
FRAUNHOFER FKIE & THALES DEUTSCHLAND
DEPT. OF COMMUNICATION SYSTEMS & THALES SIX

Research Group: **Robust Heterogeneous Networks**

**TOWARDS A CYBER DEFENSE SYSTEM IN
SOFTWARE-DEFINED TACTICAL NETWORKS**

Sean Kloth, Paulo H. L. Rettore, Philipp Zißner, Bruno P. Santos, and Peter Sevenich

Agenda

- Background
 - Motivation
 - Problem Definition & Proposed Solution
- Methodology
 - Cyber Attack Agent (CAA)
 - Cyber Defence Agent (CDA)
- Evaluation
- Limitations
- Conclusion

Background – Motivation

- **Tactical Networks (TNs) face challenges** due to the **heterogeneous communications, limited radio links, mobility, and cybersecurity threats**
- Software-defined Networks
 - **potential to host mechanisms to control the network, reducing cost and management overhead**
- However, standard SDN protocols, like Open-Flow, were designed for:
 - **non-mobile, reliable, high-speed, and low-latency** networks



Background – Problem Definition

- How does **SDN**, OpenFlow deal with **Distributed-Denial-of-Service-attacks (DDOS)** in **Tactical Networks**?
- Which problems do **Tactical Networks** introduce to a **DDOS attack**?
- How can we **detect an attack early**?
- Which **countermeasures** can be taken?

Background – Proposed Solution

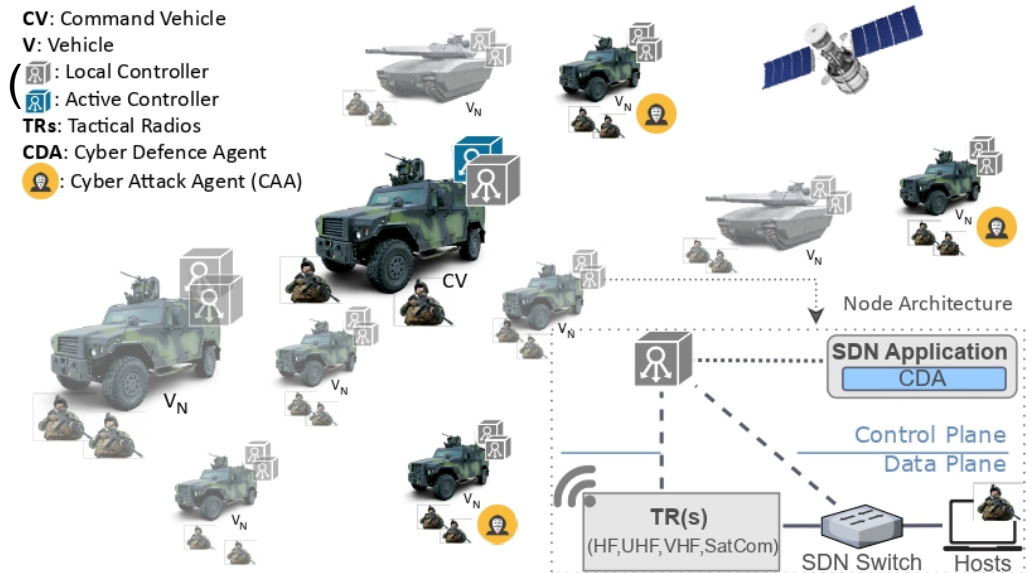
■ Resilient Controller

→ Introduce a system to **detect** and **react** to a DDOS attack

- Cyber Attack Agent (CA/ 
 - Creates DDOS attack challenging CDA
 - Data plane and Control plane
- Cyber Defence Agent (CD/ 
 - Monitor and detect attacks through features
 - React to attack

