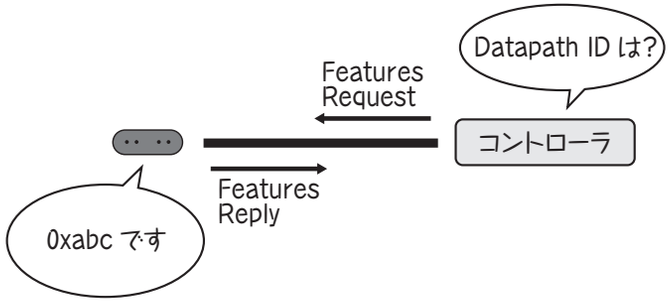
## スイッチのDatapath IDの確認

次にコントローラは接続したスイッチのDatapath IDを確認します。コントローラがスイッチにFeatures Requestメッセージを送ると、スイッチはDatapath IDとスペックを乗せたFeatures Replyメッセージを返答します（図2-3）。

Features Replyメッセージには Datapath IDに加えて、主に次のスペック情報が入っています。

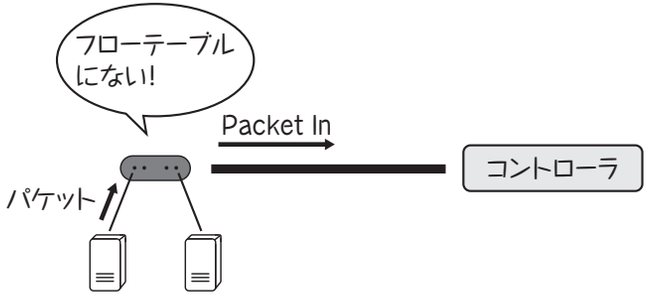
- 一度にバッファできるパケットの数
- サポートするテーブルの数
- サポートする機能の一覧




▲図2-3　Features Requestメッセージでスイッチの Datapath IDを確認

## コントローラへの受信パケットの通知

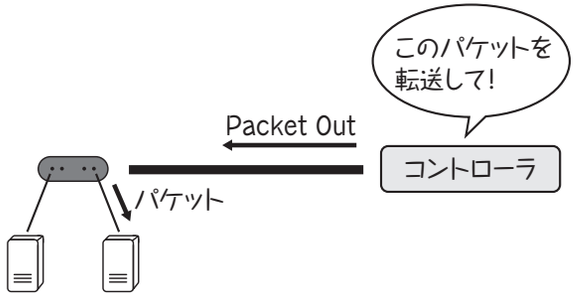
スイッチは、受信したパケットと関連情報をPacket Inメッセージでコントローラへ通知できます。たとえば、フローテーブルに登録していない通信を検知した場合など、Packet Inメッセージを使ってパケットの情報をコントローラへ送ります（図2-4）。




▲図2-4　受信パケットとその情報がPacket Inメッセージとしてコントローラに上がる

## パケットの出力

Packet Outメッセージは Packet Inメッセージの逆で、スイッチからパケットを出力するためのメッセージです（図2-5）。




▲図2-5　Packet Outメッセージでパケットをスイッチから出力

Packet Outの典型的な利用例は、Packet Inでコントローラへ届いたパケットを宛先に届ける場合です。もしもPacket Inの後にPacket Outをやらないと、パケットはコントローラに残ったままで宛先には届きません。

## フローテーブルの更新

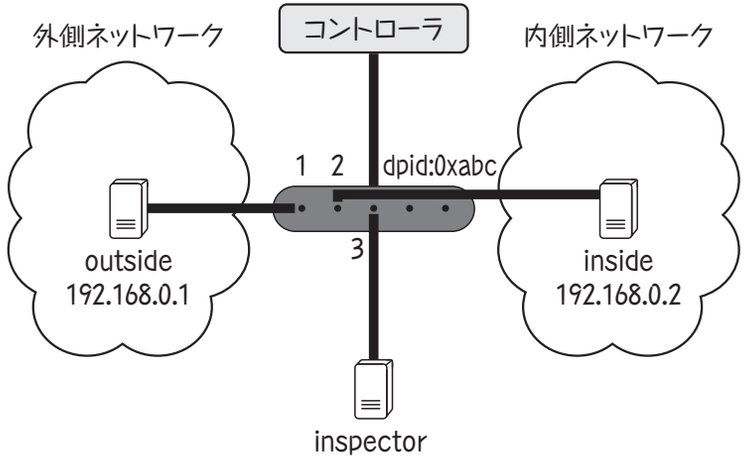
Flow Modメッセージはスイッチのフローエントリを追加・削除・変更

```
$ git clone https://github.com/trema/transparent_firewall.git ↵
```

ダウンロードしたソースツリー上で bundle install --binstubs を実行すると、Trema などの実行環境一式を自動的にインストールできます。

```
$ cd transparent_firewall ↵
$ bundle install --binstubs ↵
```

GitHub から取得したソースリポジトリ内に、仮想スイッチ 1 台、仮想ホスト 3 台の構成を持つ設定ファイル trema.conf（**リスト 11-1**）が含まれています（**図 11-3**）。




▲図 11-3　BlockRFC1918 を実行するための仮想ネットワーク構成



▲リスト 11-1　trema.conf

```
vswitch('firewall') { datapath_id 0xabc }

vhost('outside') { ip '192.168.0.1' }
vhost('inside') { ip '192.168.0.2' }
vhost('inspector') {
  ip '192.168.0.3'
  promisc true
}

link 'firewall', 'outside'
link 'firewall', 'inside'
link 'firewall', 'inspector'
```

ホスト outside は外側のネットワーク、たとえばインターネット上のホストとして動作します。ホスト inside は内側のネットワークのホストです。ホスト inspector は BlockRFC1918 ファイアウォールが落としたパケットを調べるためのデバッグ用ホストです。inspector は outside または inside 宛のパケットを受け取るので、promisc オプションを有効にすることで自分宛でないパケットも受け取れるようにしておきます。

では、いつものように trema run の -c オプションにこの設定ファイルを渡して BlockRFC1918 コントローラを実行してみましょう。

```
$ ./bin/trema run ./lib/block_rfc1918.rb -c trema.conf ↵
0xabc: connected
0xabc: loading finished
```

別ターミナルを開き、trema send_packets コマンドを使って、outside と inside ホストの間でテストパケットを送ってみます。

```
$ ./bin/trema send_packets --source outside --dest inside ↵
$ ./bin/trema send_packets --source inside --dest outside ↵
```

outside と inside はどちらもプライベートアドレスを持つので、BlockRFC1918 コントローラがパケットを落とすはずです。落としたパ