IoT MUD Enforcement in the Edge Cloud Using Programmable Switch

Harish S A Hemanth Kothapalli Shubham Lahoti

Kotaro Kataoka Praveen Tammana

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

IIT Hyderabad, India

22nd August 2022

FFSPIN Workshop

ACM SIGCOMM 2022

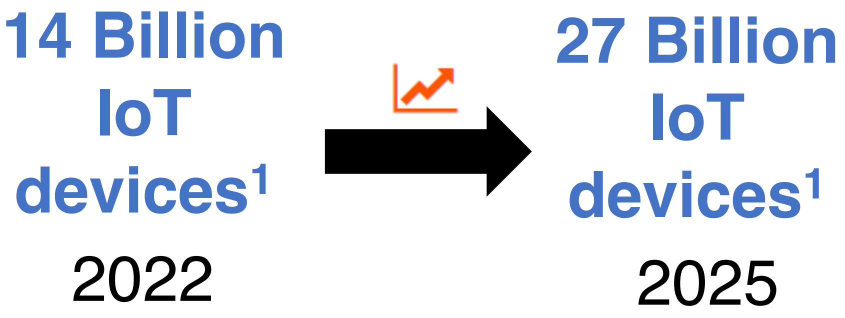
Internet of Things (IoT) security

- Perpetrate attacks on critical Infrastructure

  Bricket Bot: Compromised over 10 million IoT devices

  Mirai Botnet: Targeted DDoS attacks

- Key drivers of attacks
  - Highly competitive market space
  - Very less incentive for security
  - Patching vulnerabilities is difficult



14 Billion IoT devices[1]
2022

27 Billion IoT devices[1]
2025

- Realtime IoT security mechanisms are required

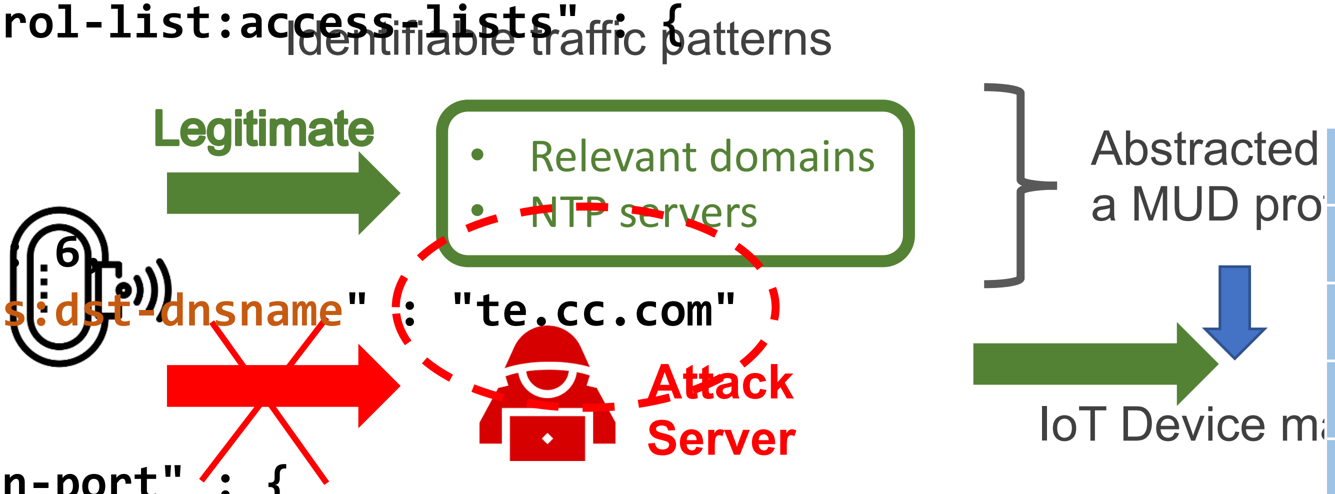
[1] **IoT Analytics, Market insights for Internet of Things:** State of IoT 2022

Manufacturer Usage Description (MUD)

MUD abstracts communication pattern of an IoT device to a MUD profile[1]

