DDoS-AID: Automated In-Network DDoS Mitigation as a First Line of Defense

Albert Gran Alcoz(1), Martin Strohmeier(2), Vincent Lenders(2), Laurent Vanbever(1)

Cyber-Alp Retreat

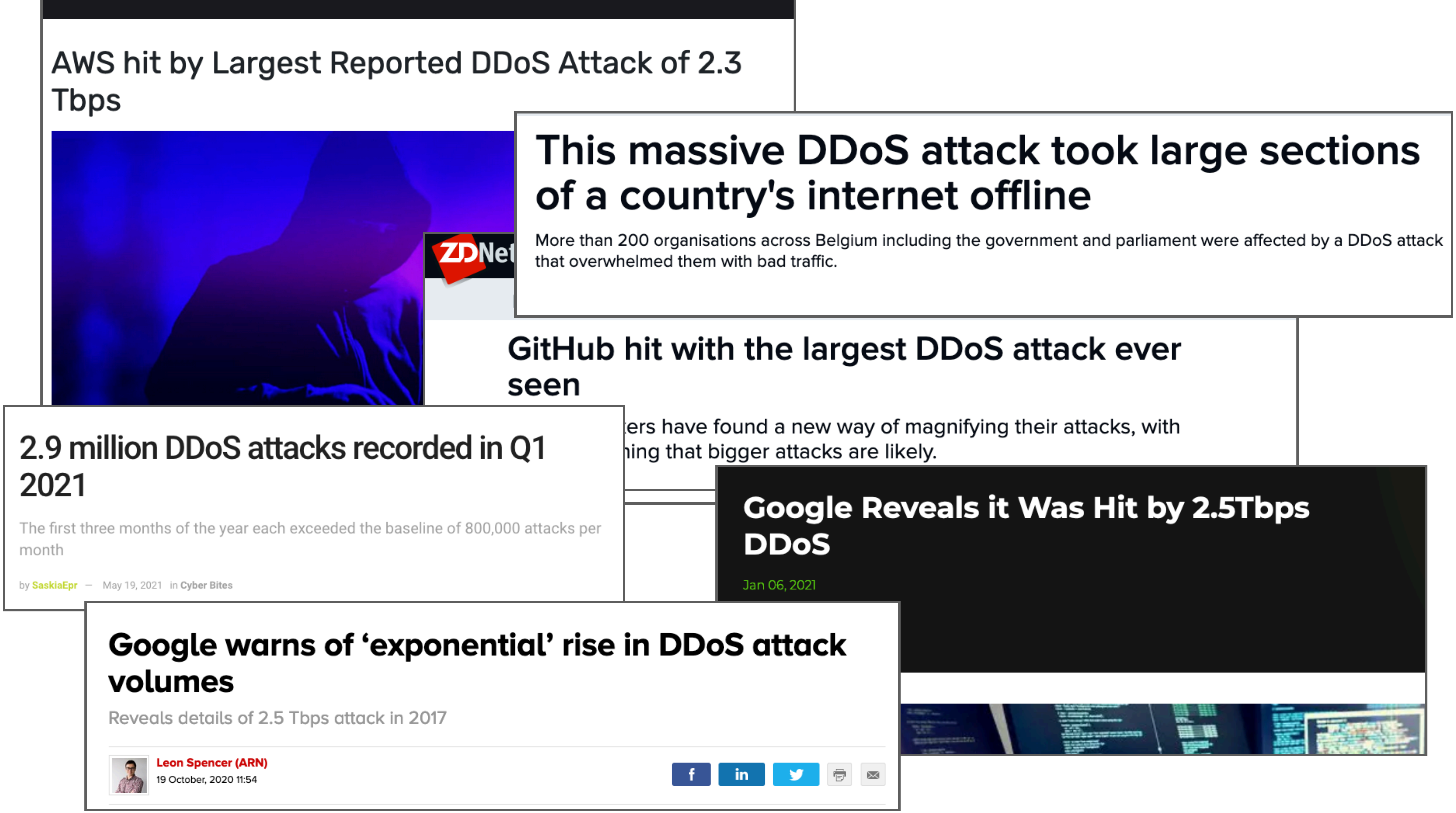
July, 01 2020

(1) ETH zürich

(2) Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

armasuisse

DDoS Attacks



AWS hit by Largest Reported DDoS Attack of 2.3 Tbps

This massive DDoS attack took large sections of a country's internet offline

More than 200 organisations across Belgium including the government and parliament were affected by a DDoS attack that overwhelmed them with bad traffic.

GitHub hit with the largest DDoS attack ever seen

ers have found a new way of magnifying their attacks, with
ning that bigger attacks are likely.

2.9 million DDoS attacks recorded in Q1 2021

The first three months of the year each exceeded the baseline of 800,000 attacks per month

by SaskiaEpr — May 19, 2021 in Cyber Bites

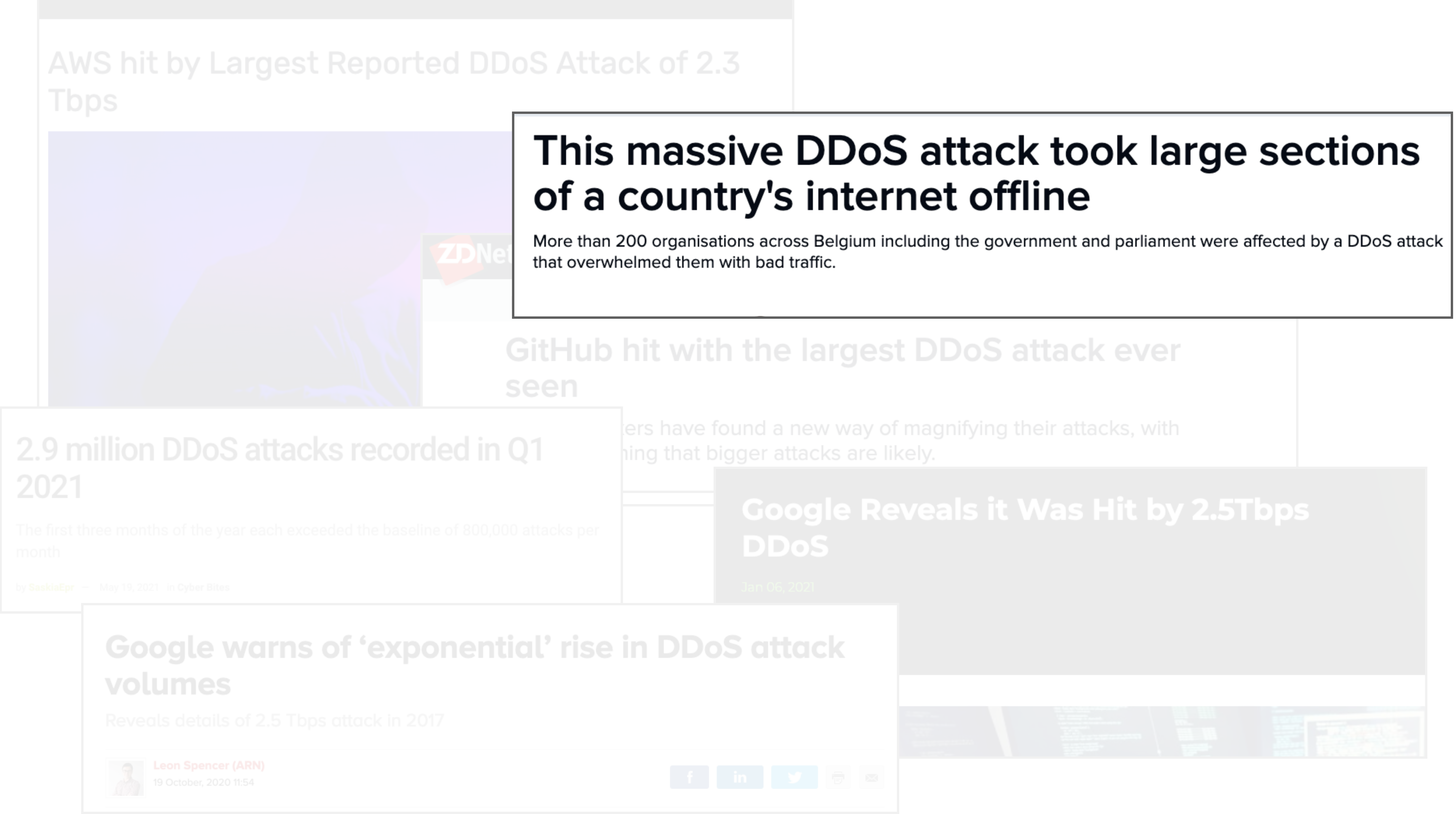
Google Reveals it Was Hit by 2.5Tbps DDoS

Jan 06, 2021

Google warns of ‘exponential’ rise in DDoS attack volumes

Reveals details of 2.5 Tbps attack in 2017

Leon Spencer (ARN)
19 October, 2020 11:54



AWS hit by Largest Reported DDoS Attack of 2.3 Tbps

This massive DDoS attack took large sections of a country's internet offline

More than 200 organisations across Belgium including the government and parliament were affected by a DDoS attack that overwhelmed them with bad traffic.

GitHub hit with the largest DDoS attack ever seen

ers have found a new way of magnifying their attacks, with
ning that bigger attacks are likely.

