Software Defined Networking (SDN) - Open Flow

软件定义网络

RTOAX
BEYOND

Introduction

平等

- Current Internet: *egalitarian* – routing/delivery based on destination address, best effort.

当前网络：根据地址路由转发

- Future Internet: *criteria* based traffic management, paths predetermined based on traffic type.

未来网络：标准的基于拥塞管理，路劲由拥塞类型决定

转发图

IP Forwarding Paradigm

- All traffic from the same source to the same destination, follows the same path.
- Datagram is routed independent on its content, or source or destination.
- We have discussed possible variations on this model:
  - IP ToS bits
  - MPLS 多协议标签交换（英语：Multi-Protocol Label Switching，缩写为MPLS）是一种在开放的通信网上利用标签引导数据高速、高效传输的新技术。
  - OSPF and different route maps

OSPF一般指组播扩展OSPF。OSPF(Open Shortest Path First开放式最短路径优先）是一个内部网关协议(Interior Gateway Protocol，简称IGP），用于在单一自治系统（autonomous system,AS）内决策路由。

Traffic Engineering & Path Selection

最短的

Moving away from a paradigm of finding shortest paths for datagrams to follow, to one where a network administrator can set paths for individual flows, where a flow is defined by a set of criteria – traffic classification.

Connection Oriented Networks and Routing Overlays

面向网络的链接

路由覆盖

- Providing per-flow control in a network:
  - Use a connection oriented network 面向网络链接

  or
  - Impose routing overlays in a packet-switched architecture.

面向连接

概述了面向连接的网络的步骤

Connection Oriented

- Each flow is set up independently 流独立
- Application cannot transmit data until the forwarding path has been set up
- Policies are applied at each switch to determine forwarding path and rules for accepting a flow.
- Switch uses these policies to determine whether to accept a flow and what path to use.
- Once accepted, it updates its forwarding table.
- Each flow has to be terminated by the application.

流必须被应用终止

路由覆盖

Routing Overlays

- A forwarding system that imposes a virtual network topology on top of an underlying packet switched architecture.

网络拓扑

- The virtual topology consists of tunnels that are pre-set to route specific traffic flows over.

拓扑

- The tunnel is like a point to point connection. The routing protocols (for the virtual topology) only find path across the tunnels.

点对点

- MPLS is an example of this type of path flow control.

多协议标签交换（英语：Multi-Protocol Label Switching，缩写为MPLS）是一种在开放的通信网上利用标签引导数据高速、高效传输的新技术。

Why a new approach

- Neither one of the two methods described are perfect.
- Connection oriented less flexible but higher speeds.
面向连接不flex
- Routing overlays, more flexibility – software defined, but hide underlying infrastructure so cannot be optimized
路由覆盖
但是基础设施基本上不能被优化
- SO – why not combine both – a hybrid approach!

为什么不联合二者呢/

SDN :混合方法

SDN: A hybrid Approach

- Can we combine the strengths of both to create the PERFECT network??? Some say YES!!!
  - Perform classification in hardware
  - Use high speed forwarding hardware
  - Avoid dynamic routing protocols that optimize on shortest path and have managers set routing policies that conform to traffic flow needs
  - To scale, allow network management software to configure and control the network devices, i.e., take the human out of the day to day management

缩放

Separation of Data and Control

- NOT a new concept….. X.25 was all about that!
- What does it mean?
- Divide the functionality of a network device, aka, a router/switch into two parts:
  - A control part, called a control plane that allows managers to configure and control the device
  - A data part, called the data plane that only handles packet processing and forwarding.

