

# グループポリシーを用いた ネットワーク設定 @ODL HackFest Tokyo

2014年10月29日

株式会社富士通研究所

中川 幸洋、李 忠翰

- はじめに
- グループポリシー(GBP, Group Based Policy)
- GBPを用いたネットワーク設定の実行例

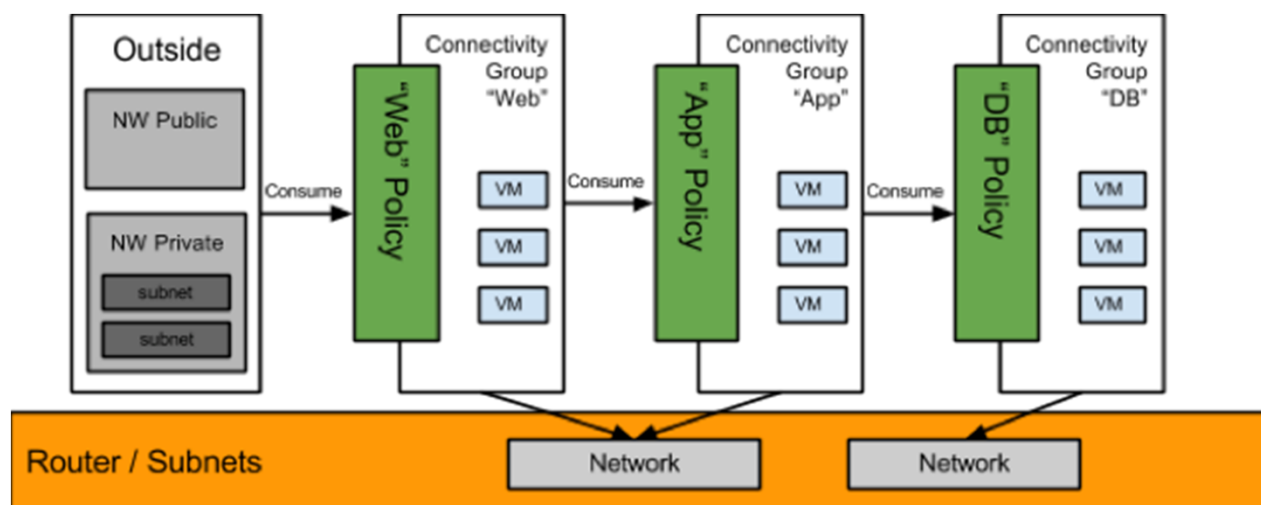
- Heliumで新たに導入された、グループポリシー (GBP)をトラフィック制御に用いる実行例をご紹介します。
- GBPバンドルに手を加えて使ってみたという程度ですが、これからODLを使う人には参考になるかと思っています。