2.9 million DDoS attacks recorded in Q1 2021

The first three months of the year each exceeded the baseline of 800,000 attacks per month

by SaskiaEpr — May 19, 2021 in Cyber Bites

Google Reveals it Was Hit by 2.5Tbps DDoS

Jan 06, 2021

Google warns of 'exponential' rise in DDoS attack volumes

Reveals details of 2.5 Tbps attack in 2017

Leon Spencer (ARN)
19 October, 2020 11:54

DDoS mitigation is challenging because attacks are constantly moving

Target network infrastructure

DDoS mitigation is challenging because attacks are constantly moving

ProtonMail DDoS Attack, November 2015

Target network infrastructure

“The attacks began to take on an unprecedented level of sophistication

… the attackers began a coordinated assault on our ISP attacking the infrastructure of our upstream providers

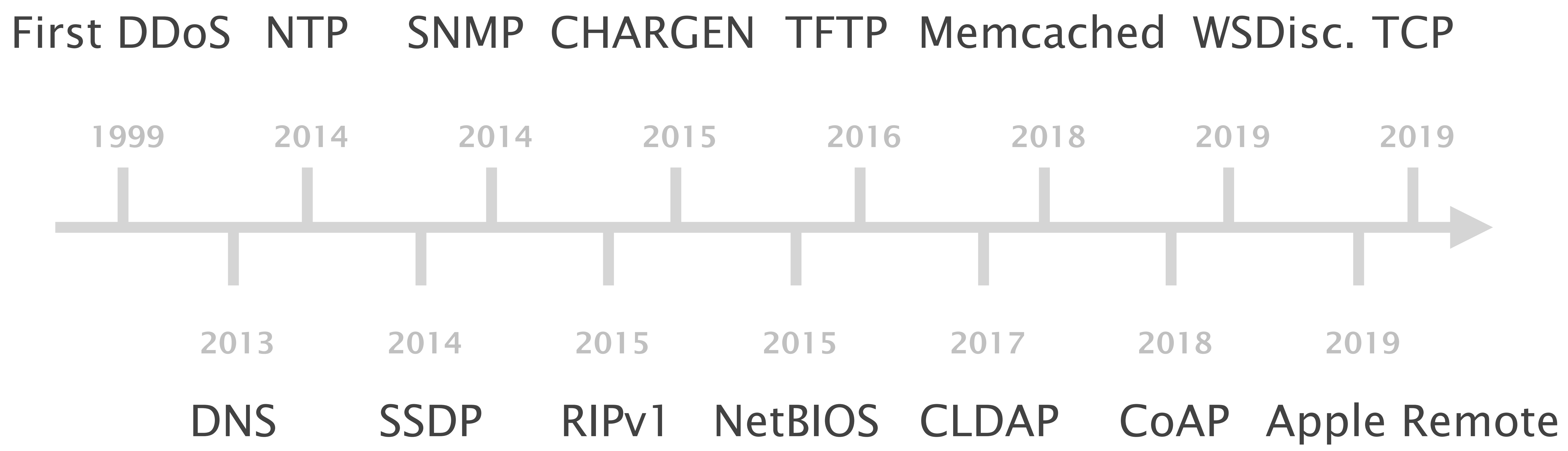
… they attacked routers in Zurich, Frankfurt, and other locations where our ISP has nodes

… managed to bring down both the datacenter and the ISP, impacting hundreds of companies, not just ProtonMail.”

DDoS mitigation is challenging because attacks are constantly moving

Target network infrastructure

New attack vectors



First DDoS 1999

NTP 2014

SNMP 2014

CHARGEN 2015

TFTP 2016

Memcached 2018

WSDisc. 2019

TCP 2019

2013 DNS

2014 SSDP

2015 RIPv1

2015 NetBIOS

2017 CLDAP

2018 CoAP

2019 Apple Remote

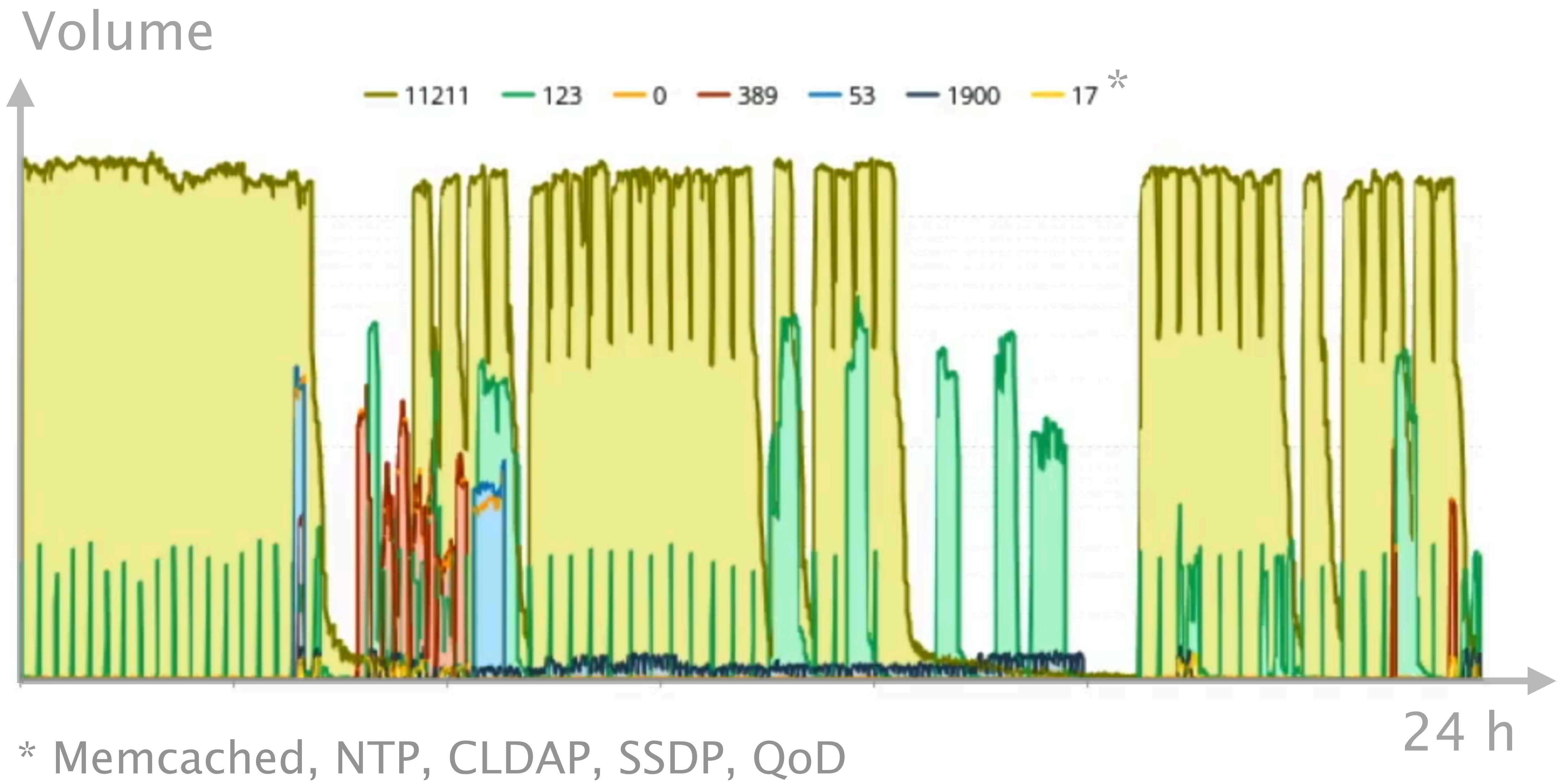
DDoS mitigation is challenging because attacks are constantly moving

Target network infrastructure

New attack vectors

Morph attacks over time

DDoS attack against Google, April 2019



* Memcached, NTP, CLDAP, SSDP, QoD

DDoS mitigation is challenging because attacks are constantly moving

Target network infrastructure → In-network defense

New attack vectors → Broad coverage of (new) patterns

Morph attacks over time → Automatically adapt online

DDoS defenses today can be classified by their level of automation

DDoS defenses today can be classified by their level of automation

Fully Manual

Handcrafted ACLs

- ✘ Very slow
- ✘ Cumbersome, prone to human mistakes
- ✘ Requires attack characterization
- ✔ Safe

DDoS defenses today can be classified by their level of automation

Fully Manual		**Semi-automated**	
Handcrafted ACLs		Preconfigured defenses	
✘	Very slow	~	Faster
✘	Cumbersome, prone to human mistakes	✔	Efficient for known attacks
✘	Requires attack characterization	✘	No protection for new attacks
✔	Safe	✔	Safe

DDoS defenses today can be classified by their level of automation

Fully Manual	**Semi-automated**	**Fully-automated**
Handcrafted ACLs	Preconfigured defenses	Unsupervised classification
✘ Very slow	~ Faster	✔ Fastest
✘ Cumbersome, prone to human mistakes	✔ Efficient for known attacks	✔ Mitigates known attacks
✘ Requires attack characterization	✘ No protection for new attacks	✔ Protection for new attacks
✔ Safe	✔ Safe	✘ Risk

DDoS defenses today can be classified by their level of automation

Fully Manual	**Semi-automated**	**Fully-automated**
Handcrafted ACLs	Preconfigured defenses	Unsupervised classification
✘ Very slow	~ Faster	✔ Fastest
✘ Cumbersome, prone to human mistakes	✔ Efficient for known attacks	✔ Mitigates known attacks
✘ Requires attack characterization	✘ No protection for new attacks	✔ Protection for new attacks
✔ Safe	✔ Safe	✔ DDoS-AID

Is it possible to build a ***fully-automated*** in-network DDoS defense

that is ***safe*** (does not hurt production traffic)?