Control and Data Planes

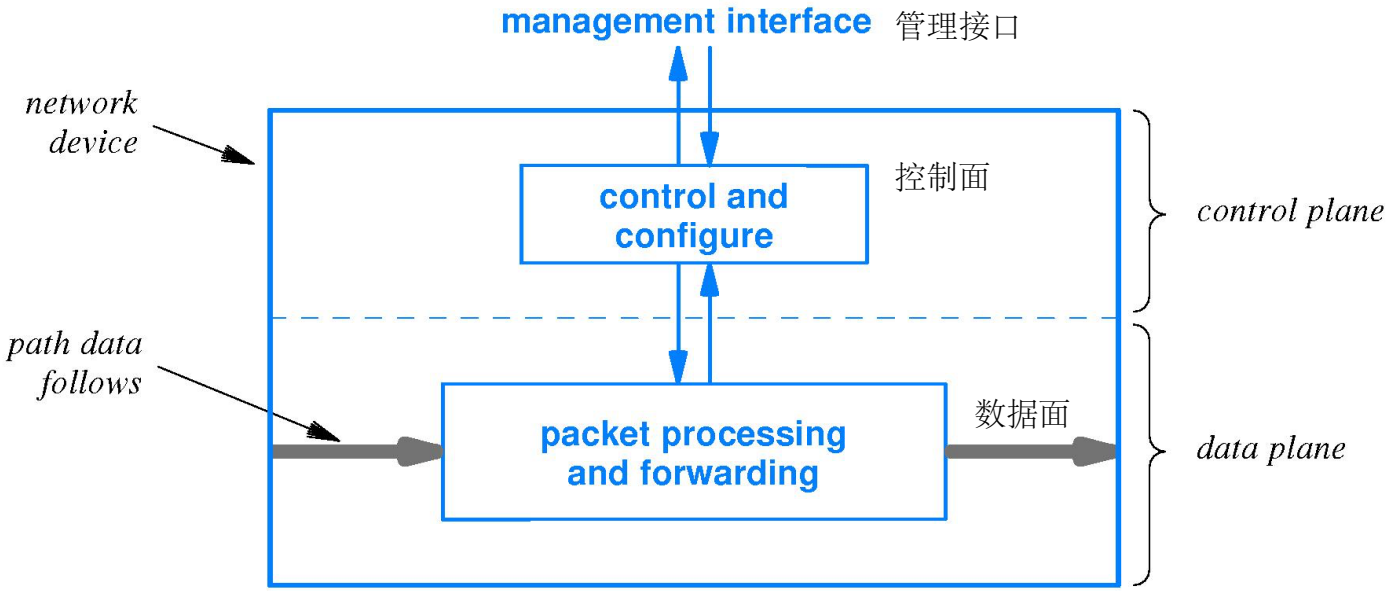
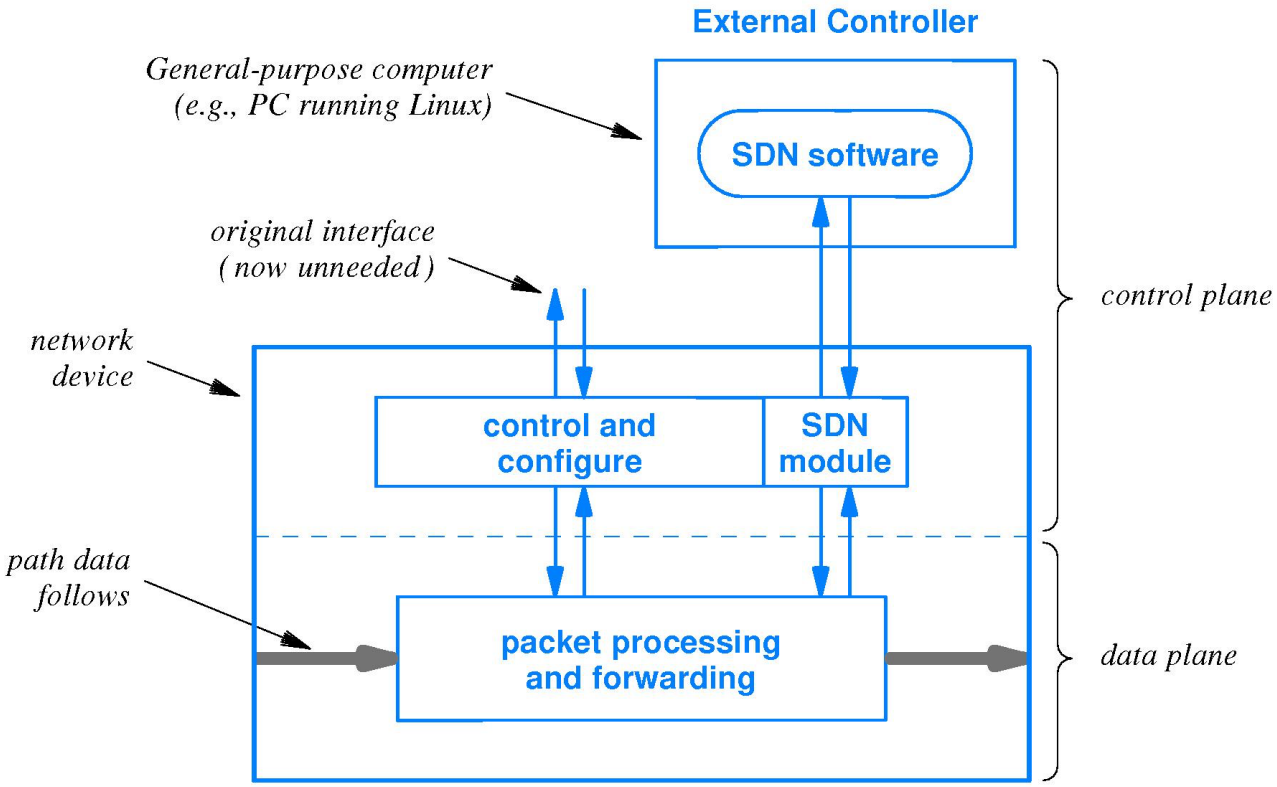