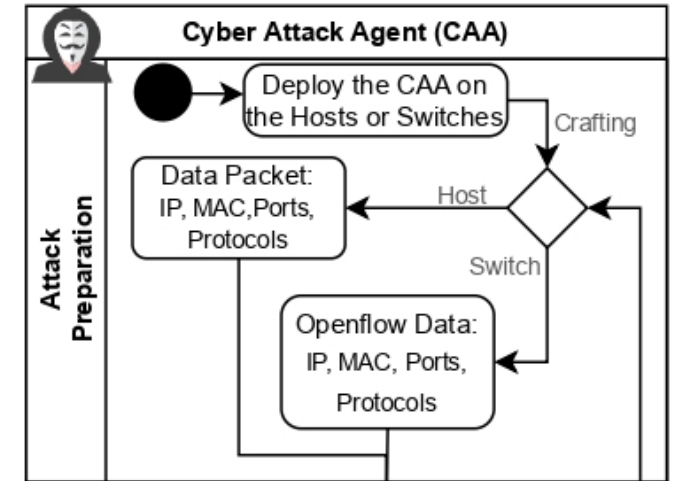
Methodolgy

- **Resilient Controller:** Ability to with-stand and respond to attacks
- Network scenario:
 - Cluster of vehicles (V) controlled by command vehicle
 - Active connection via 2MBit SatCom links with latency of 2 seconds
 - Each vehicle minimum of two local controllers
 - With at least two connected hosts
 - CDA executed in active controller
 - CAA deployed randomly across the network topology



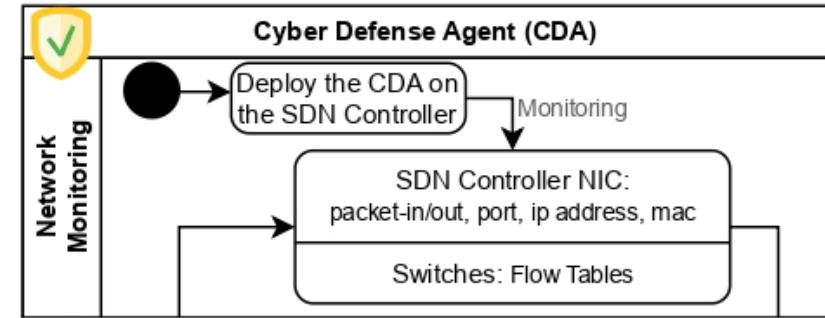
Methodology – Cyber Attack Agent (CAA)

- Create a DDOS attack by flooding the controller with **packet-in requests**
 - Craft packets that force a miss, causing controller to respond
 - Packet-in requests only packets sent to controller
- Data Plane:
 - Force switch to send **packet-in requests** by modifying IP, MAC addresses and ports for source and destination
- Control Plane:
 - Sent **packet-in requests directly** from a switch



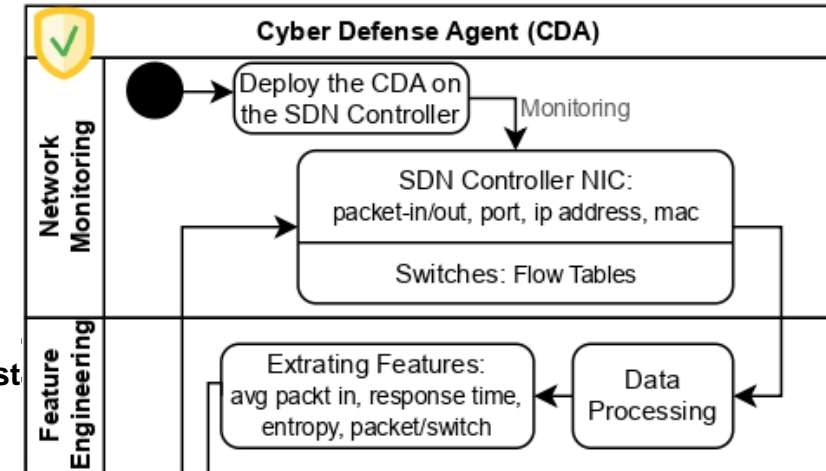
Methodology – Cyber Defence Agent (CDA)

- **Monitor** incoming traffic:
 - Collect all relevant information from the IP traffic using a packet sniffer
 - Process information as fast as possible



Methodolgy – CDA Features

- **Collect features:**
 - Entropy:
 - **IP address, ports for source and destination**
 - $H(x) = -\sum p(x)\log(p(x))$, $\mathbf{p(x)} = \{\mathbf{IP}_{src/dest}, \mathbf{Port}_{src/dest}\}$