Introducing…

DDoS-AID

A *fully automated, and-yet-safe* in-network DDoS defense

DDoS-AID: Automated In-Network DDoS Mitigation as a First Line of Defense

1 Key insights
How does it work

2 Implementation
How can it be deployed

3 Evaluation
How well does it perform

DDoS-AID: Automated In-Network DDoS Mitigation as a First Line of Defense

1 Key insights

How does it work

2 Implementation

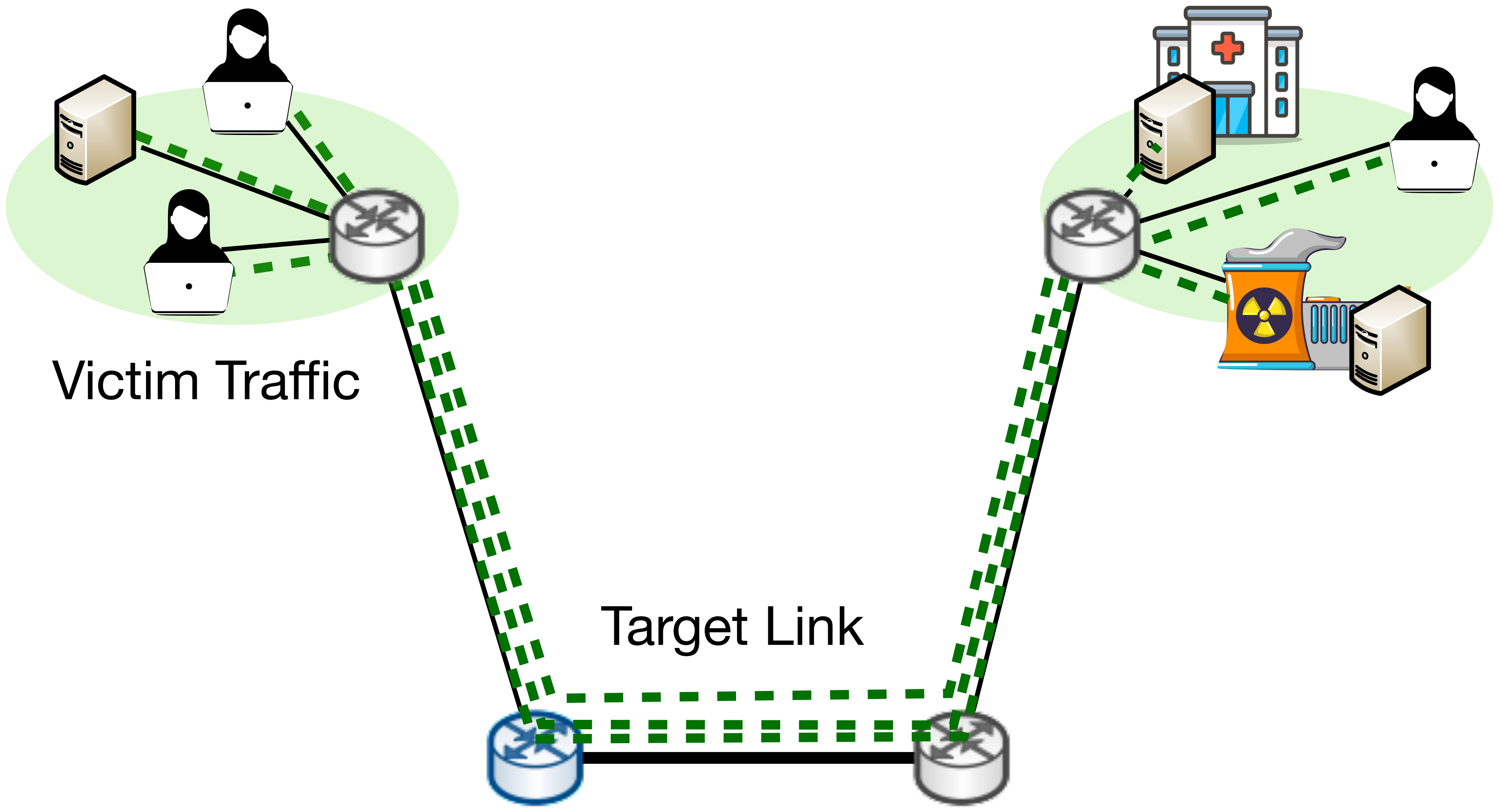
How can it be deployed

3 Evaluation

How well does it perform

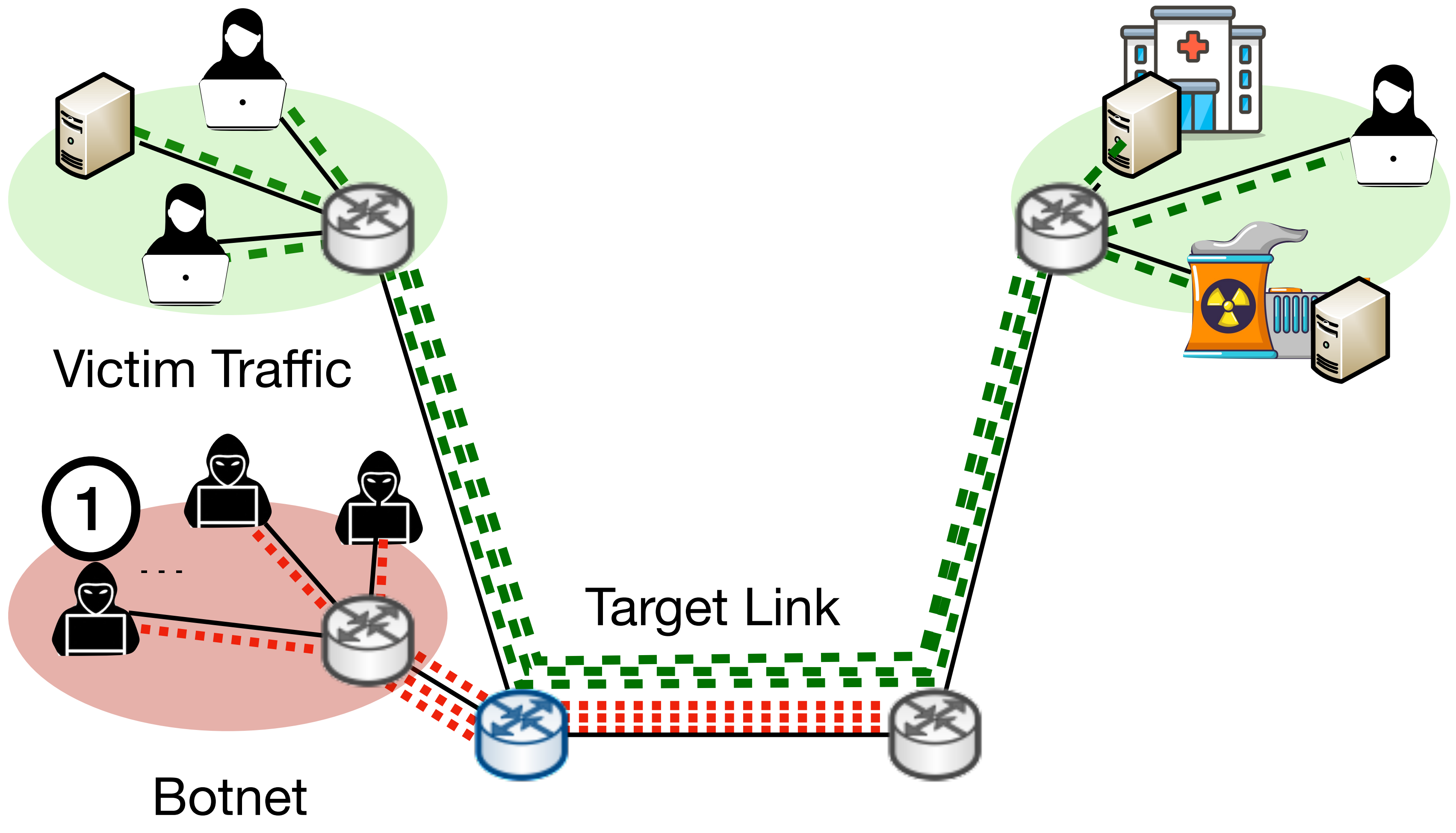
DDoS-AID focuses on volumetric attacks targeting a critical link in the network

