Introduction to P4 and BMv2

Elie Kfoury, Jorge Crichigno
University of South Carolina
http://ce.sc.edu/cyberinfra
ekfoury@email.sc.edu, jcrichigno@cec.sc.edu

The Cyberinfrastructure Lab at UofSC
Energy Sciences Network (ESnet)
National Science Foundation (NSF)

Internet2 Technology Exchange

Monday December 5th, 2022
Denver, Colorado

Hands on Session 1: Writing a Parser for IPv4 and IPv6

Programmable Parser

- The parser enables parsing arbitrary headers with a finite state machine
- The state machine defines the order of the headers within the packets
- The packet is split into the defined headers and the remaining is treated as the payload
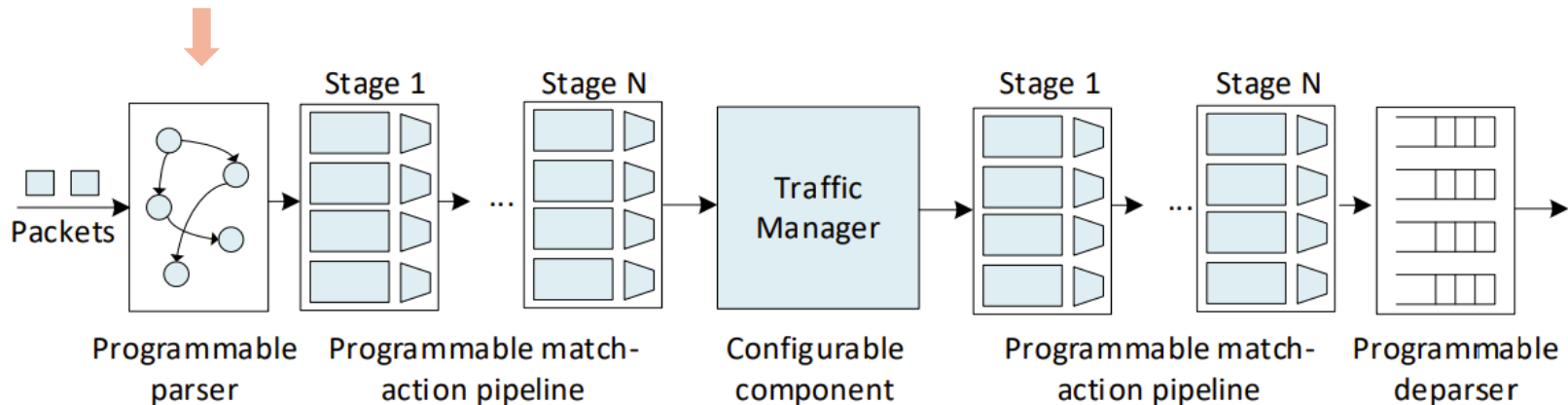
Packet Headers

- The packet headers are specified by the programmer
- The programmer has the flexibility of defining custom/non-standardized headers
- Such capability is not available in non-programmable devices

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				IHL				DSCP						ECN		Total Length															
32	Identifier																Flags			Fragment Offset												
64	Time To Live								Protocol								Header Checksum															
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

```
header ipv4_t {
    bit<4> version;
    bit<4> ihl;
    bit<8> diffserv;
    bit<16> totalLen;
    bit<16> identification;
    bit<3> flags;
    bit<13> fragOffset;
    bit<8> ttl;
    bit<8> protocol;
    bit<16> hdrChecksum;
    ip4Addr_t srcAddr;
    ip4Addr_t dstAddr;
}
```

Programmable Parser

- The parser enables declaring arbitrary headers with a finite state machine
- The state machine defines the order of the headers within the packets
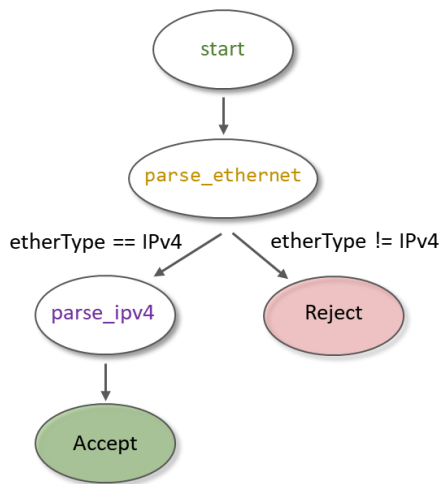


```
state start {
    transition parse_ethernet;
}
state parse_ethernet {
    packet.extract(hdr.ethernet);
    transition select(hdr.ethernet.etherType) {
        TYPE_IPV4: parse_ipv4;
        default: reject;
    }
}
state parse_ipv4 {
    packet.extract(hdr.ipv4);
    transition accept;
}
```