Illustration of the two conceptual parts of a network device: the control and data planes.

The control path has a much lower capacity than the data path as illustrated by the size of the arrows being used in the diagram above.

SDN Architecture and External Controllers



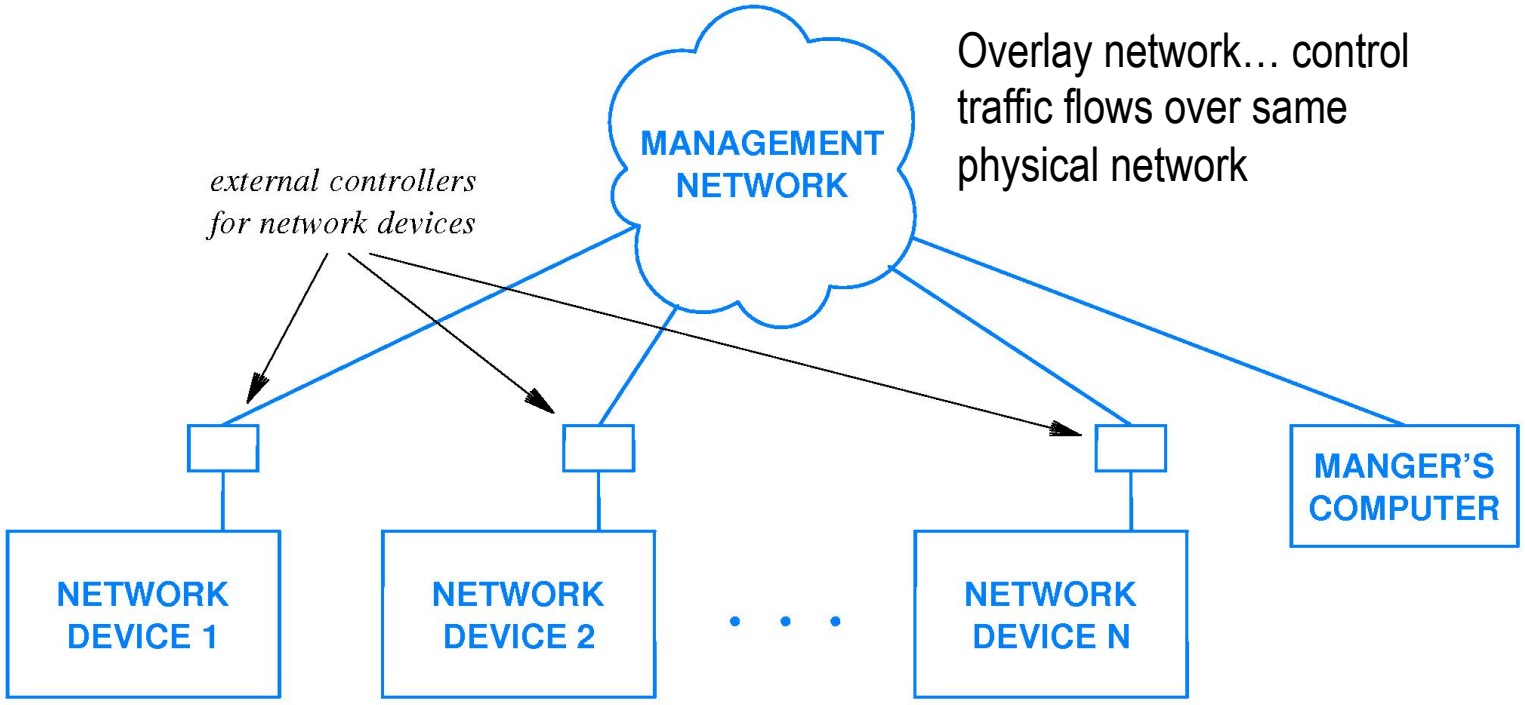
The basic SDN architecture with an external controller configuring classification and forwarding hardware in a network device.