## ■ Endpoint をグループ化し、Endpoint Group 間の接続性を記述した Policy でネットワークを抽象化

- ポリシーやインフラ変更時に、人間が介在することなくオーケストレーションを自動化

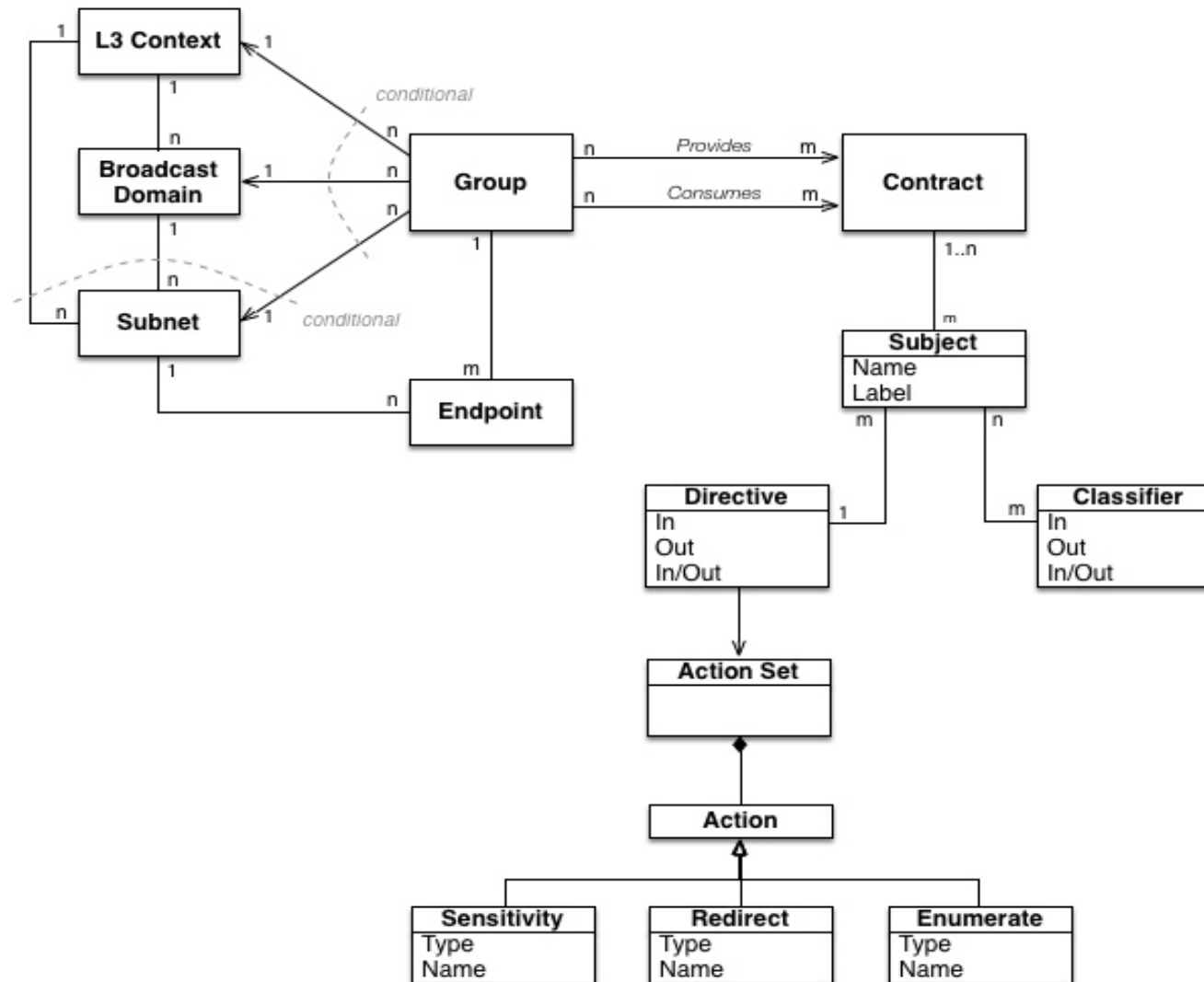
## ■ アプリ管理者とインフラ管理者の関心事を分離

- アプリの管理者は、ネットワークやルータなど意識することなく、上位の抽象化レベルを扱える
- インフラ管理者は、アプリのポート開閉など気にすることなく、インフラのポリシーを扱え、使用技術（VLANやVXLAN）を選択できる。アプリ管理者に意識させることなく、接続を制限できる（例えば、アプリ間のすべてのトラフィックを検査可能）

