Floodlight Tracing

Tulio A. Ribeiro
Fernando M. V. Ramos, Alysson Bessani

University of Lisbon, Faculty of Sciences.
LaSIGE - Large-Scale Informatic Systems Laboratory.
Portugal.

NavTalk 03/11/2016
• SDN – Context
• (Strong) Consistency Matters
• Proposed Controller Architecture
• Ravana x Floodlight Tracing
• Floodlight Pipeline Tracing
• Como Evitar Sobrecarga
• Ganho esperado
SDN - Context

Router (RIP, OSPF...)

Control Plane

Data Plane

Traditional Network Device.
Control Plane

Data Plane

Router (RIP, OSPF...)

Physically together.

Traditional Network Device.
SDN - Context

Separation of concerns.

Control Plane

Data Plane

Traditional Network Device.

SDN Controller (Control Plane) - OpenFlow Protocol

Network Application - API

Network Services

OpenFlow API

Data Plane

Network Device

OF-Sw

Software Defined Networking
SDN - Context

Application Layer
Traffic Engineering.

Control Layer
Device Manager, Routing.

Infrastructure Layer
Forwarding.

SDN Controller (Control Plane)

- Network Application
  - API
- Network Services
  - API
- OpenFlow Protocol

OpenFlow API
- Data Plane
- Network Device

OF-Sw

Software Defined Networking
SDN - Context

SDN Controller
(Control Plane)

Network Application

Network Application

OF-Sw

OF-Sw

OF-Sw

OF-Sw

OF-Sw

OF-Sw
It does not scale.
&
Single point of failure.
SDN - Context

```
SDN Controller (Control Plane)  SDN Controller (Control Plane)
| Network Services              | Network Services |
```

OF-Sw  OF-Sw  OF-Sw  OF-Sw  OF-Sw  OF-Sw

Scales better, but...
SDN - Context

Network Application

SDN Controller (Control Plane)

Network Services

. . .

SDN Controller (Control Plane)

Network Services

Is really scalable?

OF-Sw OF-Sw OF-Sw . . . OF-Sw OF-Sw OF-Sw

Synchronized?, != network view?, eventually consistency?, consensus algorithms?
SDN - Context

- SDN Controller (Control Plane)
- Network Services

SDN Controller (Control Plane)

Not Replicated, Single Point of Failure.

Overhead

Data Store

OF-Sw    OF-Sw    OF-Sw

. . .

OF-Sw    OF-Sw    OF-Sw
SDN Controller (Control Plane)  
Network Services

SDN Controller (Control Plane)  
Network Services

Data Store

Replicated and FT

OF-Sw  OF-Sw  OF-Sw  ...  OF-Sw  OF-Sw  OF-Sw

Eventually consistency scales out but problems might appear
Outline

- SDN – Context
- (Strong) Consistency Matters
- Proposed Controller Architecture
- Ravana x Floodlight Tracing
- Floodlight Pipeline Tracing
- Como Evitar Sobrecarga
- Ganho esperado
(Strong) Consistency Matters

• Why?
(Strong) Consistency Matters

Initial flow / path

Data Store

Eventually
(Strong) Consistency Matters

Link between S3-S4 is becoming congested.
(Strong) Consistency Matters

C2 writes changes into Data Store (network view).

Initial flow / path
(Strong) Consistency Matters

An eventually data store will reply as soon as it processes the request.
(Strong) Consistency Matters

C2 will immediately install the new rules at S3.
(Strong) Consistency Matters

- Consistency loop scenario.

This will create a transient loop.
(Strong) Consistency Matters

The network anomaly will eventually be corrected when the network state in the data store converges.
(Strong) Consistency Matters

- Consequences:
  - Hiccup in VoIP call.
  - Lost server connection.
  - Security breaches.
Outline

- SDN – Context
- (Strong) Consistency Matters
- Proposed Controller Architecture
- Ravana x Floodlight Tracing
- Floodlight Pipeline Tracing
- Como Evitar Sobrecarga
- Ganho esperado
Controllers of different network domains coordinate their actions using a logically centralized (consistent and fault-tolerant) data store.

- Fault Tolerant.
- Strong Consistency.
- Bottleneck is the data store.
Controllers of different network domains coordinate their actions using a logically centralized (consistent and fault-tolerant) data store.

- Fault Tolerant.
- Strong Consistency.
- Bottleneck is the data store.

**Proposed Controller Architecture**

**Data Store Overhead.**
Outline

- SDN – Context
- (Strong) Consistency Matters
- Proposed Controller Architecture
- Ravana x Floodlight Tracing
- Floodlight Pipeline Tracing
- Como Evitar Sobrecarga
- Ganho esperado
Figure 4: Steps for processing a packet in Ravana.
Outline

• SDN – Context
• (Strong) Consistency Matters
• Proposed Controller Architecture
• Ravana x Floodlight Tracing
• Floodlight Pipeline Tracing
• Como Evitar Sobrecarga
• Ganho esperado
**Floodlight Pipeline Tracing**

Worker Threads = 1

Switch specific buffer. Writer buffer.

sw 1, 2 and 3
Floodlight Pipeline Tracing

Worker Threads = 1

I/O R/W Threads

DM
LB
TM
...

msg

sw 1, 2 and 3

Switch specific buffer. Writer buffer.