To operate on any vendor switch, SDN uses an augmented approach: deactivates vendor management interface and uses an external system to configure the switch.

Why the Augmented Approach?

- SDN Module is very minimalistic – only used to pass commands to the data plane.
- All the intelligence lies in the external controller that is completely under the control of the network engineer.
- The controller bypasses the vendor control plane allowing more flexibility in defining network control strategies.

SDN: Multiple Network Devices



An idealized interconnection of SDN controllers on a separate management network. Management applications on the controllers use the management network to coordinate with one another.

The management network, is really the control plane that spans across multiple control devices and a manager's computer that controls them all – downloads s/w and policies

OpenFlow – What is it?

- What is OpenFlow?

控制协议，里面有netconf

It is a control protocol.

- It is used to communicate policies and traffic management information between a controller and a switch.
  - In other words data plane related information for the switch to use to set up its data paths.
- It operates like the SDN module shown in the switch designs earlier.
  - Translates the control messages to switch commands

OpenFlow Technology

- Specifies:
  - Communication used between controller and switch
  - Set of items that an be configured and controlled in a switch
  - Format of messages that a controller an switch use to communicate with.

Communication

SDN不适用tcp？

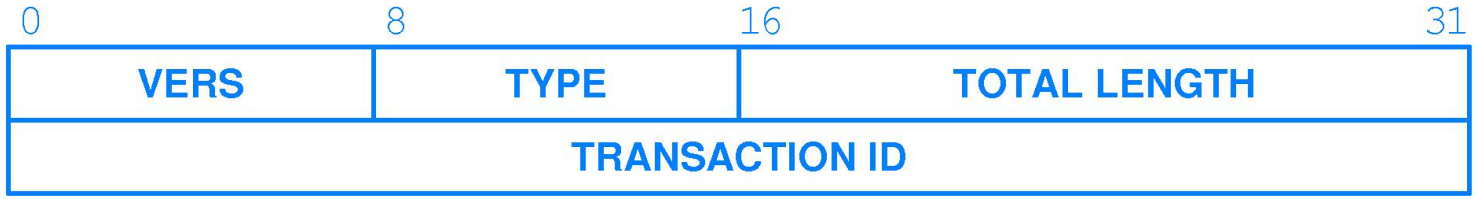
- TCP over SSL is specified as the mode of communication over the regular production network (i.e., non SDN network) for controllers to communicate with a switch.
- Require a reliable connection
- Require a secure connection
- Using TCP means that a controller need not be co-located with switch, and can communicate with switch over a network.
- Using SSL means that TCP connection is secure.