Example MUD profile

```
"ietf-access-control-list:access-lists" : {
   ...
  "matches" : {
    "ipv4" : {
      "protocol" : 6
      "ietf-acldns:dst-dnsname" : "te.cc.com"
    },
    "tcp" : {
      "destination-port" : {
        "operator" : "eq",
        "port" : 8777
      },
      "ietf-mud:direction-initiated" : "from-device"
   ...
```

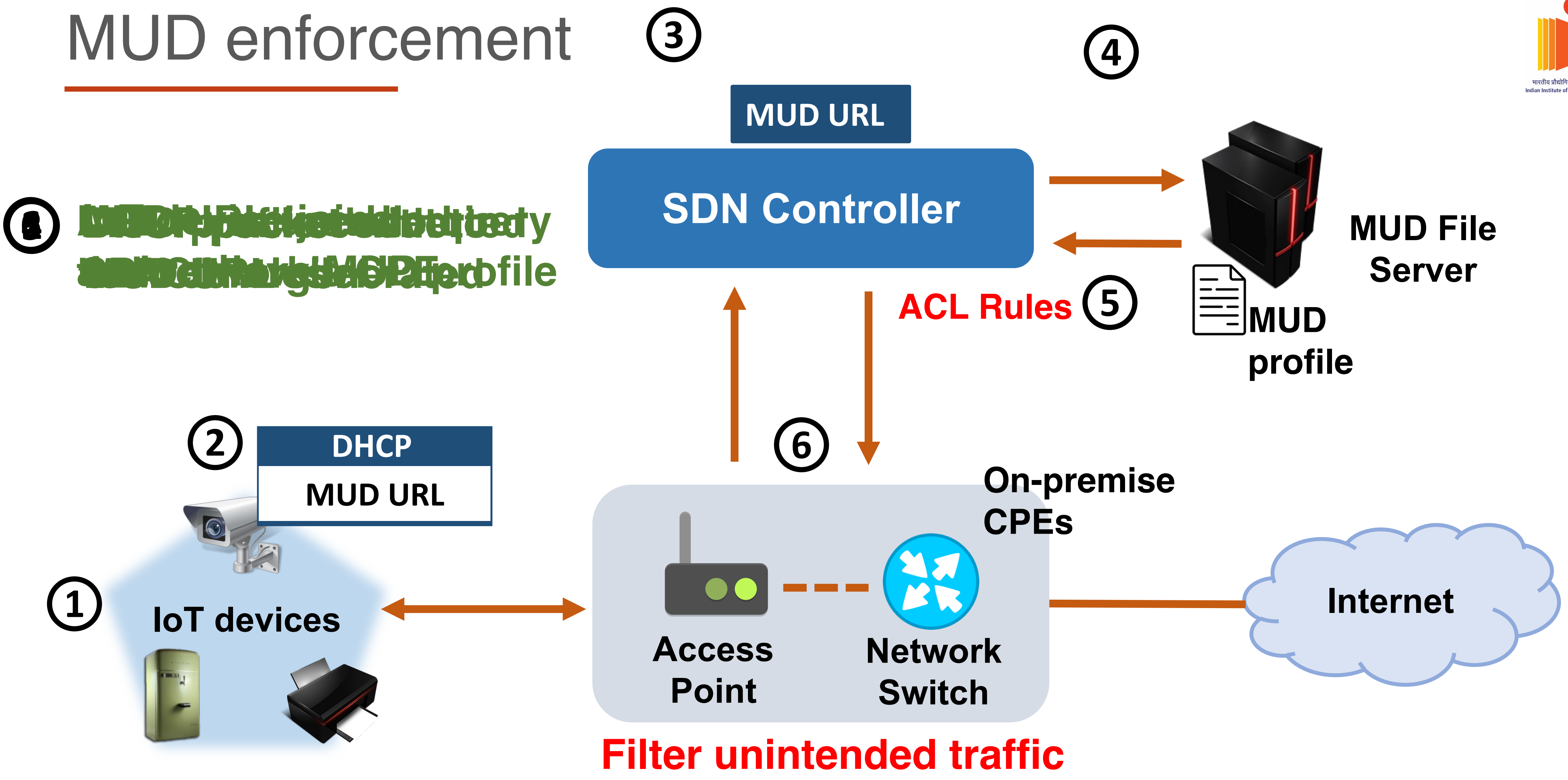


ACL rules

Type Eth	0x0800
Protocol	6
Src Port	*
Dst Port	8777
Src IP	*
Dst IP	te.cc.com 44.45.66.44