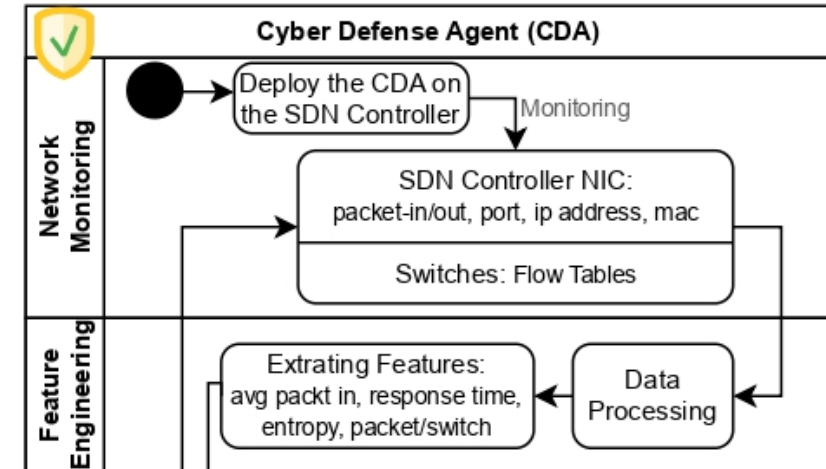
Methodology – CDA Features

- **Collect features:**

- Entropy

- Average number of packet-in requests:

- $$Pin_{avg} = \frac{\sum_{i=1}^{Number\ of\ ports} Number\ of\ packet-in\ on\ port\ i}{Time\ interval}$$



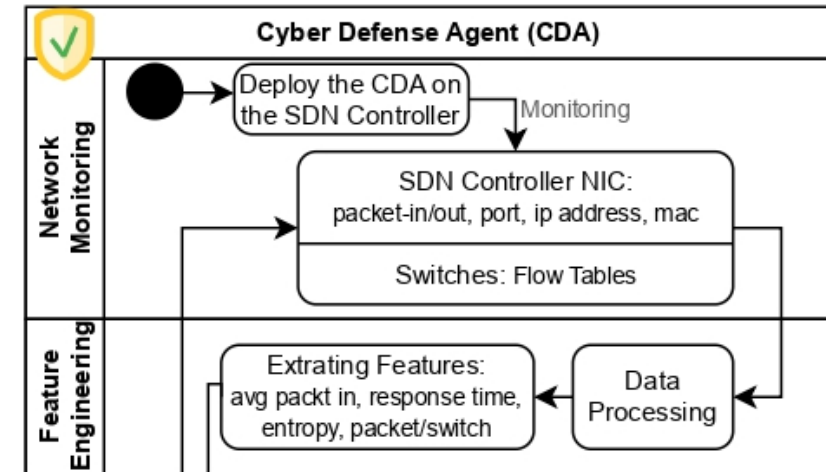
Methodolgy – CDA Features

- **Collect features:**

- Entropy
- Average number of packet-in requests

- Average response time:

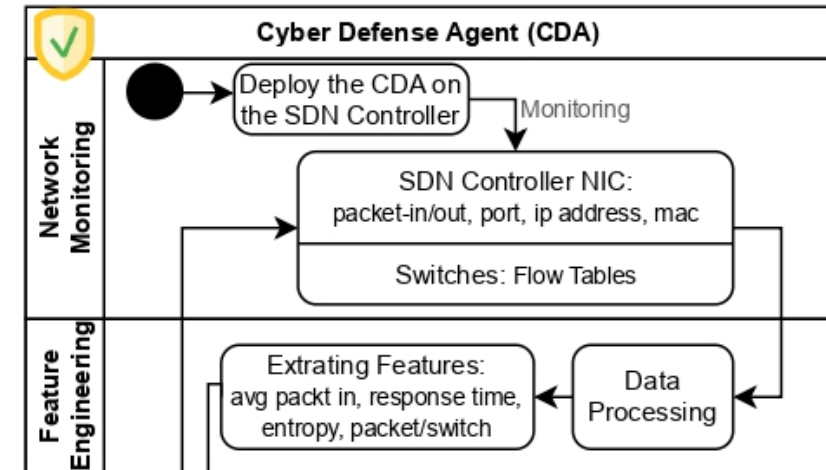
- $$Rep_{avg} = \frac{\sum_{i=1}^{Number\ of\ requests} (request\ time\ i) - (response\ time\ i)}{Time\ interval}$$