Configuration Items

- Minimum specification of OpenFlow (Type 0) requires that a switch have a Flow Table that implements classification and forwarding.
  1. Classification is based on a set of patterns that are matched against packets.
  2. Each entry in the Flow Table has an action associated with it – how to process the packet.
- Statistics are maintained for each action: count of packets, size of packets and timestamps. Used for traffic engineering.

Message Format

消息结构



The fixed-size header used for every OpenFlow message.

Vers: specifies version type, e.g., 0x02 is vers 2

Type: specifies type of message. There are 24 message types, e.g.:

- Controller to switch
- Asynchronous switch to controller – reporting an event
- Symmetric – response required

Total Length: includes payload and header measure in octets

Transaction ID: unique number that allows controlled to match requests with replies

OpenFlow Patterns

- Minimal requirements for a Type 0 Switch
  - Must be able to match a specific list of fields.
  - Cannot distinguish between ARP request and reply
  - Cannot identify specific ICMP messages, e.g., ping traffic.
  - Because of Ethernet Type Field, it can allow experimenters to use an unassigned Ethernet Type

Type 0 Switch OpenFlow

Field	Meaning
In Port	Switch port over which the packet arrived
Ether src	48-bit Ethernet source address
Ether dst	48-bit Ethernet destination address
Ether Type	16-bit Ethernet type field
VLAN id	12-bit VLAN tag in the packet
IPv4 src	32-bit IPv4 source address
IPv4 dst	32-bit IPv4 destination address
IPv4 Proto	8-bit IPv4 protocol field
TCP/UDP/SCTP src	16-bit TCP/UDP/SCTP source port
TCP/UDP/SCTP dst	16-bit TCP/UDP/SCTP destination port

Figure 28.6 Fields in packet headers that can be used for classification in a Type 0 OpenFlow switch.

Actions of a Type 0 Switch

Action	Effect
1	Forward the packet to a given switch port or a specified set of switch ports.
2	Encapsulate the packet and send to the external controller for processing.
3	Drop (discard) the packet without any further processing.

Possible actions a Type 0 OpenFlow switch can take when a packet matches one of the classification rules.

Action 1: Most common case: a switch will be pre-configured with rules and all that needs to be done is follow forwarding commands when pattern is matched.

Action 2: Allows switch to handle packets that don't match a pattern, i.e., no forwarding rule has been set up. Used to handle per flow forwarding. E.g., all TCP connection requests are encapsulated and sent to controller to set up a new classification for that flow. Packet then returned for further processing by switch based on new classification rule.

Action 3: Allow handling of problem traffic such as DOS, or over active/excessive broadcasting. The controller can identify the source and set up a special rule to handle that traffic.

OpenFlow Extensions and Additions

- Additions can be classified into 5 categories:
  - Multiple Flow Tables that are arranged in a pipeline
  - Additional packet header fields for matching
  - A field used to pass information along the pipeline
  - New actions that provide significant functionality
  - A Group table that allows a set of actions to be performed

Pipeline of Flow Tables

- Several Flow Tables are linked by conditions.
- If you satisfy the first match you jump to another Flow Table. You can satisfy another match that takes you to a next table or if not an action is taken at that point.
- This allows sub branching on matching rules. Initial table not very large. Pipelined tables allow extensions to one common classifier.

Additional Packet Header Fields

Field	Meaning
Ingress Port	Switch port over which the packet arrived
Metadata	64-bit field of metadata used in the pipeline
Ether src	48-bit Ethernet source address
Ether dst	48-bit Ethernet destination address
Ether Type	16-bit Ethernet type field
VLAN id	12-bit VLAN tag in the packet
VLAN priority	3-bit VLAN priority number
MPLS label	20-bit MPLS label
MPLS class	3-bit MPLS traffic class
IPv4 src	32-bit IPv4 source address
IPv4 dst	32-bit IPv4 destination address
IPv4 Proto	8-bit IPv4 protocol field
ARP opcode	8-bit ARP opcode
IPv4 tos	8-bit IPv4 Type of Service bits
TCP/UDP/SCTP src	16-bit TCP/UDP/SCTP source port
TCP/UDP/SCTP dst	16-bit TCP/UDP/SCTP destination port
ICMP type	8-bit ICMP type field
ICMP code	8-bit ICMP code field

Fields available for use with Version 1.1 of OpenFlow.