[1] National Cybersecurity Center of Excellence (NCCoE) : MUD related Resources

MUD enforcement



Filter unintended traffic

*Combining MUD policies for IDS [IoT S&P'18] | Volumetric attack detection using MUD [SOSR'19] | SoftMUD [NIST, ICN]

Advantages of MUD enforcement at network edge

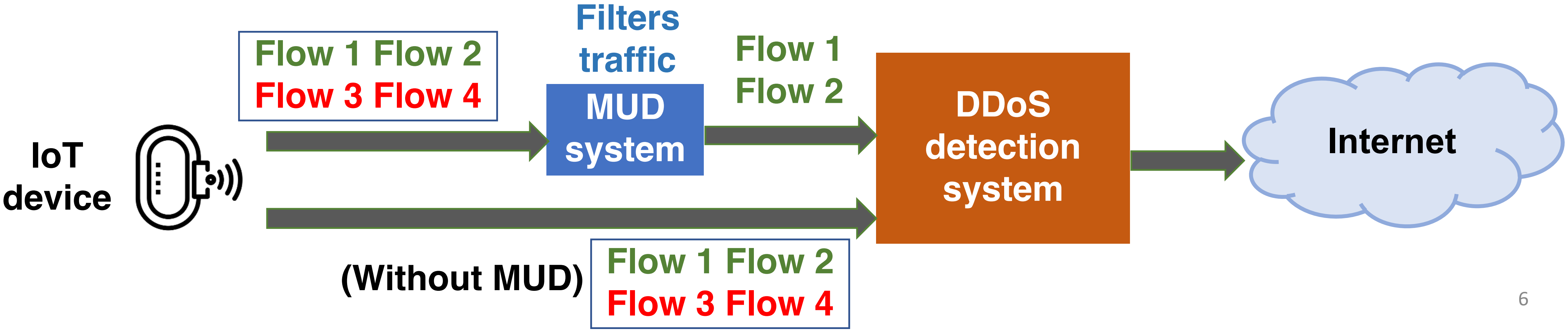
- Ease of management
  - Managing many CPEs vs few switches at the edge
  - Heterogeneity across CPEs is complex to handle

- Reduces overheads of the existing security infrastructure
  - Ex: DDoS detection systems, Deep packet inspection

How?

Overhead reduction of DDoS system

- Without MUD
  - The whole traffic is incident on the DDoS system
- With MUD
  - Consider that MUD blocks non-compliant traffic
  - DDoS system monitors only MUD compliant traffic
  - Reduction in DDoS system overheads (memory, processing)



Existing works that enforce MUD

Clear as MUD [IoT S&P’18] **I** **Combining MUD policies for IDS** [IoT S&P’18]

Volumetric attack detection using MUD [SOSR’19] **I** **SoftMUD** [NIST, ICN]