Methodolgy – CDA Features

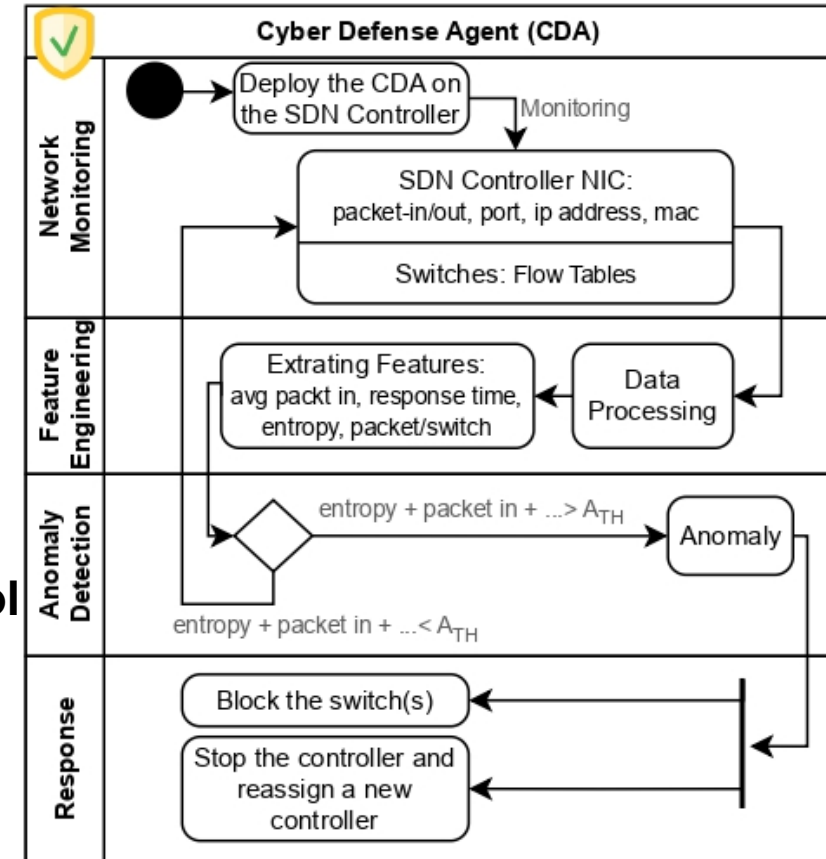
■ Collect features:

- Entropy
- Average number of packet-in requests
- Average response time
- Identification of compromised switches:
 - $Pkt_{switch} = \sum_{i=1}^{Time\ interval} i, i: Pkts\ to\ Controller$



Methodolgy – CDA Features

- **Detect** anomalies:
 - Use features and define **threshold**
 - **Alternative:** Use machine learning model
- **React:**
 - **Block** compromised switch(es)
 - **Reassign** non-compromised switches to **backup control**