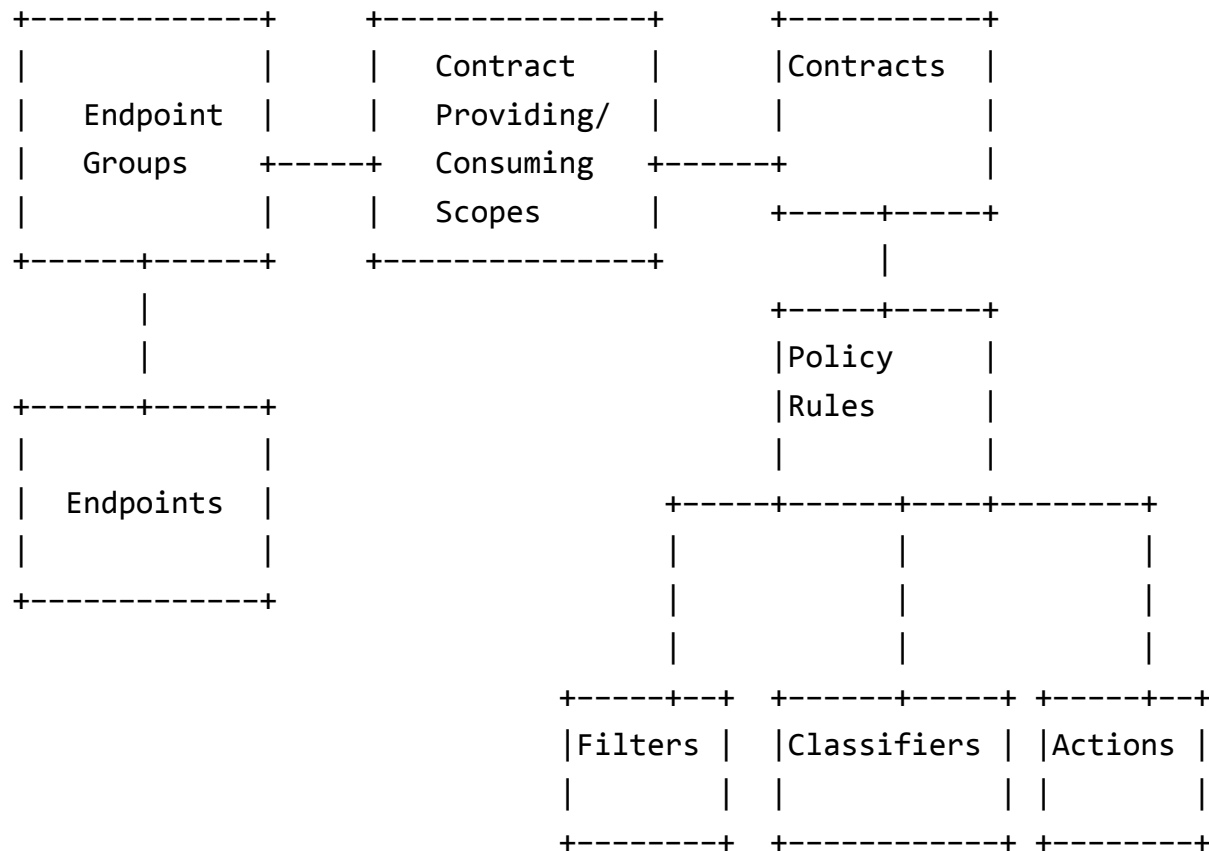


# Policyのモデル (OpenDaylight)

## ■ OpenDaylight GBP Pluginプロジェクトのモデル



# Policyのモデル (OpenStack)



## Classifier

- \* protocol - enum: TCP, IP, ICMP
- \* port\_range - single port number or range (as used in FWaaS firewall\_rule)
- \* direction - enum: IN, OUT, BI

## Action

- \* type - enum: ALLOW, REDIRECT, QOS, LOG, MARK, COPY
- \* value - uuid of a resource that performs the action, for example in the case of REDIRECT, its the uuid of the ServiceWrapper

## ■ このデモでは

- IPオーバーレイでElephant flowを優先制御するためにポリシーを定義し、ODLの最新リリースHeliumを用いて、GBPのレンダラーで物理スイッチが提供するトラフィック制御に合わせて仮想スイッチを設定します。