On premise MUD enforcement

X **Fragmented across multiple LANs, thus hard to manage**

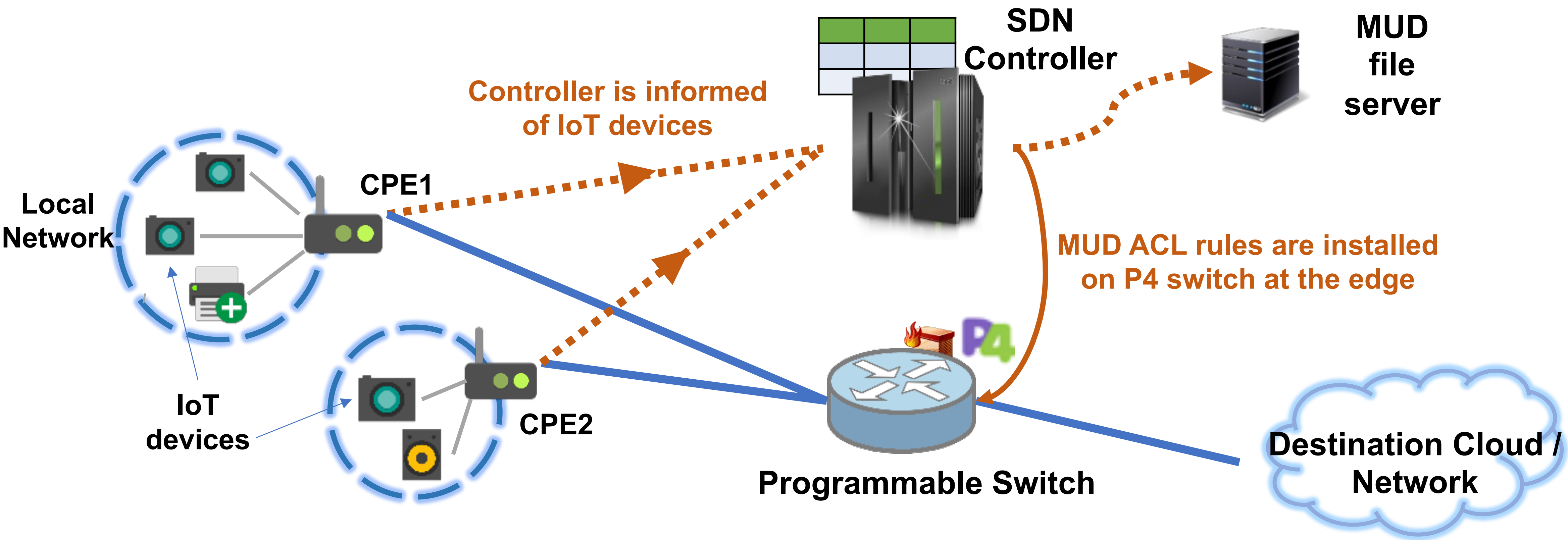
IoT security at ISP using NFV [NOMS’20]

X **Invokes control plane for every new flow from each IoT device**

X **High resource overhead (processing and bandwidth)**

Key idea: Leverage features of P4-based Programmable data planes at the network edge

Easy to manage and scales well


SDN Controller
MUD file server
Controller is informed of IoT devices
Local Network
CPE1
CPE2
IoT devices
MUD ACL rules are installed on P4 switch at the edge
P4
Programmable Switch
Destination Cloud / Network

However, there are few questions to be answered

Questions to be addressed

- How to map an IoT device to its corresponding MUD?
  - Issue: MAC masking, NATing
- How to enforce MUD on reverse traffic (backward)?
  - Issue: Destinations do not mark the traffic
- How to scale to a large number of IoT devices?
  - Issue: Switch has limited memory resources

Use packet marking to identify IoT device in forward direction

Remember forward connections and perform lookup on it for reverse traffic

Use space-efficient decision tree-based data structure to maintain MUD rules

- How to map an IoT device to its corresponding MUD?
  - Issue: MAC masking, NATing
- How to enforce MUD on reverse traffic (backward)?
  - Issue: Destinations do not mark the traffic
- How to scale to a large number of IoT devices?
  - Issue: Switch has limited memory resources