Matching Header Fields

- Current version 1 of OpenFLow only does field matching.
- In newer versions, instead of matching the header field, OpenFlow will specify:

  (bit_offset, length, pattern)

- Why? To take advantage of underlying switch hardware that might be able to do bit matching for faster packet processing.

Intra Pipeline Communications

- Metadata Field in the Fields available for OpenFlow, is used to communicate data between Flow Tables.
- OpenFlow does not specify the content of the Metadata field. Next stages of the pipeline need to know the contents and format of the information in the Metadata field.
  - For example: at the first table, we may have to compute the next hop IP address. It passes that along to the next FlowTable which knows what to do with it, i.e., use it for pattern matching for next level processing.

New Actions

- In new version, an action main not be performed immediately when a pattern match occurs.
- Instead a set of actions are accumulated as the packet moves along a pipeline.
- Actions can be added or removed at each stage, as a packet traverses the pipeline.
- When a packet arrives at a stage where there is no action, basically the END of the pipeline, all the actions are performed.
- The set of actions has to contain an output action, that specifies how to release the packet, e.g., exit port number.

Example of New Actions

- OpenFlow defines a list of required actions that have to be supported, and a list of recommended actions that are optional for implementation at a switch.
- New actions:
  - TTL manipulation
    - Forward a packet to switch's local protocol stack
    - Several actions related to MPLS – Encapsulation and de encapsulation with an MPLS header
    - Qos (priority) queueing

Group Table - Functionality

- Adds flexibility to the forwarding paradigm.
- An entry can specify how to forward a packet (i.e., specific port), and multiple patterns can point to that entry.
- Or an entry can point to a group of exit ports based on a pattern match. In other words, several outgoing links can be aggregated to support a particular traffic flow (or set of flows if more pattern matches result in that exit action).

Group Table Entries

- A group table entry consists of 4 items:
  - 32 bit identifier – uniquely identifies a group
  - Type – see table next slide
  - Counters – collect statistics
  - Action Buckets – identify actions in an ordered list that are used only if the previous action cannot be completed. For example a failed link – Identify a backup port to send data to in that case. Can specify a set of backup links to be used. This is used with the fast fail over type of action

Action Types

Type	Meaning
all	Execute all Action Buckets for the group (e.g., to handle broadcast)
select	Execute one Action Bucket (e.g., using a hash or round-robin algorithm for selection)
indirect	Execute the only Action Bucket defined for the group (designed to allow multiple Flow Table entries to point to a single group)
fast fail-over	Execute the first live Action Bucket (or drop the packet if no bucket is live)

The four OpenFlow group types and the meaning of each.

Uses of OpenFlow

- Openflow allows a network administrator to manage forwarding in a switch as a function of certain header field matchings.
- Some examples:
  - Experimental protocol used between two hosts (identified in the Ethernet Type field)
  - Source based IP forwarding
  - On demand VPN set up between two sites (per TCP connection)

Reality Check on OpenFlow

- Although OpenFlow gives us an environment to rethink network management, it has some shortcomings.
- It does not extend the SDN functionality to all network devices, most of the focus is on switches.
- It is defined for use over Ethernet. Ignores WiFi and other framing types on digital circuits
- Currently IPv4 focused, emerging versions extend its functionality to more protocol types, e.g., IPv6.

Sofware Defined Radio (SDR)

- In the same vein as SDN, there has emerged a new standard for use with radio devices – SDR
- It allows a network engineer to reconfigure the frequencies being used by the device dynamically based on capacity need or interference.
- Limitation is the antenna design, has to be able to handle the frequencies that can be chosen. New antenna technology is emerging that addresses this issue. E.g., Mutliple smaller antennas to additively can achieve the required frequency range.

RTOAX
BEYOND