# GBPを用いたネットワーク設定の実行例



# データセンターのトラフィックの特徴

## ■ データセンターのトラフィックの分類

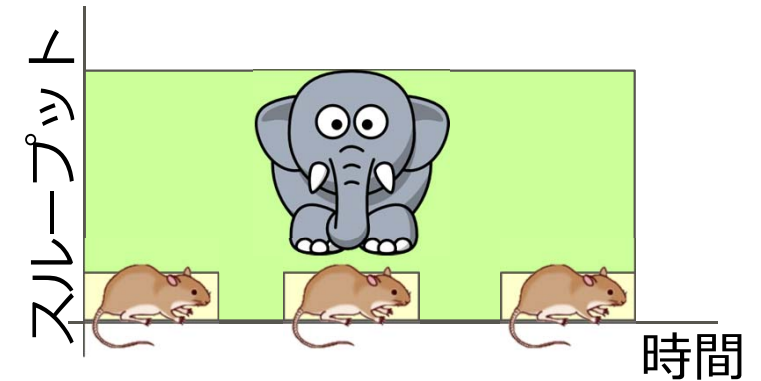
### ■ 大容量のデータ通信

■ Elephant : スループットが重要



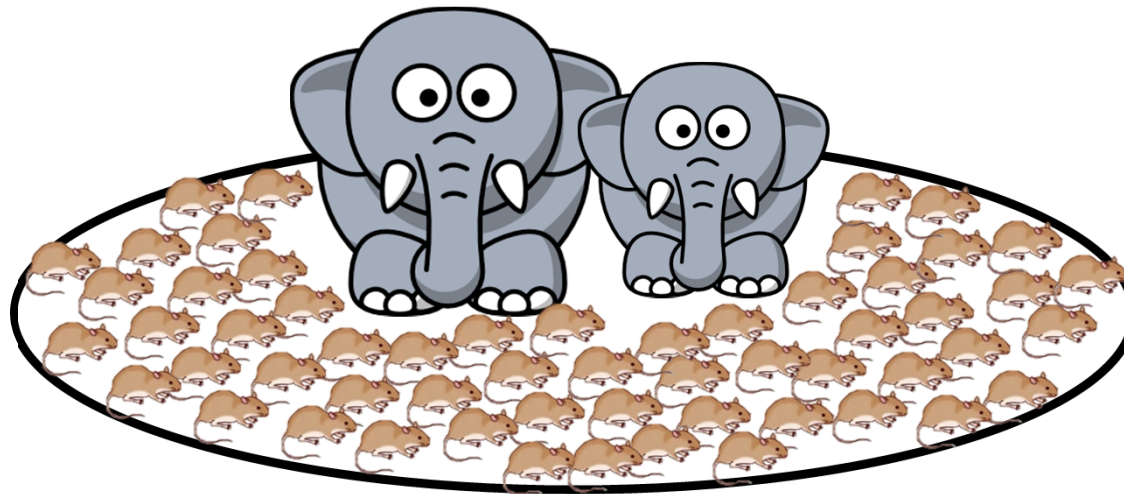
### ■ 細かい制御メッセージやその応答

■ Mice : 遅延が重要



<スループットの比較>

## ■ データセンターのトラフィックの構成<sup>1)</sup>



多数のMice(90%)と少数のElephant(10%)で構成

1) The Nature of Data Center traffic: Measurements & Analysis, In IMC, 2009

## ■ねらい

- IPオーバーレイでElephantを優先制御したい

## ■実現方法

- Elephantを優先制御するポリシーを作成

- 次頁参照

- Renderingする対象としてVXLANによるオーバーレイを想定

- ポリシーに基づいてDSCPを設定するようにOVSを設定

- 内部ヘッダのDSCPマーキングはVXLAN外部ヘッダにコピー

- DSCPに基づいてキューを分ける物理ネットワークを予め用意

## ■ 公開Demo<sup>\*</sup>に対する差分を次ページ以降で説明

\* [https://wiki.opendaylight.org/view/Group\\_Policy:Building\\_and\\_Running](https://wiki.opendaylight.org/view/Group_Policy:Building_and_Running)

# Elephantを優先制御するポリシー(1)

## ■ 全てのTCPに対して、direction in/outを許可

```
"contract": [
  {
    "clause": [
      {
        "name": "allow-tcp-clause",
        "subject-refs": [
          "allow-tcp-subject",
          "allow-icmp-subject"
        ]
      }
    ],
    "id": "22282cca-9a13-4d0c-a67e-a933ebb0b0ae",
    "subject": [
      {
        "name": "allow-tcp-subject",
        "rule": [
          {
            "classifier-ref": [
              {
                "direction": "in",
                "name": "tcp-src"
              },
              {
                "direction": "out",
                "name": "tcp-src"
              }
            ]
          }
        ]
      },
      {
        "name": "allow-icmp-subject",
        "rule": [
          {
            "classifier-ref": [
              {
                "name": "icmp"
              }
            ]
          }
        ],
        "name": "allow-icmp-rule"
      }
    ]
  },
  {
    "name": "allow-tcp-rule",
    "rule": [
      {
        "direction": "out",
        "name": "tcp-dest"
      },
      {
        "direction": "in",
        "name": "tcp-dest"
      }
    ]
  }
]

```

赤色:修正部分

# Elephantを優先制御するポリシー(2)

## ■ 全てのポートをOpenして、DSCPを20に設定

```
"subject-feature-instances": {
  "classifier-instance": [
    {
      "classifier-definition-id":
        "4250ab32-e8b8-445a-aebb-e1bd2cdd291f",
      "name": "tcp-dest",
      "parameter-value": [
        {
          "name": "type",
          "string-value": "TCP"
        },
        {
          "name": "destport",
          "string-value": "ALL"
        }
      ],
      {
        "int-value": "20",
        "name": "modified_dscp"
      }
    }
  ],
},
```

DSCPの  
値として  
20を設定

```
{
  "classifier-definition-id":
    "4250ab32-e8b8-445a-aebb-e1bd2cdd291f",
  "name": "tcp-src",
  "parameter-value": [
    {
      "name": "type",
      "string-value": "TCP"
    },
    {
      "name": "sourceport",
      "string-value": "ALL"
    }
  ],
  {
    "int-value": "20",
    "name": "modified_dscp"
  }
},
```