Definição Estrutura de Dados

Sem tracing

```
ConcurrentHashMap<Long, Device> deviceMap; //original
```

Com tracing

```
ConcurrentHashMapTracing<Long, Device> deviceMap;
```

Inicialização Estrutura de Dados

Sem tracing

```
deviceMap = new ConcurrentHashMap<Long, Device>(); //original
```

Com tracing

```
/**
 * @param 1: Data structure (HashMap, ArrayList...)
 * @param 2: Unique name on system (there is no conflict resolution).
 * @param 3: Scope: GLOBAL / LOCAL.
 */

deviceMap = tarService.createConcurrentHashMapTracing(new
ConcurrentHashMap<Long, Device>(), "deviceMap", Scope.GLOBAL);
```
Floodlight Pipeline Tracing

Worker Threads = 1

I/O R/W Threads

sw1, sw2, sw3

serialized

HashMapTracing

DM

LB

TM

32

71

Worker Threads = 1

Switch specific buffer. Writer buffer.

get(AF)

put(AF, ...)
### Floodlight Pipeline Tracing

**Worker Threads = 1**

- sw1, sw2, sw3 (Switch specific buffer. Writer buffer.)
- I/O R/W Threads
- DM, LB, TM

**Hash Map Tracing**
- get(AF)
- put(AF, ...)

**Floodlight Pipeline Tracing**
- Module: devicemanager
- Module: linkdiscovery
- Module: topology
- Module: forwarding
  - xId: 97 Type: FLOW_MOD
  - xId: 98 Type: FLOW_MOD
  - xId: 99 Type: FLOW_MOD
  - xId: 100 Type: FLOW_MOD
  - xId: 101 Type: PACKET_OUT
- Module: loadbalancer

**MSG BUFFERED Per Module and Sw:**

- SW n Module: devicemanager
- SW n Module: linkdiscovery
- SW n Module: topology
- SW n Module: forwarding
  - xId: 97 Type: FLOW_MOD
  - xId: 98 Type: FLOW_MOD
  - xId: 99 Type: FLOW_MOD
  - xId: 100 Type: FLOW_MOD
  - xId: 101 Type: PACKET_OUT
- Module: loadbalancer
Outline

- SDN – Context
- (Strong) Consistency Matters
- Proposed Controller Architecture
- Ravana x Floodlight Tracing
- Floodlight Pipeline Tracing
- Como Evitar Sobrecarga
- Ganho esperado
Como Evitar Sobrecarga

Ping: host 1 → host 2

Memory Changes Per PacketIn/Switch: 00:00:00:00:00:00:00:01, ThreadID: 62:
# DM:deviceMap
---: put(K,V)
---: 6
---: Device [deviceKey=6, entityClass=DefaultEntityClass,
MAC=00:00:00:00:01:10, IPv4s=[], IPv6s=[], APs=[SwitchPort [switchDPI=00:00:00:00:00:00:00:01, port=1, errorStatus=null]]]
---: LOCAL
---: 0

DISPATCHING MESSAGE BUFFER threadId: 62
Module: devicemanager
Module: linkdiscovery
Module: topology
Module: forwarding
  xId: 157 Type: PACKET_OUT
Module: loadbalancer

Memory Changes Per PacketIn/Switch: 00:00:00:00:00:00:00:02, ThreadID: 62:
# DM:deviceMap
---: get(K)
---: 6
---: LOCAL
---: 0

DISPATCHING MESSAGE BUFFER threadId: 62
Module: devicemanager
Module: linkdiscovery
Module: topology
Module: forwarding
  xId: 158 Type: PACKET_OUT
Module: loadbalancer
Como Evitar Sobrecarga

Retorno do ping: host 2 → host 1

Memory Changes Per PacketIn/Switch: 00:00:00:00:00:00:00:02, ThreadID: 62:
# DM:deviceMap
---: put(K,V)
---: 7
---: Device [deviceKey=7, entityClass=DefaultEntityClass, MAC=00:00:00:00:02:20, IPv4s=[], IPv6s=[], APs=[SwitchPort
[switchDPID=00:00:00:00:00:00:00:02, port=1, errorStatus=null]]]
---: LOCAL
---: 0
# DM:deviceMap
---: get(K)
---: 6
---: LOCAL
---: 0

DISPATCHING MESSAGE BUFFER threadId: 62
Module: devicemanager
Module: linkdiscovery
Module: topology
Module: forwarding
  xId: 160 Type: FLOW_MOD
  xId: 161 Type: FLOW_MOD
  xId: 162 Type: PACKET_OUT
Module: loadbalancer
Como Evitar Sobrecarga

Segundo ping: host 1 → host 2

Memory Changes Per PacketIn/Switch: 00:00:00:00:00:00:00:01, ThreadID: 62:
# DM:deviceMap
---: get(K)
---: 6
---: LOCAL
---: 0
# DM:deviceMap
---: get(K)
---: 7
---: LOCAL
---: 0

DISPATCHING MESSAGE BUFFER threadId: 62
Module: devicemanager
Module: linkdiscovery
Module: topology
Module: forwarding
    xId: 639 Type: FLOW_MOD
    xId: 640 Type: FLOW_MOD
    xId: 641 Type: PACKET_OUT
Module: loadbalancer

Memory Changes Per PacketIn/Switch: 00:00:00:00:00:00:00:02, ThreadID: 62:
# DM:deviceMap
---: get(K)
---: 6
---: LOCAL
---: 0
# DM:deviceMap
---: get(K)
---: 7
---: LOCAL
---: 0

DISPATCHING MESSAGE BUFFER threadId: 62
Module: devicemanager
Module: linkdiscovery
Module: topology
Module: forwarding
    xId: 642 Type: PACKET_OUT
Module: loadbalancer
Outline

• SDN – Context
• (Strong) Consistency Matters
• Proposed Controller Architecture
• Ravana x Floodlight Tracing
• Floodlight Pipeline Tracing
• Como Evitar Sobrecarga
• Ganho esperado
Ganho esperado

Data Store throughput: 100 Kops/s
Controlador throughput: 100 Kops/s
Obrigado.


Questões / Dúvidas?

FIM