Lab 4: Parser Implementation

Lab Topology and Objectives

- The topology consists of two hosts: h1 and h2; one P4 switch: s1

- Defining the headers for Ethernet, IPv4 and IPv6

- Implementing the parser

- Testing and verifying the switch behavior when IPv4 and IPv6 packets are received
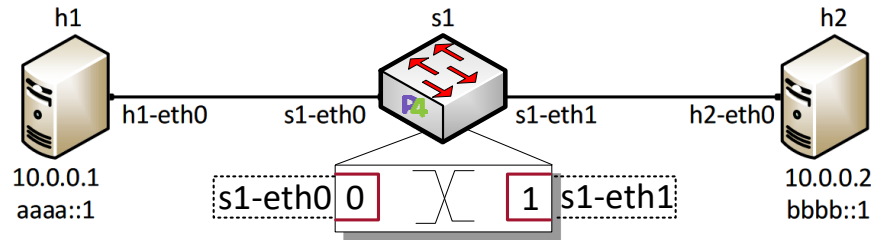
Headers Format

- Ethernet header:

48 bits	48 bits	16 bits
Destination Address	Source Address	Ether Type

- IPv4 header:
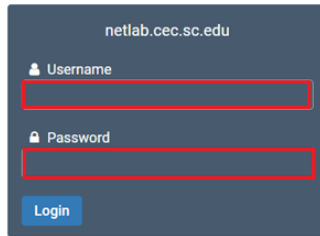
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				IHL				DSCP						ECN		Total Length															
32	Identifier																Flags			Fragment Offset												
64	Time To Live								Protocol								Header Checksum															
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

- IPv6 header:

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				Traffic Class								Flow Label																			
32	Payload Length																Next Header								Hop Limit							
64	Source IP Address																															
192	Destination IP Address																															

Accessing the Platform

- Please use the following link to access the platform:
  - https://netlab.cec.sc.edu/
- Login using your credentials

Accessing the Platform

- Please use the following link to access the platform:
  - https://netlab.cec.sc.edu/
- Login using your credentials

Scheduling a Reservation

- Click on New Lab Reservation
- Click on Schedule Lab for Myself
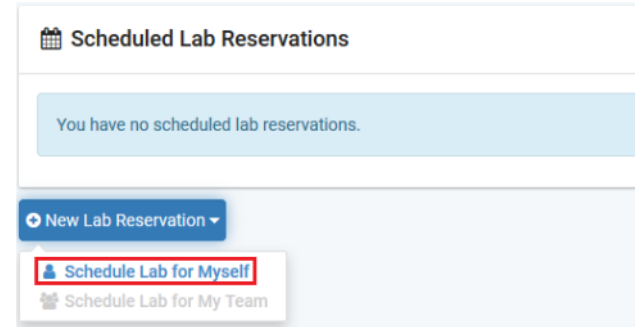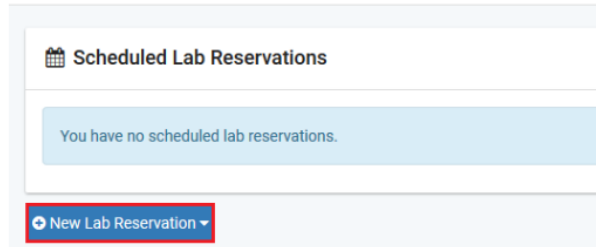
Scheduling a Reservation

- Select the course
- For this session, we will use "Intro. To P4 Programmable Data Planes"

Multiple course topics are available. Please select one.

Intro. to P4 Programmable Data Planes
Introduction to P4 programmable data planes with BMv2

P4 Applications and Custom Processing
This lab series presents P4 applications, stateful elements, and custom packet processing

Scheduling a Reservation

- Select the Lab
- For this session, we will run:
  - Lab 4: Parser Implementation

🧪 **Introduction to P4 programmable data planes with BMv2**

Lab Name	Action
Lab 1: Introduction to Mininet	▾
Exercise 1: Building a Basic Topology	▾
Lab 2: Introduction to P4 and BMv2	▾
Exercise 2: Compiling and Running a P4 Program	▾
Lab 3: P4 Program Building Blocks	▾
Lab 4: Parser Implementation	▾
Exercise 3: Parsing UDP and RTP	▾
Lab 5: Introduction to Match-action Tables (Part 1)	▾
Lab 6: Introduction to Match-action Tables (Part 2)	▾
Exercise 4: Implementing NAT using Match-action Tables	▾
Lab 7: Populating and Managing Match-action Tables at Runtime	▾
Exercise 5: Configuring Match-action Tables at Runtime	▾

Scheduling a Reservation

- Select the next available POD and allocate time

Website URL and Accessing the Platform

- Tutorial website with slides and URL to resources:

  http://ce.sc.edu/cyberinfra/workshop_2022_ie2_p4.html

- Access to virtual platform for this tutorial:

  https://netlab.cec.sc.edu/