A Hands-on Workshop on P4 Programmable Switches

Elie Kfoury, Ali AlSabeh, Jose Gomez
University of South Carolina
http://ce.sc.edu/cyberinfra
ekfoury@email.sc.edu, aalsabeh@email.sc.edu, gomezgaj@email.sc.edu

February 16th, 23rd, 2022
Hands on Session 2: 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

```
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
Lab 4: Parser Implementation
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:**

- **IPv4 header:**

- **IPv6 header:**