DSCPの  
値として  
20を設定

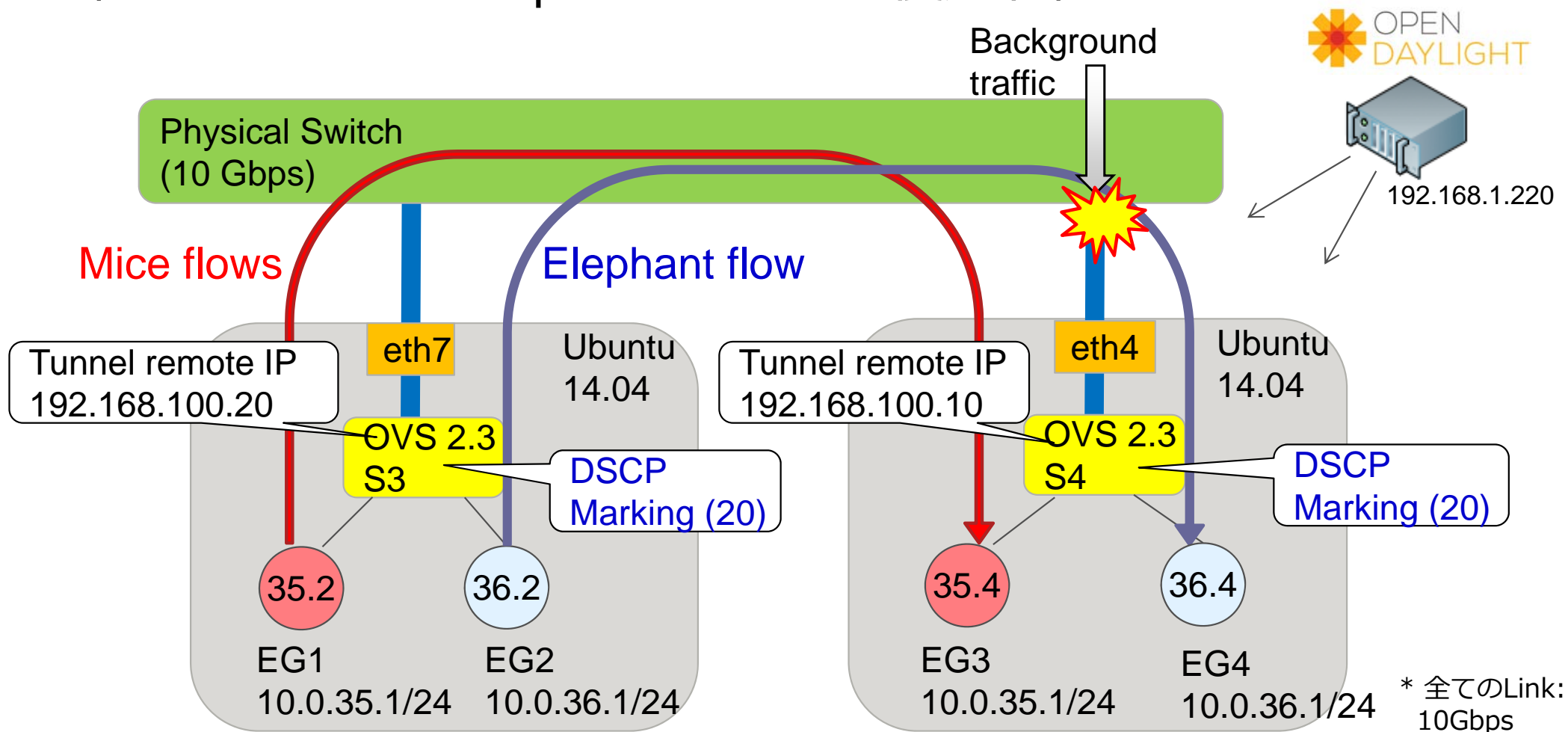
赤色:修正部分

## ■ Icmpの部分はデモと同じ

```
{
  "classifier-definition-id": "79c6fdb2-1e1a-4832-af57-c65baf5c2335",
  "name": "icmp",
  "parameter-value": [
    {
      "int-value": "1",
      "name": "proto"
    }
  ]
}
```

# GBP+Elephant デモ構成

- VXLANによるGBPデモをベースとしてレンダーラを試作
  - ポリシーに基づいて、OVSでDSCPをマーキング
  - 内部ヘッダのDSCPマーキングは、VXLANの外部ヘッダにコピー
- 物理スイッチでElephantフローを優先制御



## ■ データセンターのWorkloadを想定し、スループット測定

### ■ Workloadの構成

- Elephant (TCP): 100 MBytes (MTU : 8500 byte)
- Mice (TCP): 256 KBytes \* 10個 (MTU : 1500 byte)
- 固定のBackground traffic (UDP): 5 Gbps