Use packet marking to identify IoT device in forward direction

Remember forward connections and perform lookup on it for reverse traffic

Use space-efficient decision tree-based data structure to maintain MUD rules

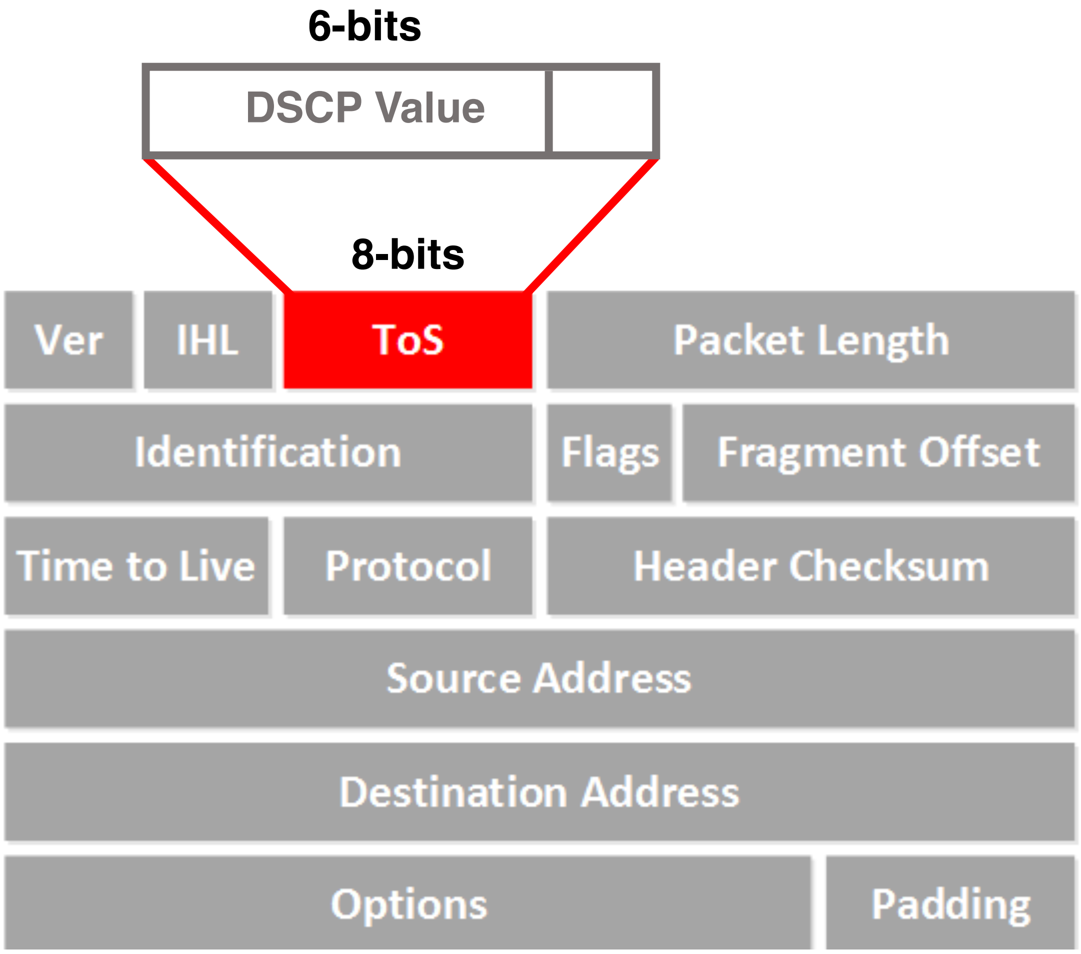
IoT device identification

- MAC address of IoT device is not visible at the edge

  Solution: Use DHCP discover packets to inform the SDN controller

- IoT device type information is not available at the edge

  Solution: Instruct CPE to mark IoT traffic using the 6-bit DSCP value[1]

