plane**
  - resilient routing
  - multipath
  - network coding
  - (secure) path monitoring

- **Resilient controller**
  - Fault tolerant controller
  - With a focus on correctness
    - consistency of both data and control planes (and their interaction) to avoid network anomalies (loops, security breaches)
Fulfilling other design requirements

- Resource snapshot and migration
  - Migrate the VM and the network
  - VM state + network state (OvS forwarding rules)
    - Do it live if possible

- Scalability and performance
  - Distributed SDN controller
  - Fast algorithms for virtual/physical mapping, for network embedding
  - Storage+Networking solutions (e.g., deduplication techniques in SDN)

- Compatibility and interoperability
  - OASIS TOSCA: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca
    - Open standard that defines the interoperable description of services and applications hosted on the cloud
    - Allows expressing application requirements independently from cloud provider capabilities
    - Flexible movement between different clouds
  - A facilitator for network migration
Design requirements

Fulfilling the design requirements

Preliminary architecture

Demo: SUPERCLOUD network virtualization platform v0.1
User-centric management of security and dependability in clouds of clouds
Preliminary network virtualization architecture

User-centric management of security and dependability in clouds of clouds
Preliminary network virtualization architecture

User-centric management of security and dependability in clouds of clouds
Design requirements

Fulfilling the design requirements

Preliminary architecture

Demo: SUPERCLOUD network virtualization platform v0.1
A fatal exception 0E has occurred at 0028:C0011E36 in VXD UMM(01) + 00010E36. The current application will be terminated.

* Press any key to terminate the current application.
* Press CTRL+ALT+DEL again to restart your computer. You will lose any unsaved information in all your applications.

Press any key to continue
Instantiating the SUPERCLOUD provider network: 3 servers
Instantiating the SUPERCLOUD provider network: emulating 2 clouds

Cloud A

Server S1

Server SG1 (proxy)

Cloud B

Server S14 (proxy)

192.168.4.x/24

192.168.5.x/24
Instantiating the SUPERCLOUD provider network: virtualized

Cloud A

Server S1

Xen with OvS

Server SG1 (proxy)

Xen with OvS

Cloud B

Server S14 (proxy)

Xen with OvS

192.168.4.x/24

192.168.5.x/24
Instantiating the SUPERCLOUD provider network: with SDN control

- **Server S1**
- **Server SG1** (proxy)
- **Server S14** (proxy)

**Cloud A**
- **Xen with OvS**
  - IP range: 192.168.4.x/24

**Cloud B**
- **Xen with OvS**
  - IP range: 192.168.5.x/24

**NetHypervisor v0.1**
- SDN controller (Floodlight)
- OpenFlow & OVSDB over TCP
Instantiating the SUPERCLOUD user virtual network

Same IP range: 10.200.0.x/24

Cloud A
- Server S1
  - Xen with OvS

Cloud B
- Server S14
  - Xen with OvS

NetHypervisor v0.1
- SDN controller (Floodlight)

OpenFlow & OVSDB over TCP

192.168.4.x/24  192.168.5.x/24

User X

User Y
Instantiating the SUPERCLOUD user virtual network

Cloud A

Server S1

Xen with OvS

1 2 1 2

Server SG1 (proxy)

Xen with OvS

5 6 5 6

192.168.4.x/24

NetHypervisor v0.1

SDN controller (Floodlight)

Cloud B

Server S14 (proxy)

Xen with OvS

3 4 3 4

192.168.5.x/24

Same IP range: 10.200.0.x/24

Xen with OvS

Xen with OvS

Xen with OvS

User X

User Y

1 2 3 4 5 6

OpenFlow & OVSDB over TCP

6 5 4 3 2 1
Instantiating the SUPERCLOUD user virtual network

Server S1

Cloud A

Server SG1 (proxy)

Xen with OvS

1 2 1 2

5 6 5 6

Server S14 (proxy)

Xen with OvS

3 4 3 4

Same IP range: 10.200.0.x/24

Gre tunnel

192.168.4.x/24 192.168.5.x/24

OpenFlow & OVSDB over TCP

NetHypervisor v0.1

SDN controller (Floodlight)