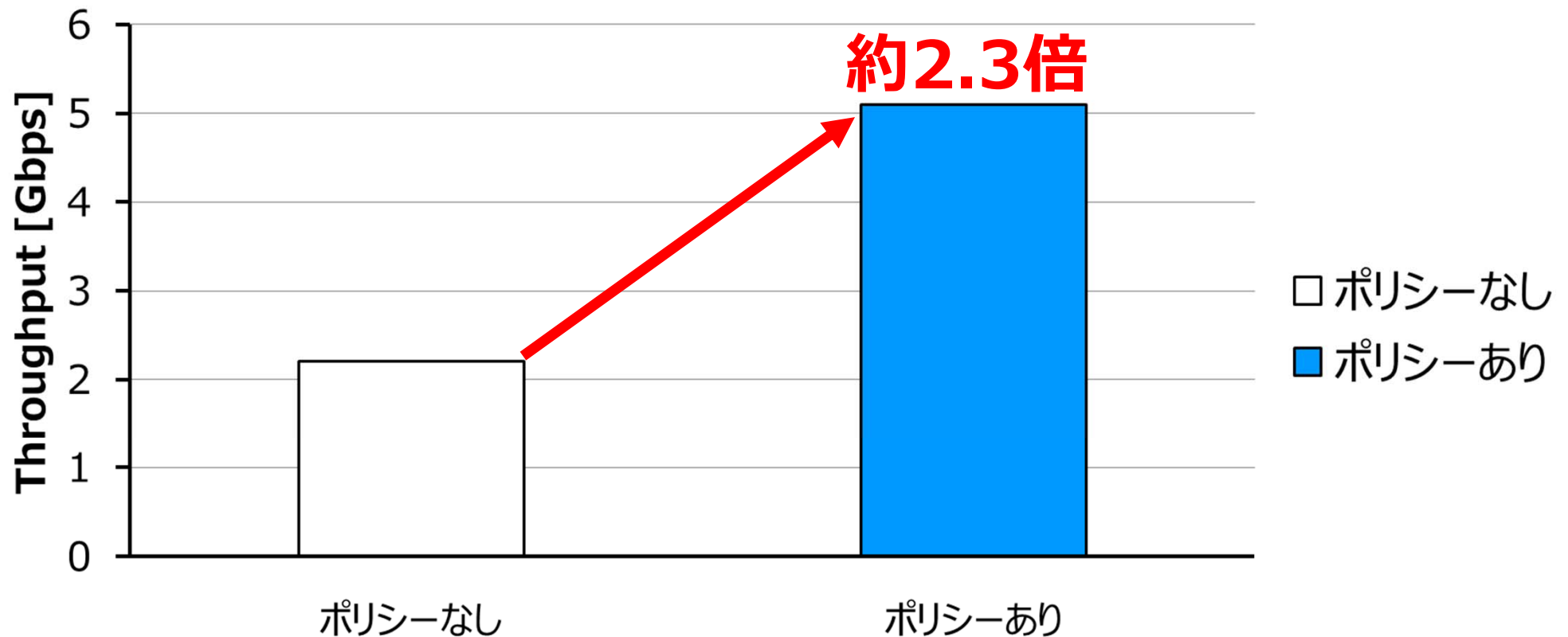
### ■ 評価方法

- **ポリシーなし** : Background traffic/Mice/Elephant同じキュー使用
- **ポリシーあり** : Background traffic/Mice同じキュー使用  
**Elephant専用キュー使用**

## ■ Elephantのスループット

- **ポリシーなし** : Background traffic/Mice/Elephant同じキュー使用
- **ポリシーあり** : Background traffic/Mice同じキュー使用  
Elephant専用キュー使用

Elephantを優先制御するポリシーにより、スループットが**約2.3倍**増加!





## ■ Bundleの作成・実行手順

- GBP RendererのCode (JAVA)を修正して、Mavenでコンパイル
- ODLの実行・ODLでBundleのinstall

## ■ ポリシーを適用する試作プログラム作成・実行手順

- OVSでBridgeやVXLANなどを設定してODLと接続
- VXLANを通すために、ifconfigでMTUサイズを修正
- Restconfでポリシーを適用するための試作プログラムの実行 (Python)
  - Pythonだけではなく、Restconfを実行できるものであれば何でも良い

## ■ Testをskipして実行も可能

- Source codeを修正し、Testを通らないケースがたくさん存在…

```
[root@odl groupbasedpolicy]# mvn clean install package -Dmaven.test.skip=true
```

```
[INFO] --- maven-antrun-plugin:1.3:run (default) @ distribution-karaf ---
[INFO] Executing tasks
[INFO] Executed tasks
[INFO]
[INFO] --- karaf-maven-plugin:3.0.1:instance-create-archive (package) @ distribution-karaf ---
[INFO] -----
[INFO] Reactor Summary:
[INFO]
[INFO] commons.groupbasedpolicy ..... SUCCESS [2.294s]
[INFO] groupbasedpolicy.project ..... SUCCESS [0.040s]
[INFO] groupbasedpolicy ..... SUCCESS [29.046s]
[INFO] groupbasedpolicy-ofoverlay-config ..... SUCCESS [0.525s]
[INFO] features-groupbasedpolicy ..... SUCCESS [1.905s]
[INFO] distribution-karaf ..... SUCCESS [8.193s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 43.853s
[INFO] Finished at: Mon Oct 27 09:38:51 EDT 2014
[INFO] Final Memory: 96M/634M
[INFO] -----
[root@odl groupbasedpolicy]#
```

Build成功!