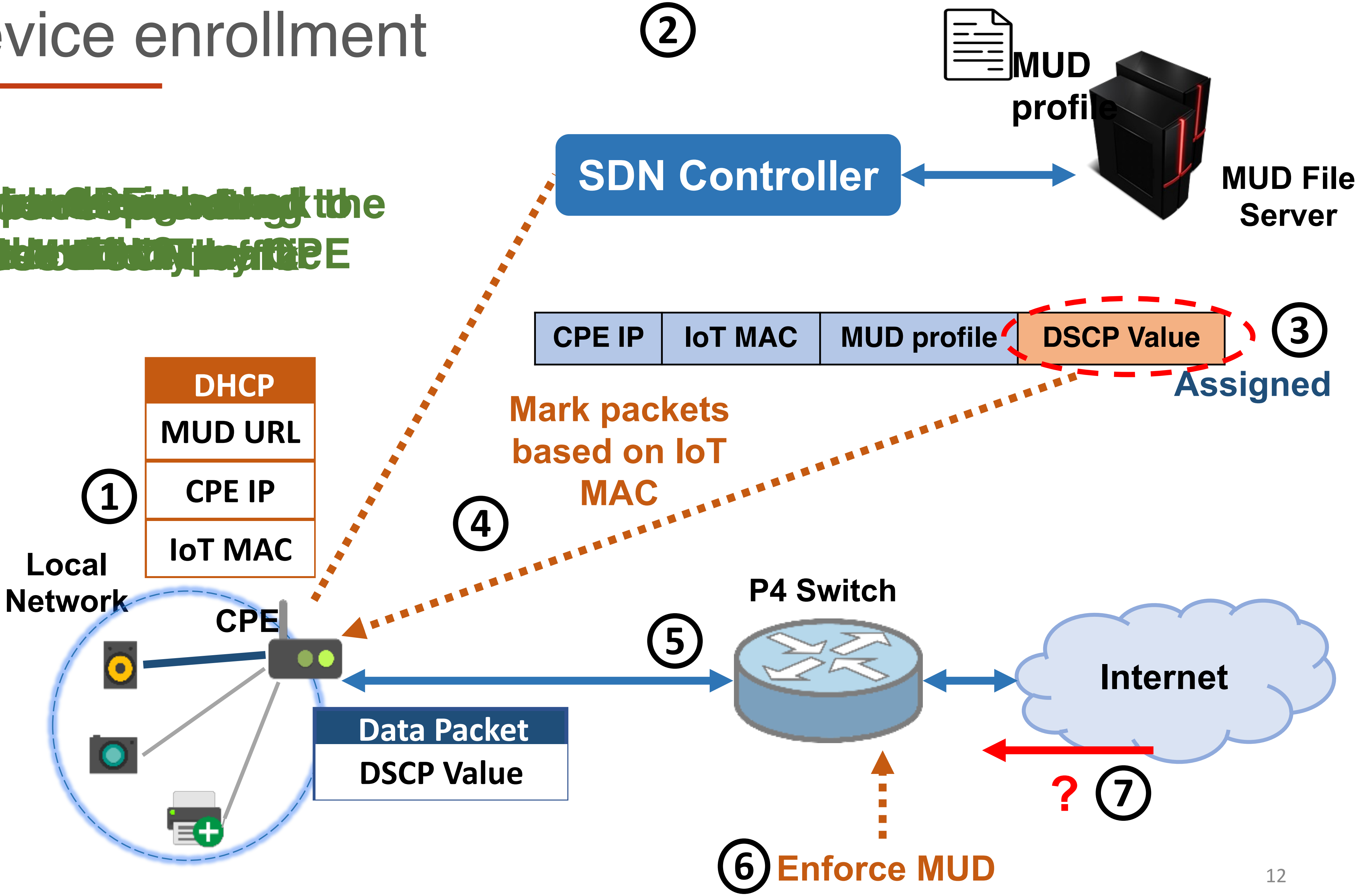


IP Header

[1]TR-069 Remote management protocol

IoT device enrollment

[illegible]



- How to map an IoT device to its corresponding MUD?
  - Issue: MAC masking, NATing

- How to enforce MUD on reverse traffic (backward)?
  - Issue: Destinations do not mark the traffic

- How to scale to a large number of IoT devices?
  - Issue: Switch has limited memory resources

Use packet marking to identify IoT device in forward direction

Remember forward connections and perform lookup on it for reverse traffic

Use space-efficient decision tree-based data structure to maintain MUD rules

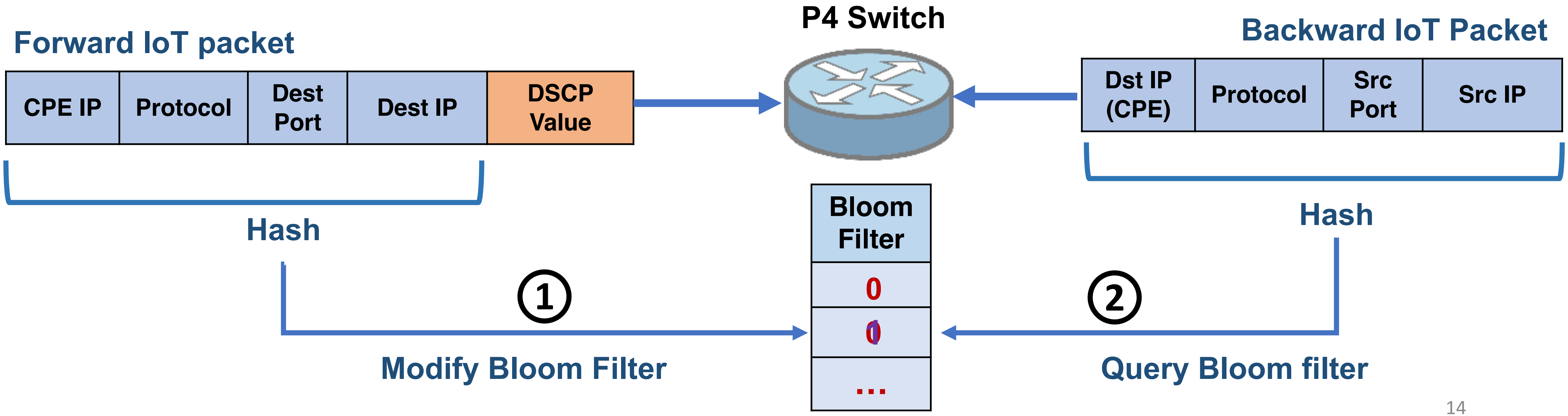
MUD enforcement on backward traffic

- DSCP mark is lost in the backward traffic

  Solution: Keep track of forward IoT traffic in a bloom filter.