Threat model



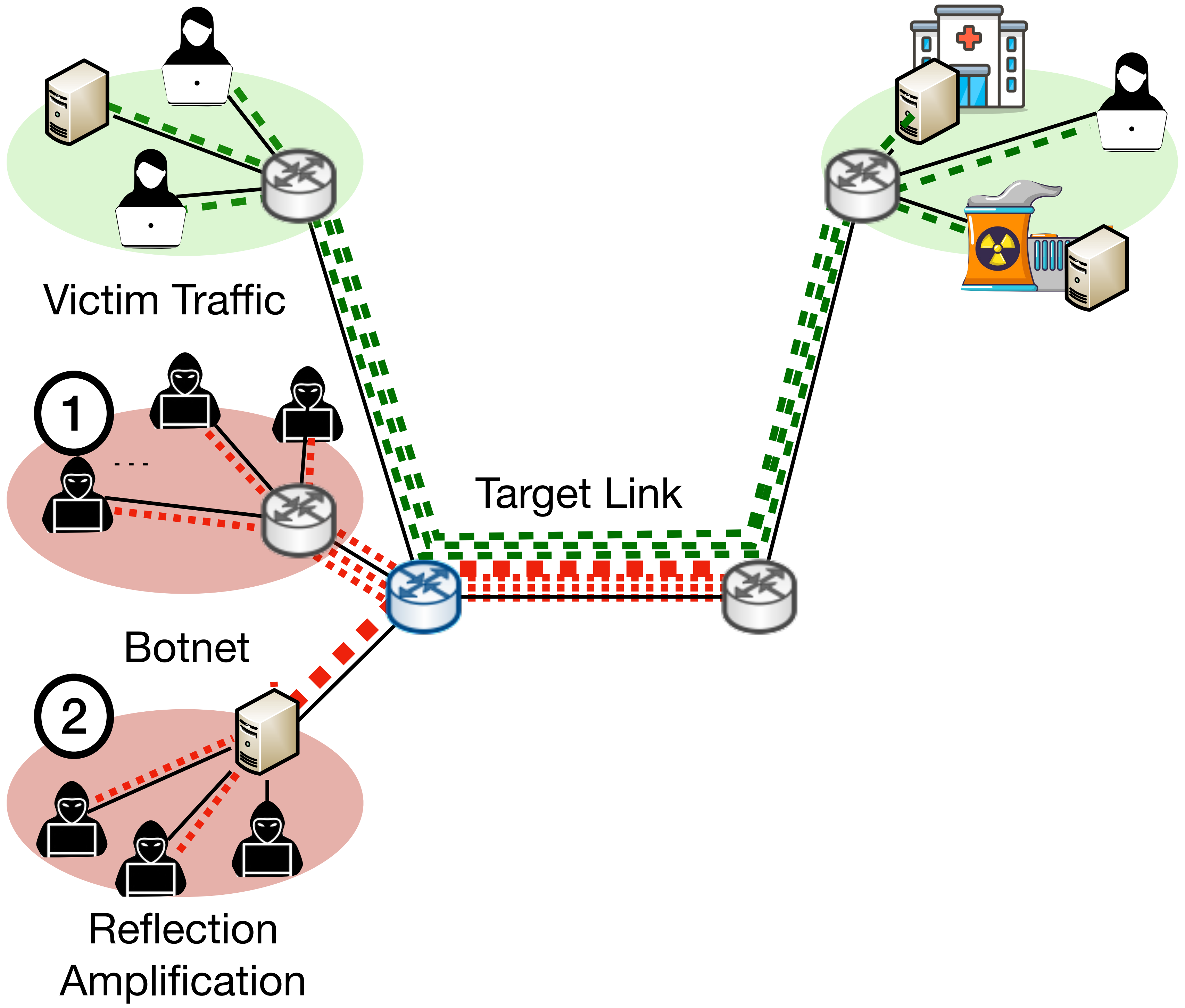
DDoS-AID focuses on volumetric attacks targeting a critical link in the network

Threat model



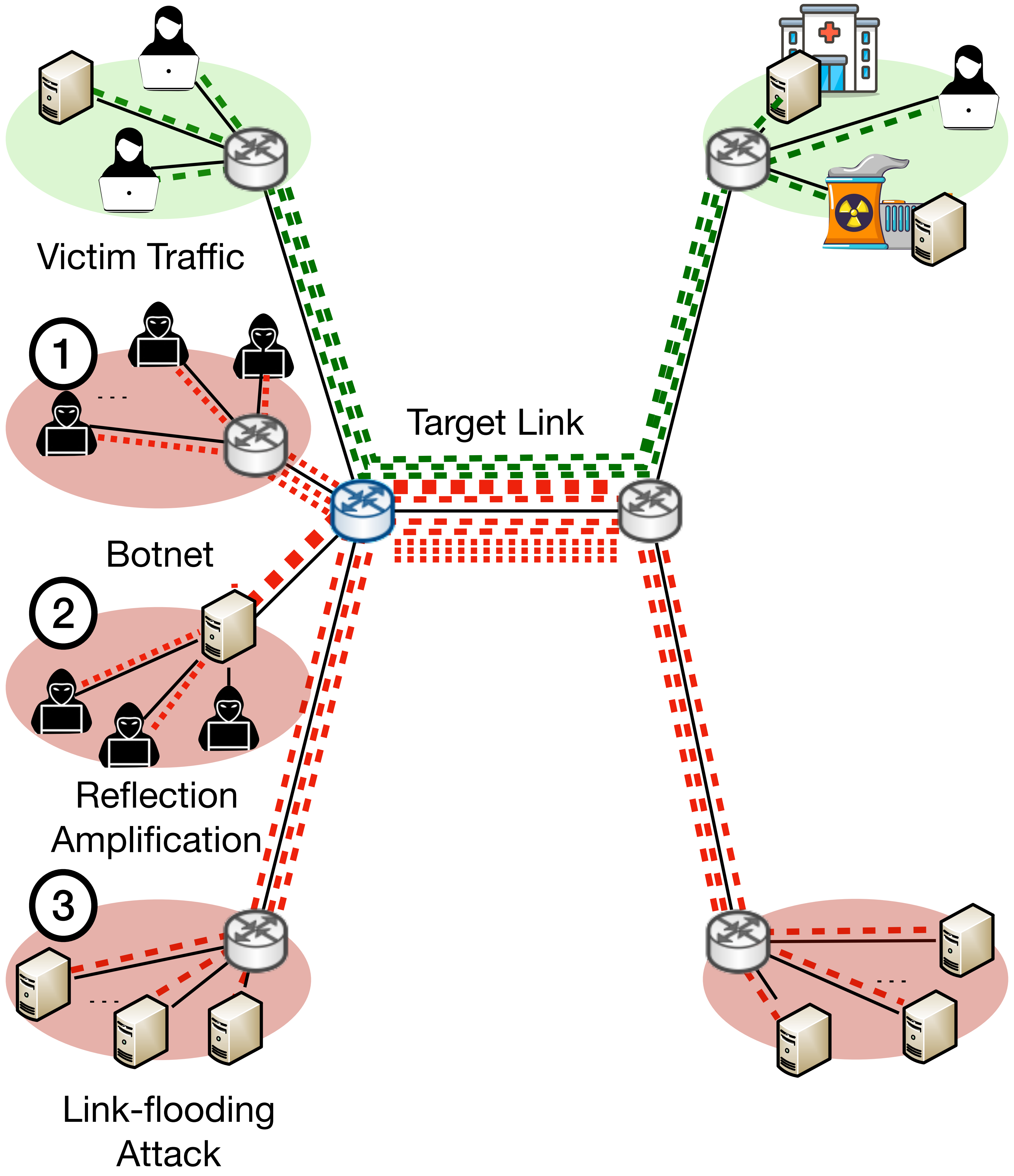
DDoS-AID focuses on volumetric attacks targeting a critical link in the network

Threat model



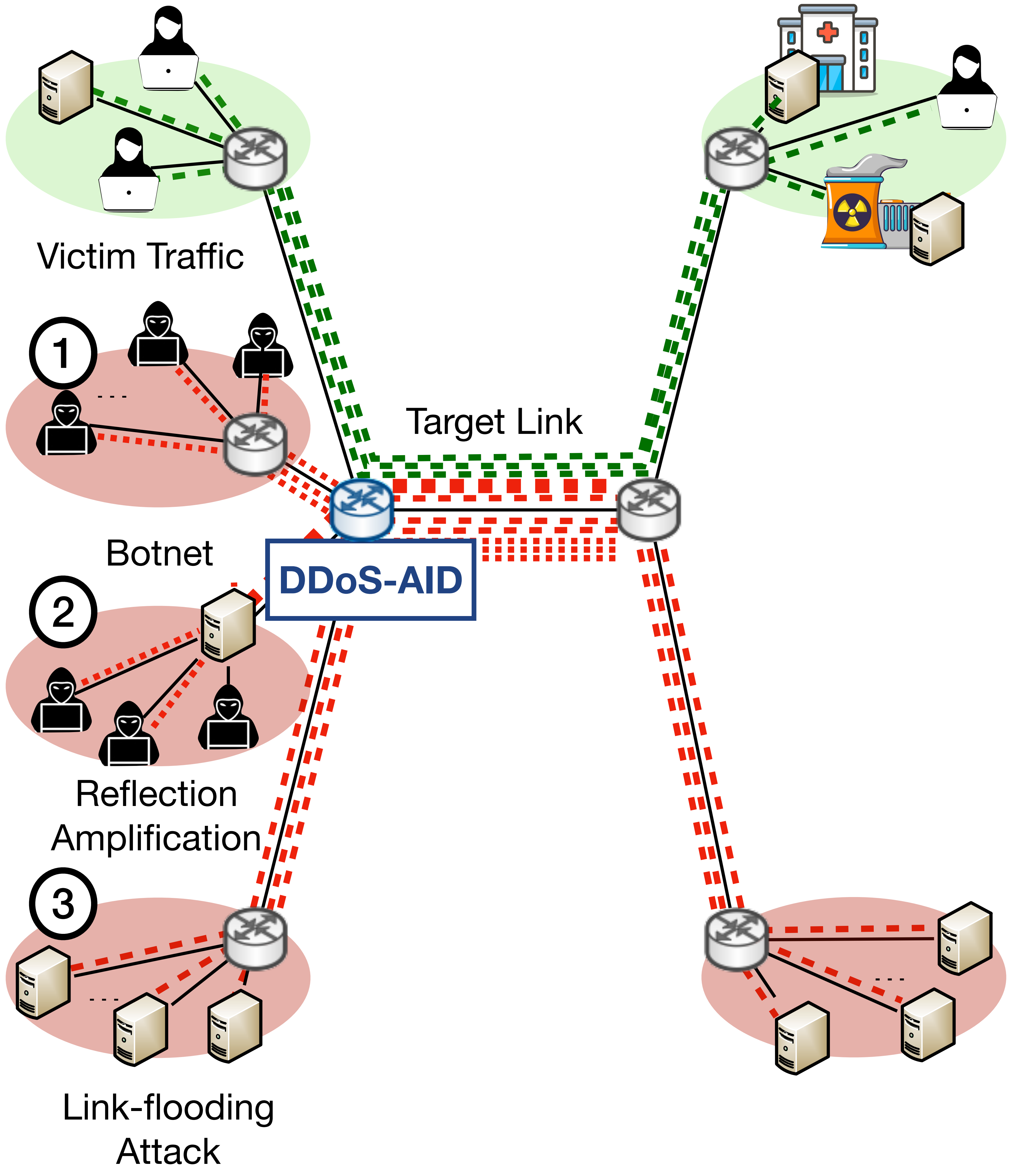
DDoS-AID focuses on volumetric attacks targeting a critical link in the network