## ■ ODL helium (karaf distribution)

```
[root@odl bin]# ./karaf
```



```
Hit '<tab>' for a list of available commands  
and '[cmd] --help' for help on a specific command.  
Hit '<ctrl-d>' or type 'system:shutdown' or 'logout' to shutdown OpenDaylight.
```

```
opendaylight-user@root> feature:uninstall odl-groupbasedpolicy-ofoverlay  
opendaylight-user@root> feature:install odl-groupbasedpolicy-ofoverlay  
opendaylight-user@root>
```

作成したbundleを  
実行するために、  
Uninstall/installを実施

```
[root@odl 0.2.0-SNAPSHOT]# pwd  
/root/.m2/repository/org/opendaylight/groupbasedpolicy/groupbasedpolicy/0.2.0-SNAPSHOT  
[root@odl 0.2.0-SNAPSHOT]# ls -al  
total 1020  
drwxr-xr-x. 2 root root 4096 Oct 27 22:38 .  
drwxr-xr-x. 3 root root 4096 Oct 27 22:09 ..  
-rw-r--r--. 1 root root 1014710 Oct 27 22:38 groupbasedpolicy-0.2.0-SNAPSHOT.jar  
-rw-r--r--. 1 root root 8282 Oct 16 11:22 groupbasedpolicy-0.2.0-SNAPSHOT.pom  
-rw-r--r--. 1 root root 735 Oct 27 22:38 maven-metadata-local.xml  
-rw-r--r--. 1 root root 209 Oct 27 22:38 _remote.repositories
```

Mavenで生成した  
GBPが存在する  
Directory!  
(.m2/repositoryの下)

## ■ 内部ヘッダのDSCPマーキングはVXLAN外部ヘッダにコピー

```
lch@odl-03:~$ sudo ovs-vsctl show
d48477f2-40a5-4d79-8db8-d44bc2563267
  Bridge "s3"
    Controller "tcp:192.168.1.220:6653"
      is_connected: true
    Port "s3"
      Interface "s3"
        type: internal
    Port "s3-eth1"
      Interface "s3-eth1"
        type: internal
    Port "s3-eth0"
      Interface "s3-eth0"
        type: internal
    Port "s3_vxlan0"
      Interface "s3_vxlan0"
        type: vxlan
        options: {key=flow, remote_ip="192.168.100.20", source_ip="192.168.100.10", tos=inherit}
  ovs_version: "2.3.0"
lch@odl-03:~$ █
```

これによってDSCPが  
VXLANのOuterヘッダにコピーされる

## ■ 内部ヘッダのDSCPマーキングはVXLAN外部ヘッダにコピー

```
lch@odl-04:~/workspace/groupbasedpolicy/util/test0f0verlay$ sudo ovs-vsctl show
22aff420-834a-4d71-8e17-5bd22ef13878
  Bridge "s4"
    Controller "tcp:192.168.1.220:6653"
      is_connected: true
    Port "s4-eth0"
      Interface "s4-eth0"
        type: internal
    Port "s4_vxlan0"
      Interface "s4_vxlan0"
        type: vxlan
        options: {key=flow, remote_ip="192.168.100.10", source_ip="192.168.100.20", tos=inherit}
    Port "s4"
      Interface "s4"
        type: internal
    Port "s4-eth1"
      Interface "s4-eth1"
        type: internal
  ovs_version: "2.3.0"
lch@odl-04:~/workspace/groupbasedpolicy/util/test0f0verlay$ █
```

これによってDSCPが  
VXLANのOuterヘッダにコピーされる



# S3のインターフェースの構成

## ■ VXLANのためにMTUの調整が必要

■ eth7 : 物理インターフェース

■ s3-eth0, s3-eth1 : OVSのインターフェース

```
lch@odl-03:~$ ifconfig eth7
```

```
eth7      Link encap:Ethernet  HWaddr 00:19:99:6d:40:23
          inet addr:192.168.100.10  Bcast:192.168.100.255  Mask:255.255.255.0
          inet6 addr: fe80::219:99ff:fe6d:4023/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9000 Metric:1
          RX packets:49353997 errors:0 dropped:0 overruns:0 frame:0
          TX packets:74725650 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:57702627743 (57.7 GB)  TX bytes:344139349345 (344.1 GB)
```

物理インターフェース  
(Mice/Elephantを通すため)

```
lch@odl-03:~$ ifconfig s3-eth0
```

```
s3-eth0   Link encap:Ethernet  HWaddr de:e5:a0:3a:66:e3
          inet addr:10.0.35.2  Bcast:10.0.35.255  Mask:255.255.255.0
          inet6 addr: fe80::dce5:a0ff:fe3a:66e3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500 Metric:1
          RX packets:31173943 errors:0 dropped:725 overruns:0 frame:0
          TX packets:11002738 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:39281327994 (39.2 GB)  TX bytes:20189913033 (20.1 GB)
```

Mice用 : Ethernetサイズ

```
lch@odl-03:~$ ifconfig s3-eth1
```

```
s3-eth1   Link encap:Ethernet  HWaddr f6:2e:66:b9:98:f3
          inet addr:10.0.36.2  Bcast:10.0.36.255  Mask:255.255.255.0
          inet6 addr: fe80::f42e:66ff:feb9:98f3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:8500 Metric:1
          RX packets:18049030 errors:0 dropped:20661 overruns:0 frame:0
          TX packets:9597091 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:15981114199 (15.9 GB)  TX bytes:315782274804 (315.7 GB)
```