① Header fields are hashed and recorded in the bloom filter

② The backward packet headers are queried at the bloom filter



P4 Switch

Forward IoT packet

CPE IP	Protocol	Dest Port	Dest IP	DSCP Value

Backward IoT Packet

Dst IP (CPE)	Protocol	Src Port	Src IP

- How to map an IoT device to its corresponding MUD?
    - Issue: MAC masking, NATing

- How to enforce MUD on reverse traffic (backward)?
    - Issue: Destinations do not mark the traffic

- How to scale to a large number of IoT devices?
    - Issue: Switch has limited memory resources

Use packet marking to identify IoT device in forward direction

Remember forward connections and perform lookup on it for reverse traffic

Use space-efficient decision tree-based data structure to maintain MUD rules

Scaling MUD rules at the switch

Two types of switch memory

- **TCAM**
  - Enables fast parallel search, but the size is small
  - Used by default for MUD ACL rules with wildcards (*)

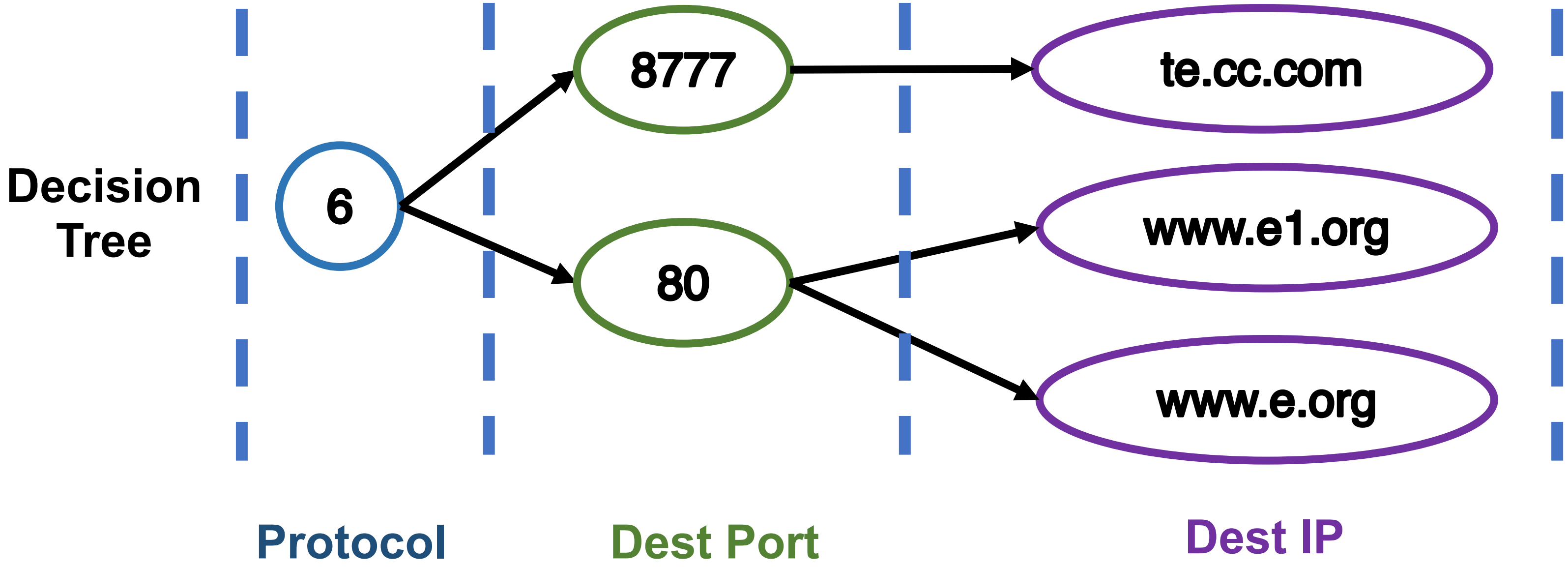
- **SRAM**
  - Relatively abundant (100’s of MBs)
  - Supports exact matches