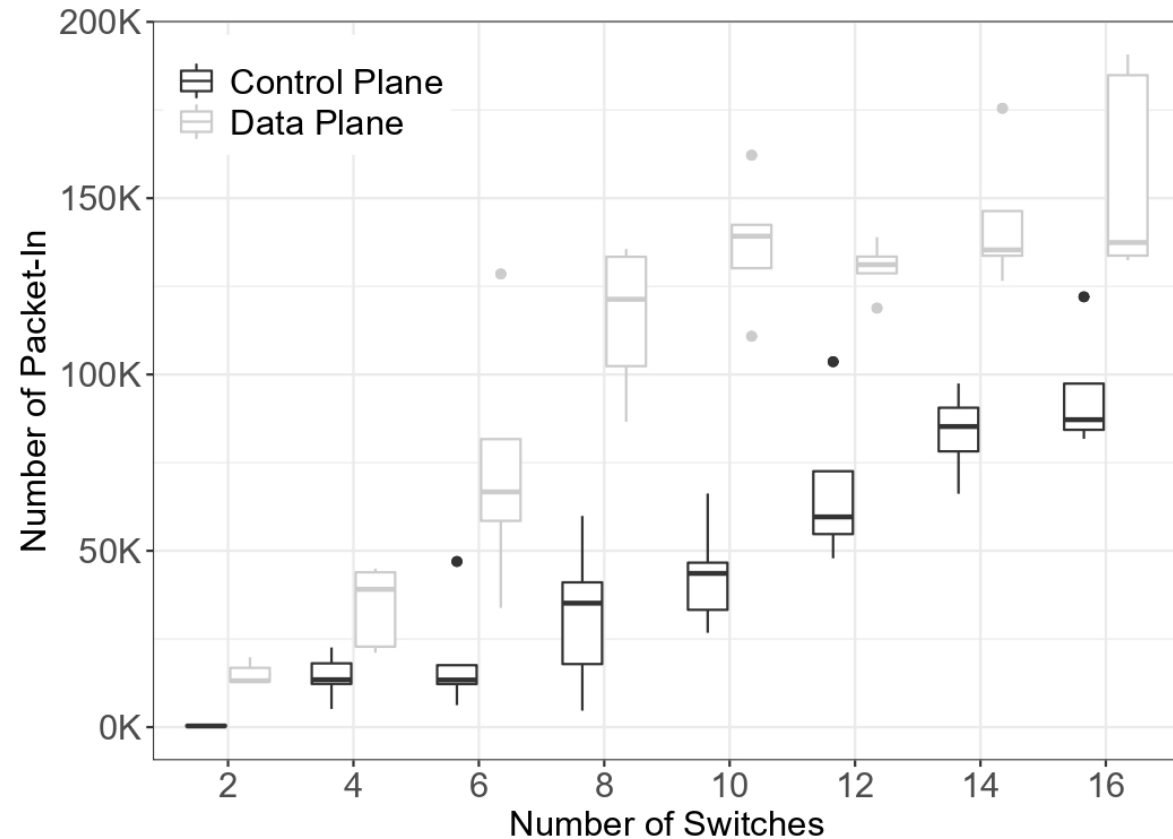


Evaluation – General Settings

- Topology:
 - **10** vehicles, with each **2** Soldiers
 - **Linear** connection between vehicles
 - **Controller**: Ryu
 - Traffic: **TCPreplay**, with **UDP** and **TCP**
- Connections:
 - Vehicles - Soldiers : 2MBit/s, delay 2 seconds
 - Vehicle – Vehicle: 2MBit/s, delay 2 seconds

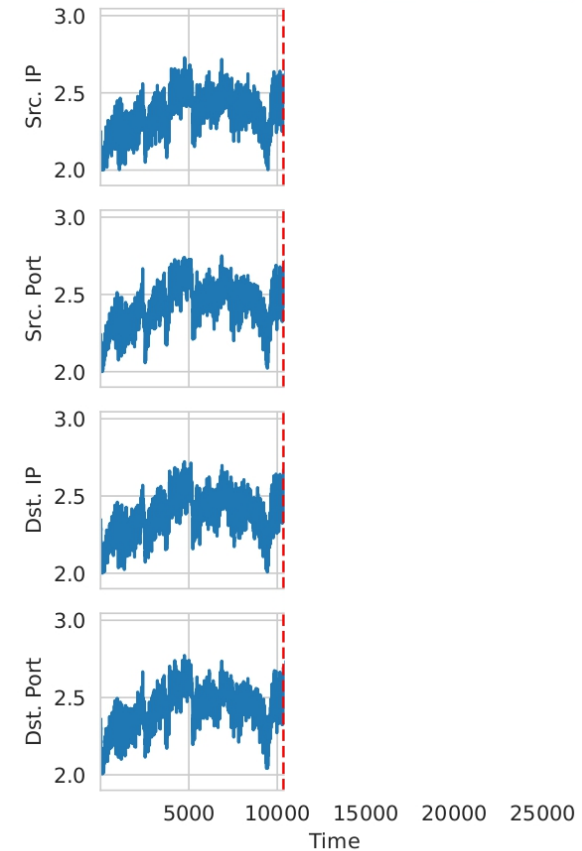
Evaluation – Data Plane vs. Control Plane