Elephant用:Jumboサイズ



# S4のインターフェースの構成

## ■ VXLANのためにMTUの調整が必要

■ eth4 : 物理インターフェース

■ s4-eth0, s4-eth1 : OVSのインターフェース

```
lch@odl-04:~$ ifconfig eth4
eth4      Link encap:Ethernet  HWaddr 00:19:99:6d:40:57
          inet addr:192.168.100.20  Bcast:192.168.100.255  Mask:255.255.255.0
          inet6 addr: fe80::219:99ff:fe6d:4057/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9000 Metric:1
          RX packets:74366145 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48610557 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:344122070602 (344.1 GB)  TX bytes:57648567580 (57.6 GB)
```

物理インターフェース  
(Mice/Elephantを通すため)

```
lch@odl-04:~$ ifconfig s4-eth0
s4-eth0   Link encap:Ethernet  HWaddr 76:06:a4:3e:86:50
          inet addr:10.0.35.4  Bcast:10.0.35.255  Mask:255.255.255.0
          inet6 addr: fe80::7406:a4ff:fe3e:8650/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500 Metric:1
          RX packets:23715113 errors:0 dropped:2579 overruns:0 frame:0
          TX packets:5641998 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:21028239933 (21.0 GB)  TX bytes:37594656043 (37.5 GB)
```

Mice用 : Ethernetサイズ

```
lch@odl-04:~$ ifconfig s4-eth1
s4-eth1   Link encap:Ethernet  HWaddr 4e:11:34:8c:03:ad
          inet addr:10.0.36.4  Bcast:10.0.36.255  Mask:255.255.255.0
          inet6 addr: fe80::4c11:34ff:fe4e:3ad/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:8500 Metric:1
          RX packets:50378967 errors:0 dropped:20753 overruns:0 frame:0
          TX packets:8082746 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:318473568026 (318.4 GB)  TX bytes:15323350161 (15.3 GB)
```

Elephant用:Jumboサイズ

# ポリシーを適用する試作プログラムの実行

```
lch@odl-04:~/workspace/groupbasedpolicy/util/testElephantPolicy$ ls
201410261543_s4.txt  elephantPolicy.py  mininet_gbp.py  ms_config.py~  odl_gbp.py~
config.py           lch_config.py      mininet_gbp.py~ odl_gbp_flow.py odl_gbp.pyc
config.py~          lch_config.py~     mininet_gbp.pyc odl_gbp_flow.py~
config.pyc          lch mininet gbp.py ms config.py    odl gbp.py
```

プログラムの実行

```
lch@odl-04:~/workspace/groupbasedpolicy/util/testElephantPolicy$ sudo ./elephantPolicy.py --controller 192.168.1.220 --policy
```

```
PUT http://192.168.1.220:8181/restconf/config/opendaylight-inventory:nodes
```

```
{
  "opendaylight-inventory:nodes": {
    "node": [
      {
        "id": "openflow:0000000000000001",
        "ofoverlay:tunnel-ip": "192.168.100.10"
      },
      {
        "id": "openflow:0000000000000002",
        "ofoverlay:tunnel-ip": "192.168.100.20"
      }
    ]
  }
}
```

RESTconfを利用してODLで登録

```
PUT http://192.168.1.220:8181/restconf/config/policy:tenants
```

```
{
  "policy:tenants": {
    "tenant": [
      {
        "contract": [
          {
            "clause": [
              {
                "name": "allow-tcp-clause",
                "subject-refs": [
                  "allow-tcp-subject",
                  "allow-icmp-subject"
                ]
              }
            ]
          }
        ]
      }
    ]
  }
}
```

RESTconfを利用してODLで登録



## ■ デモを作る過程でトラブったこと、注意してほしいことなど

### ■ Python

#### ■ 版数に注意

- Python 2.7.X (○)
- Python 2.6.X (✗) : 試作プログラムのために、パッケージが足りない…

### ■ Ubuntu vs CentOS

#### ■ CentOS

- mininetなどの環境構築が面倒であり、CentOS6.5ではmininetを遠隔のODL(controller)に接続しようとしたら、うまくいかないケースが存在

#### ■ Ubuntu

- パッケージがそろっているので、環境構築がCentOSより楽

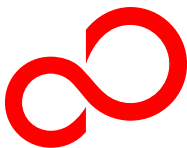
### ■ OpenFlow

- OVSに版数よって、OpenFlowのルールの記述が異なる

### ■ Karaf

#### ■ Feature-xx-all

- e.g.) mdsal, restconfなどを全部実行するとGBPと衝突して、restconfのrpcなどがうまく動かない恐れが存在する



**FUJITSU**

shaping tomorrow with you