Threat model



DDoS-AID focuses on volumetric attacks targeting a critical link in the network

Threat model



How can DDoS-AID mitigate (unknown) attacks automatically?

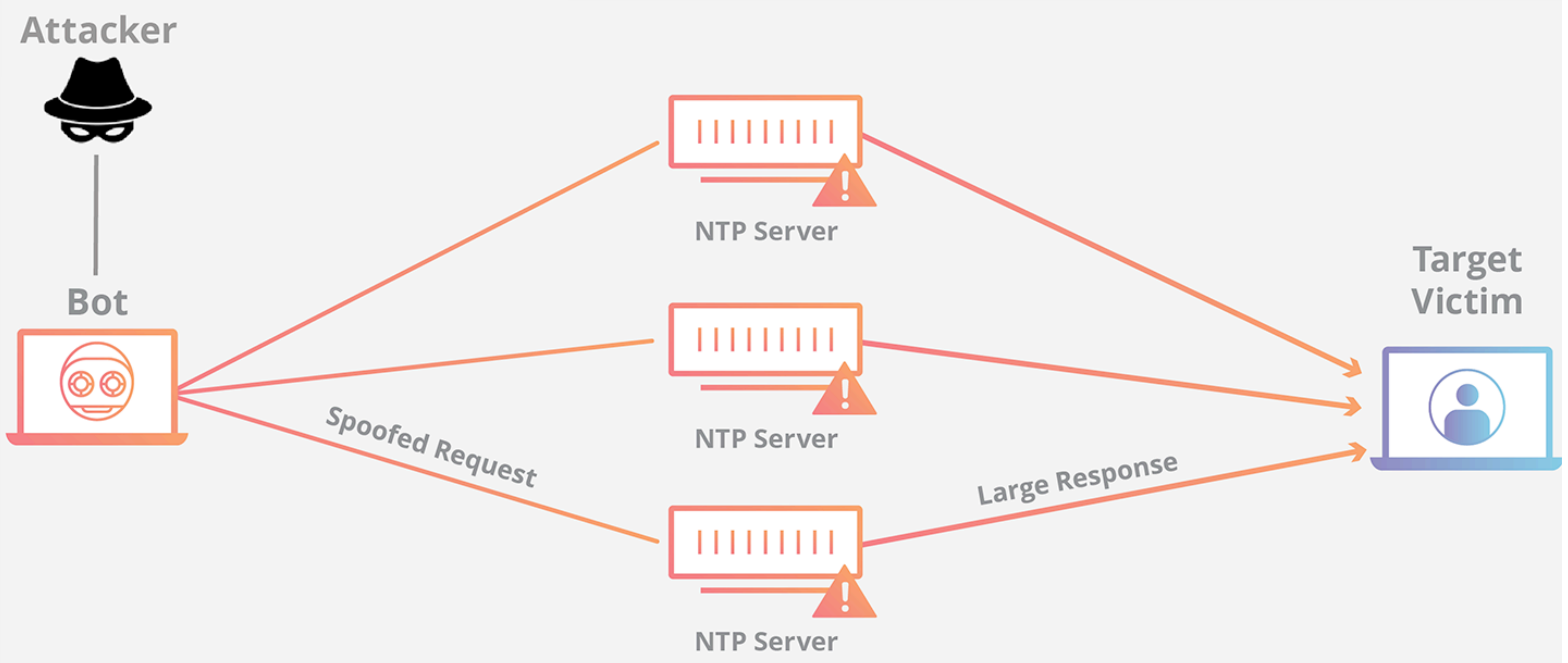
Observation

In practice, most DDoS attacks are composed of **unexpectedly-high rates** of **very-similar packets**

IT'S CLOBBERIN' TIME —

Biggest DDoS ever aimed at Cloudflare's content delivery network

Network Time Protocol attack reached 400Gbps.



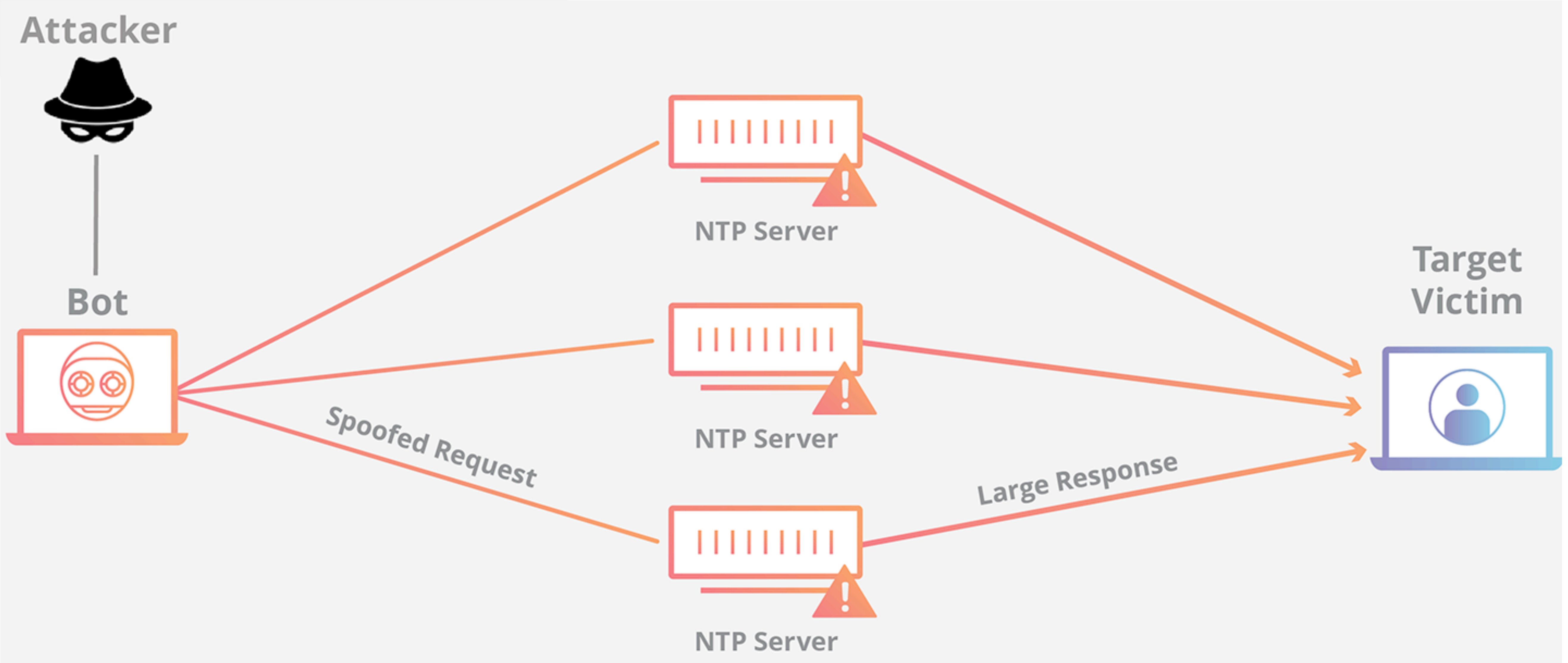
How can DDoS-AID mitigate (unknown) attacks automatically?

Observation In practice, most DDoS attacks are composed of **unexpectedly-high rates** of **very-similar packets**

IT'S CLOBBERIN' TIME —

Biggest DDoS ever aimed at Cloudflare's content delivery network

Network Time Protocol attack reached 400Gbps.