- Inject a total of 1.000.000 packets, compare resulting packet-in requests
- Control Plane:
 - Performs **worse** due to encapsulation
- Data Plane:
 - **Amplification effect** observable
 - Conversion of packets **higher**
- Compromising hosts is **sufficient**



Evaluation – Features

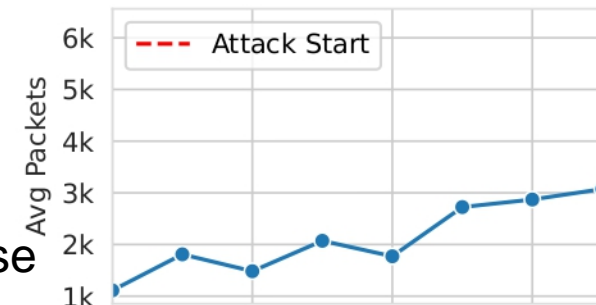
- Entropy:
 - Idle: 2-2.7
 - Attack: 2.5-3, trending towards 3.5
 - Clear difference between Idle and Attack phase



Evaluation – Features

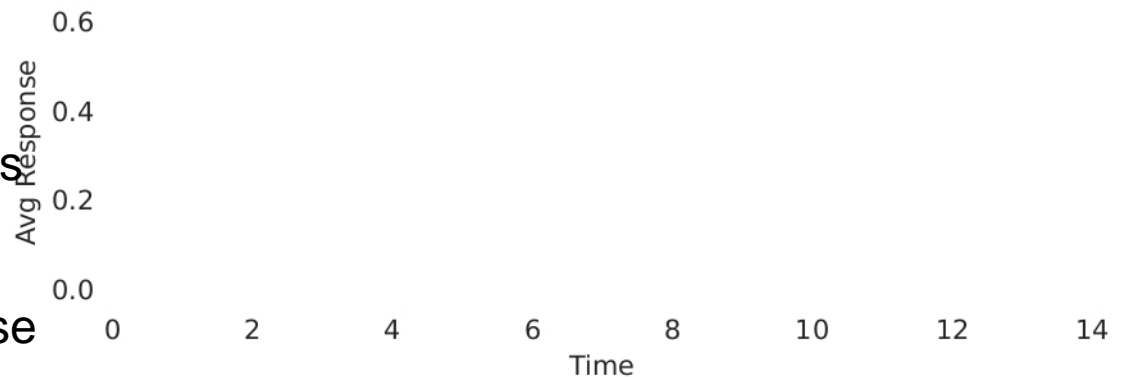
■ Average packet-in requests:

- Idle: 1000 and 3000
- Attack: 6000
- Clear difference between Idle and Attack phase



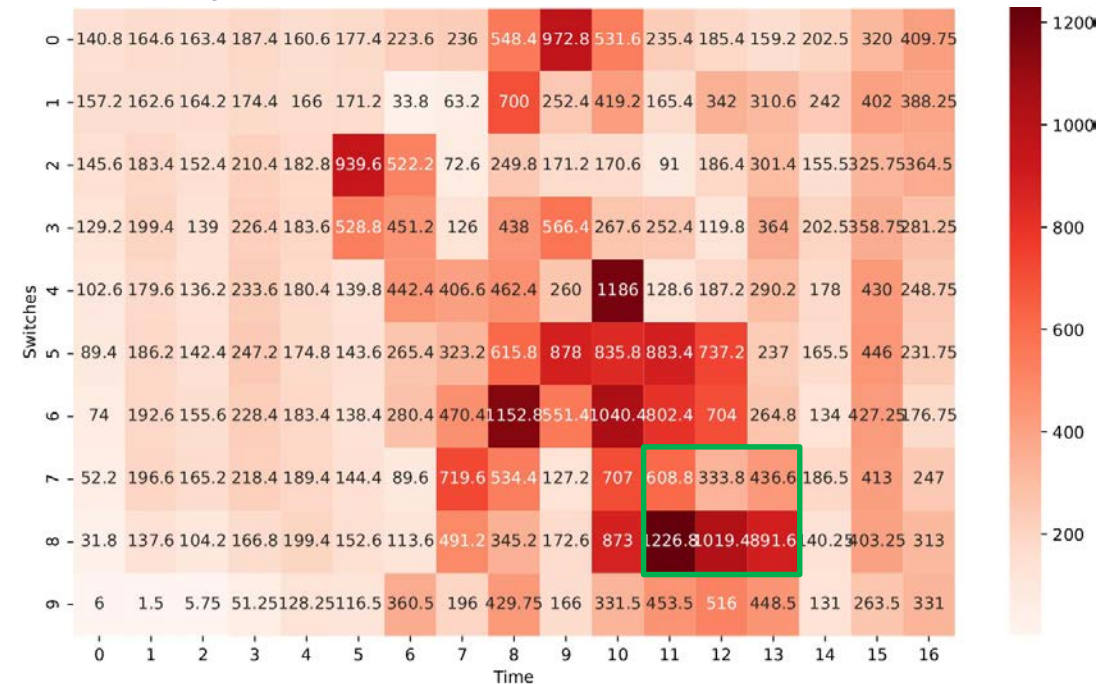
■ Average response time:

- Idle: Response almost immediate, 0.1 seconds
- Attack: increase of 0.6, with average of 0.4
- Clear difference between Idle and Attack phase



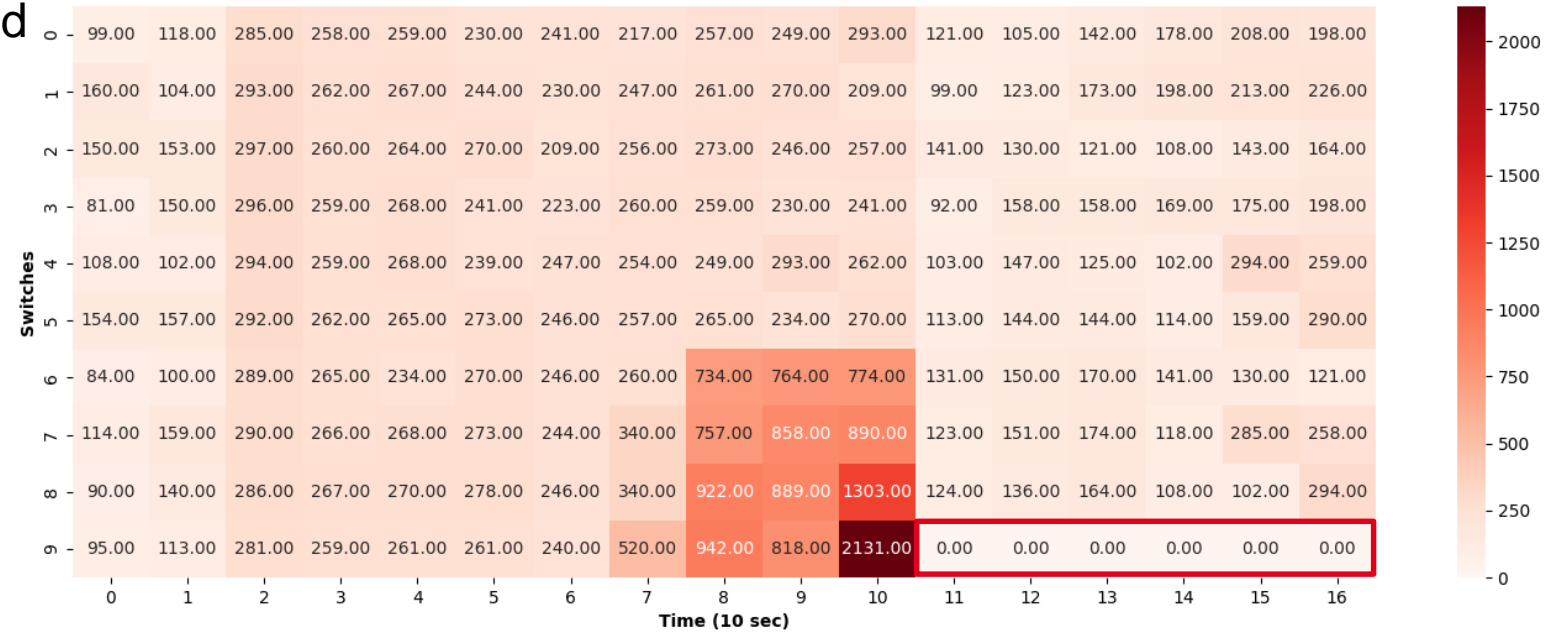
Evaluation – Features

- Identification of compromised switches:
 - During the idle phase, all switches show similar behaviour
 - During the attack, compromised switches send **drastically** more
 - Neighbouring switches send more
 - **Amplification effect**