Solution: Use SRAM-based packet classification algorithm

Decision tree-based representation

Observation: MUD-based ACLs have repeating values

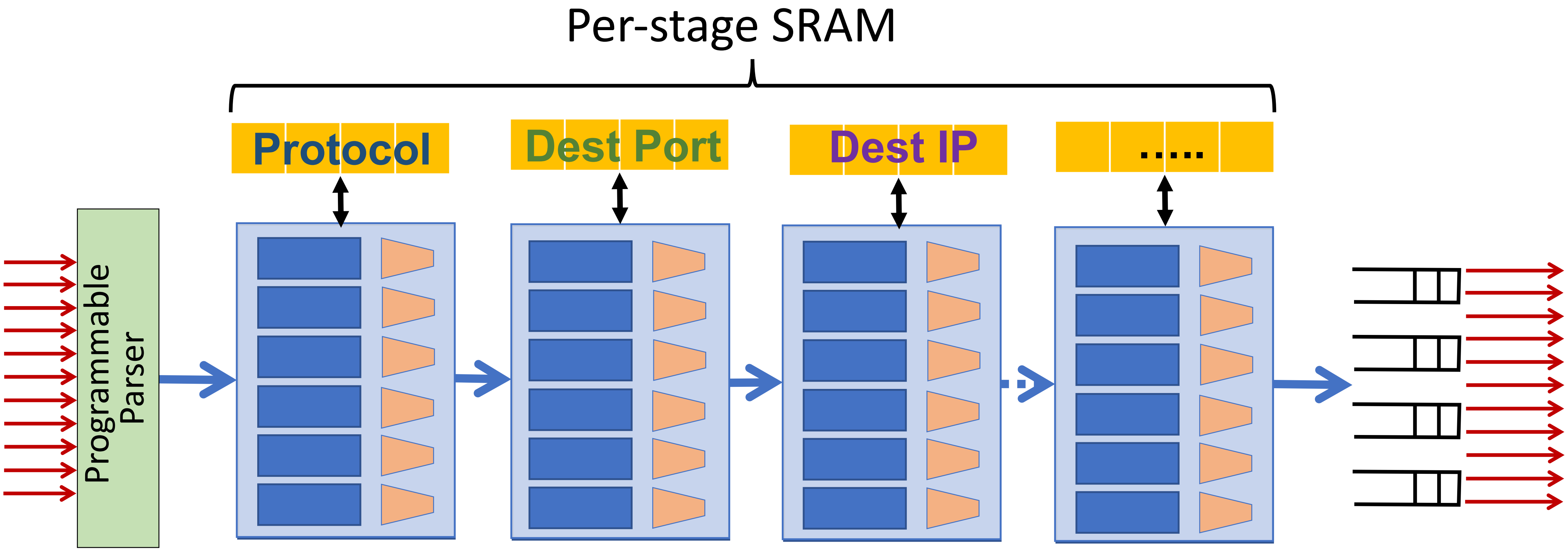
Rule No.	typeEth	protocol	sPort	dPort	srcIP	dstIP
1	0x0800	6	*	8777	*	te.cc.com
2	0x0800	6	*	80	*	www.e.org
3	0x0800	6	*	80	*	www.e1.org
:		:	:	:	:	:



Encode DT using a match-action table at the switch

Decision tree in switch match action table

- Each pipeline stage has some allocated SRAM
- Each decision tree layer can be mapped to a stage



Match + Action pipeline stages

Future Work

- Using DSCP limits support to only 41 IoT device types per CPE

  Alternative: Better packet marking alternative with CPE support

- Attackers could send spurious MUD URL requests to the controller

  Prevention: Explore certificate-based authentication mechanisms like X.509

- Implementation on real testbed

Conclusion

- A system design for MUD enforcement at the network edge

- **Key benefits:**
  - Easy to manage different types of local networks
  - Reduces resource overheads on the existing security infrastructure

- **Key ideas:**
  - Use packet marking capabilities of CPEs to identify IoT device
  - Use programmable switch features to scale well