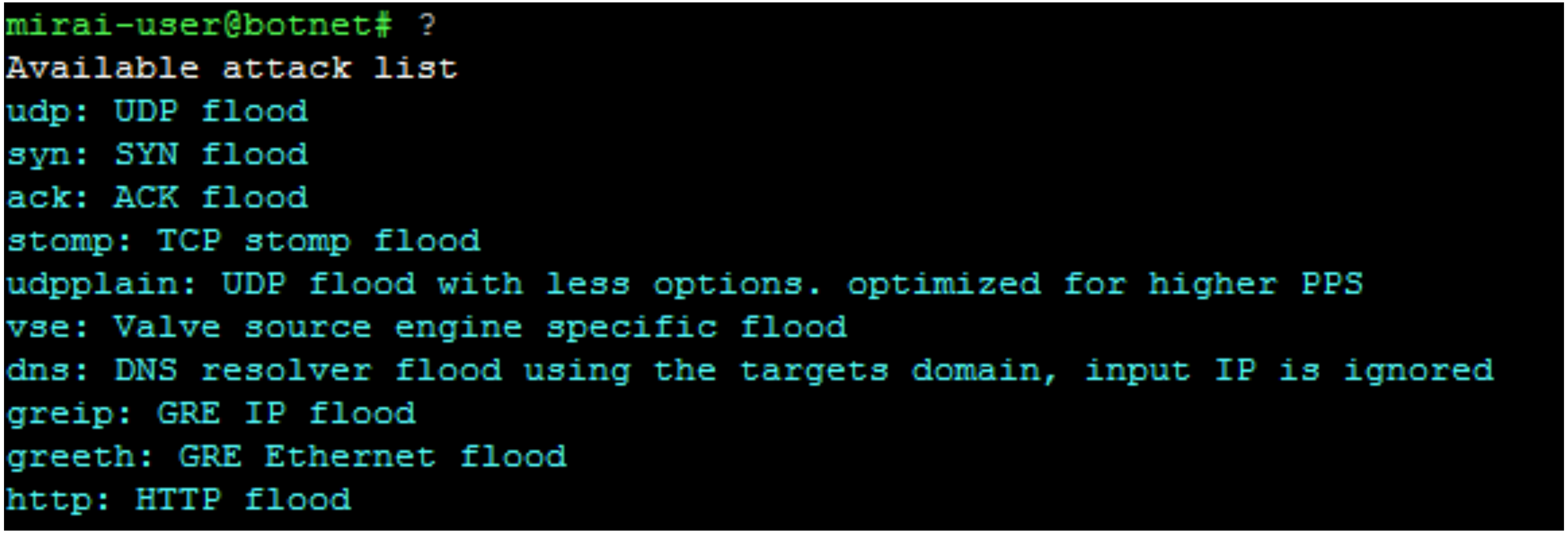
Cloudflare 2014: NTP

Google 2017: DNS

GitHub 2018: Memcached

How can DDoS-AID mitigate (unknown) attacks automatically?

Observation In practice, most DDoS attacks are composed of **unexpectedly-high rates** of **very-similar packets**



```
mirai-user@botnet# ?
Available attack list
udp: UDP flood
syn: SYN flood
ack: ACK flood
stomp: TCP stomp flood
udpplain: UDP flood with less options. optimized for higher PPS
vse: Valve source engine specific flood
dns: DNS resolver flood using the targets domain, input IP is ignored
greip: GRE IP flood
greeth: GRE Ethernet flood
http: HTTP flood
```

How can DDoS-AID mitigate (unknown) attacks automatically?

Observation In practice, most DDoS attacks are composed of **unexpectedly-high rates** of **very-similar packets**

Challenge **We don’t know** in advance **where** this similarity will be

How can DDoS-AID mitigate (unknown) attacks automatically?

Observation

In practice, most DDoS attacks are composed of **unexpectedly-high rates** of **very-similar packets**

Challenge

We don't know in advance **where** this similarity will be

Opportunity

Online clustering allows us to **automatically infer this pattern**

How can DDoS-AID mitigate (unknown) attacks safely?

Challenge

Being fully-automated requires **making decisions under uncertainty**

This implies the **risk of false positives**

Filtering (dropping) is **too drastic**

Throttling is very hard: How to set the right rate?

How can DDoS-AID mitigate (unknown) attacks safely?

Challenge

Being fully-automated requires **making decisions under uncertainty**

This implies the **risk of false positives**

Filtering (dropping) is **too drastic**

Throttling is very hard: How to set the right rate?

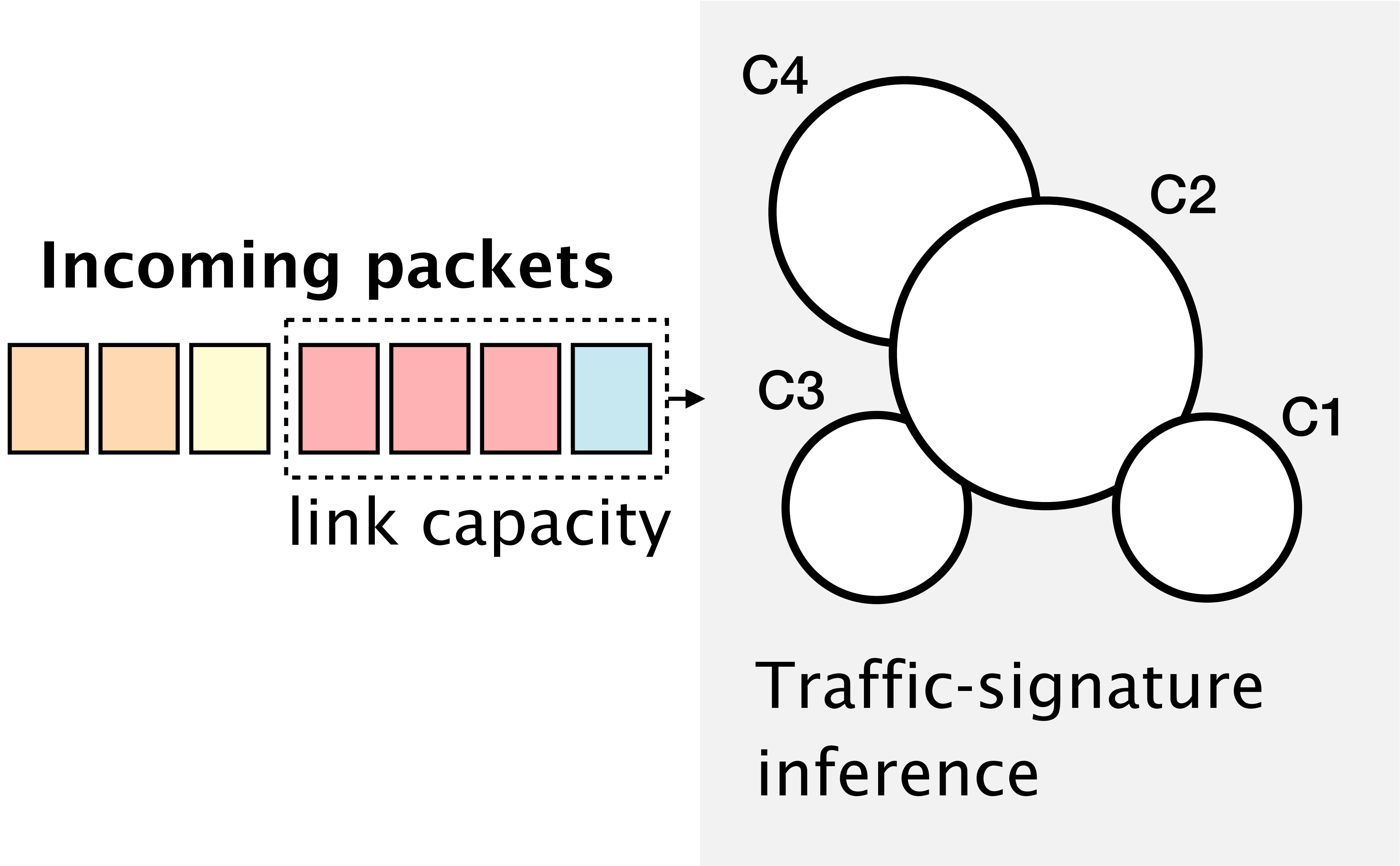
Opportunity

Programmable scheduling allows us to **automatically throttle traffic** at the **right rate**

DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

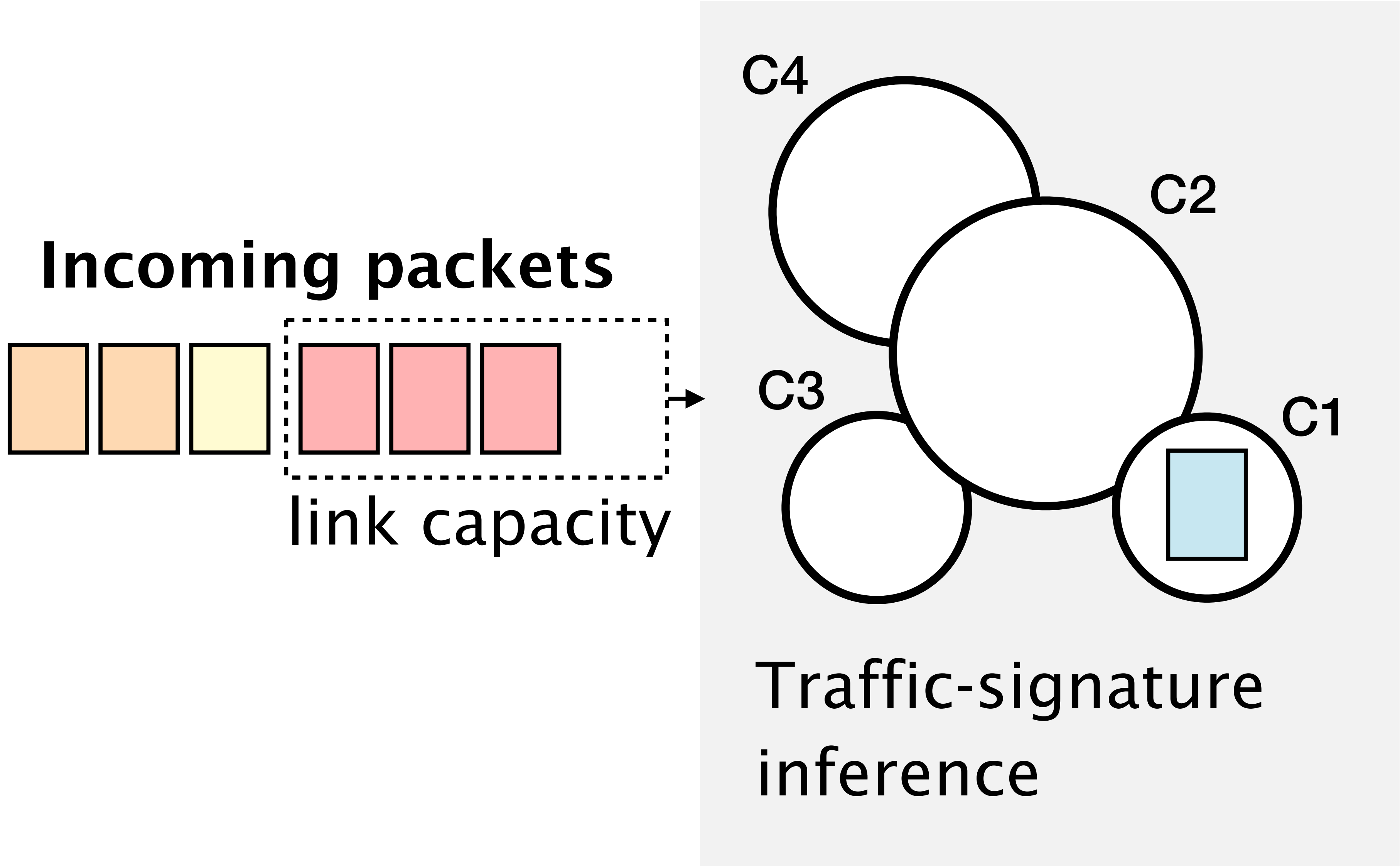
programmable scheduling



DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques directly **in the network**

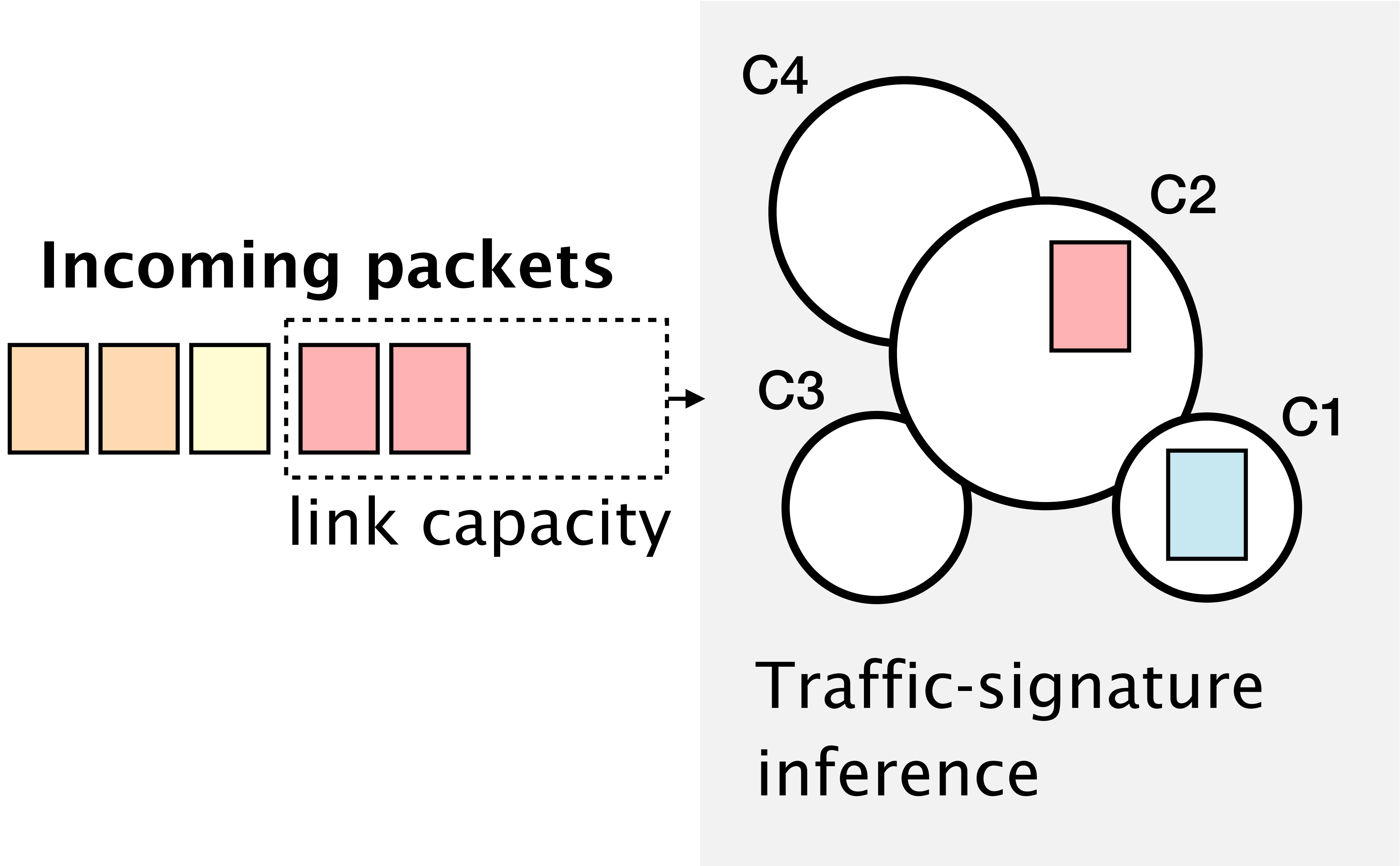
programmable scheduling



DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques directly **in the network**