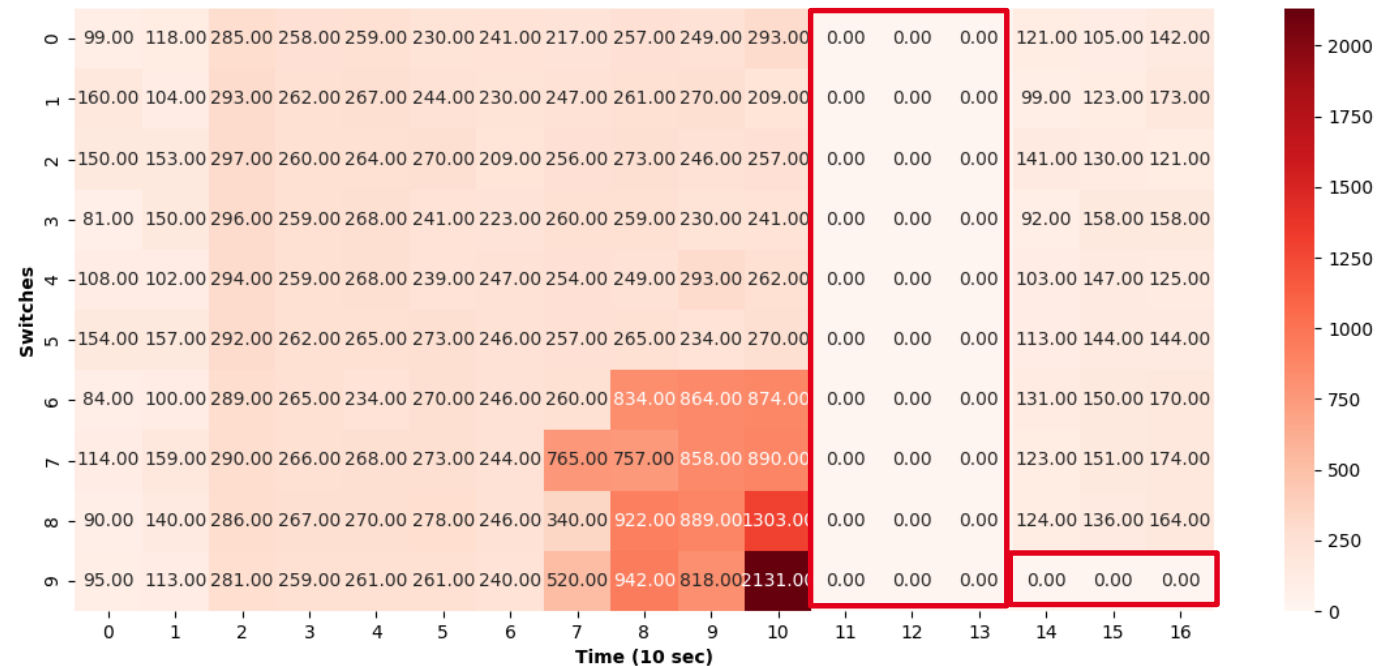
Evaluation – Response Mechanism

- **Block** compromised switch
 - **Keep** network structure
 - **Remove** switch/install flow table entry **dropping** every packet
 - Normal traffic **barely** affected



Evaluation – Response Mechanism

- **Replace controller**
 - **Rebuild** network with only **uncompromised** switches and **backup** controller
 - Compromised switches connected to old controller
 - All communication is **interrupted**



Limitations

- Virtual machine and Mininet environment:
 - Hardware limitations cause distortion
 - Switches are bottlenecks
- Defensive mechanism:
 - Requires threshold

Conclusion

- Threshold-based detection
 - Can detect DDOS
 - Two different reactions: **block** port, **replace** controller
 - **Too** domain specific

- Future Work:
 - Machine Learning based approach

Thank you for your
attention!