Cloud B

User X

User Y

Server SG1

Xen with OvS

Xen with OvS

Xen with OvS

GRE tunnel

1 2 3 4 5 6

1 2 3 4 5 6
Instantiating the SUPERCLOUD provider network

Cloud A

1. Server S1
2. Server S1

Server SG1 (proxy)

3. Server S14
4. Server S14

SDN controller (Floodlight)

NetHypervisor v0.1

Same IP range: 10.200.0.x/24

Cloud B

User X

User Y

Xen with OvS

Xen with OvS

Xen with OvS

MAC: 00:01:00:00:00:x
MAC: 00:02:00:00:00:x

GRE tunnel

192.168.4.x/24
192.168.5.x/24

OpenFlow & OVSDB over TCP
User Y: Web server in host 10.200.0.4

Same IP range: 10.200.0.x/24

Cloud A

Server S1

Xen with OvS

Server SG1 (proxy)

Xen with OvS

GRE tunnel

192.168.4.x/24

GRE tunnel

192.168.5.x/24

Cloud B

Server S14 (proxy)

Xen with OvS

User X

User Y

MAC 00:01:00:00:00:x

MAC 00:02:00:00:00:x
User X: ping 1 -> 4

Same IP range: 10.200.0.x/24

Cloud A

Server S1

Server SG1 (proxy)

Cloud B

Server S14 (proxy)

Xen with OvS

Xen with OvS

GRE tunnel

192.168.4.x/24

192.168.5.x/24

User X

User Y

MAC 00:01:00:00:00:x
MAC 00:02:00:00:00:x

MAC 00:01:00:00:00:x
MAC 00:02:00:00:00:x
User X: ssh 1 -> 4

Cloud A

Server S1

Server SG1 (proxy)

Server S14 (proxy)

Cloud B

Same IP range: 10.200.0.x/24

User X

User Y

GRE tunnel

192.168.4.x/24

192.168.5.x/24

MAC: 00:01:00:00:00:x

MAC: 00:02:00:00:00:x
User X: wget 1 -> 4 (FAILURE == ISOLATION)

Same IP range: 10.200.0.x/24
Same IP range: 10.200.0.x/24

Server S1

Server SG1 (proxy)

Server S14 (proxy)

Xen with OvS

Xen with OvS

Xen with OvS

GRE tunnel

GRE tunnel

User X

User Y

1 2 5 6

2 6

1 2 5 6

Xen with OvS

00:01:00:00:00:x

00:02:00:00:00:x

192.168.4.x/24

192.168.5.x/24

User Y: ping 1->4
User Y: ssh 1->4

- Cloud A
  - Server S1
  - Server SG1 (proxy)
  - Xen with OvS
  - MAC 00:01:00:00:00:x
  - MAC 00:02:00:00:00:x
- Cloud B
  - Server S14 (proxy)
  - Xen with OvS
  - Same IP range: 10.200.0.x/24
  - MAC 00:01:00:00:00:x
  - MAC 00:02:00:00:00:x
- GRE tunnel
  - 192.168.4.x/24
  - 192.168.5.x/24
- User X
  - 1
  - 2
  - 3
  - 4
  - 5
  - 6
- User Y
  - 1
  - 2
  - 3
  - 4
  - 5
  - 6

OvS
User Y: `wget 1 -> 4`

Cloud A

- Server S1
  - Xen with OvS
  - 1 2 1 2
  - MAC

- Server SG1 (proxy)
  - Xen with OvS
  - 5 6 5 6
  - MAC

- GRE tunnel
  - 192.168.4.x/24

Cloud B

- Server S14 (proxy)
  - Xen with OvS
  - 3 4 3 4
  - MAC

- GRE tunnel
  - 192.168.5.x/24

Same IP range: 10.200.0.x/24

User X

- 1 2 5 6

User Y

- 1 6 2 5 3 4
"The project SUPERCLOUD has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 643964.”

If you need further information, please contact the coordinator:
TECHNIKON Forschungs- und Planungsgesellschaft mbH
Burgplatz 3a, 9500 Villach, AUSTRIA
Tel: +43 4242 233 55     Fax: +43 4242 233 55 77
E-Mail: coordination@supercloud-project.eu