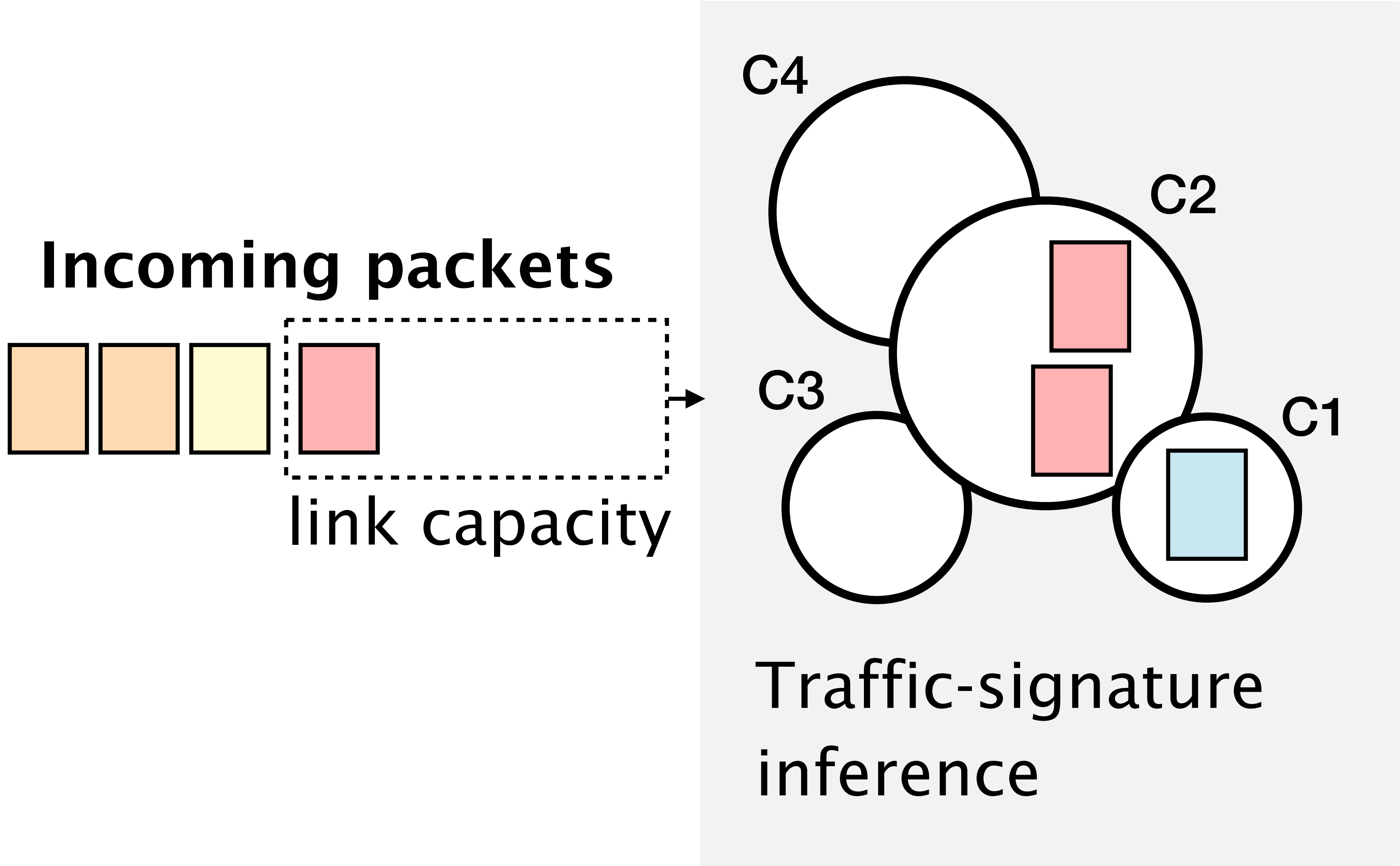
programmable scheduling



DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

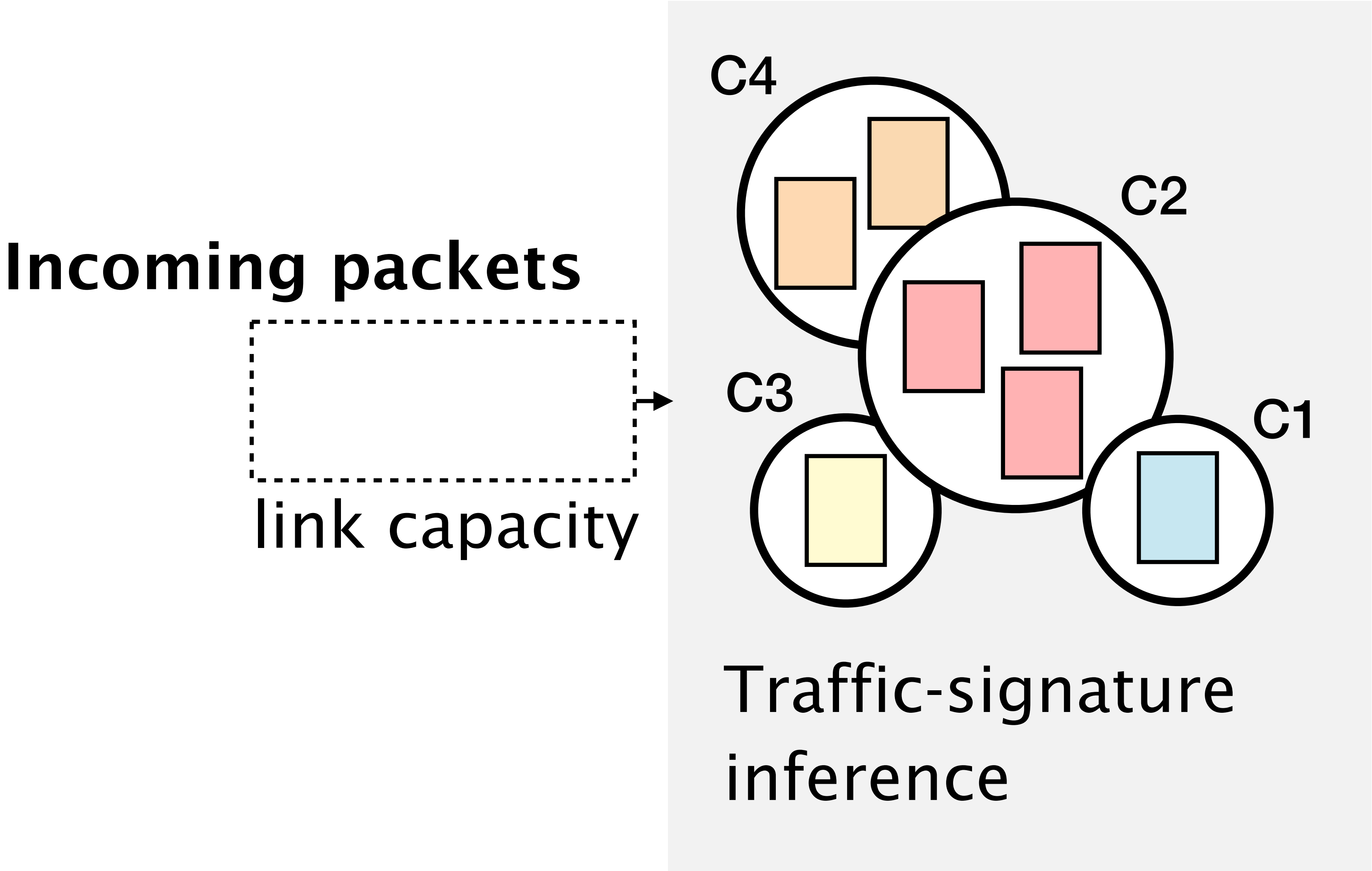
programmable scheduling



DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques directly **in the network**

programmable scheduling



Incoming packets

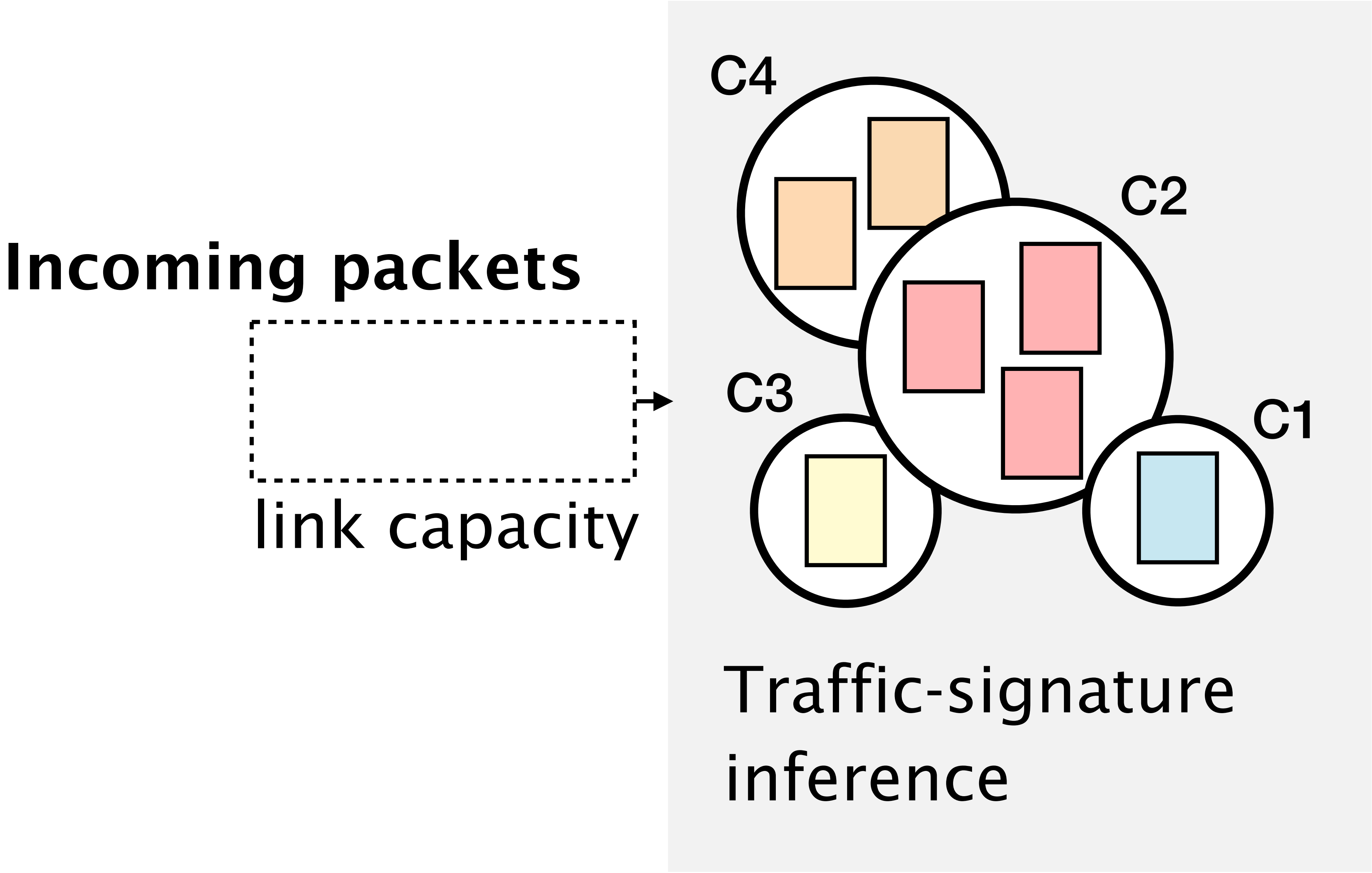
link capacity

Traffic-signature inference

DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques directly **in the network**

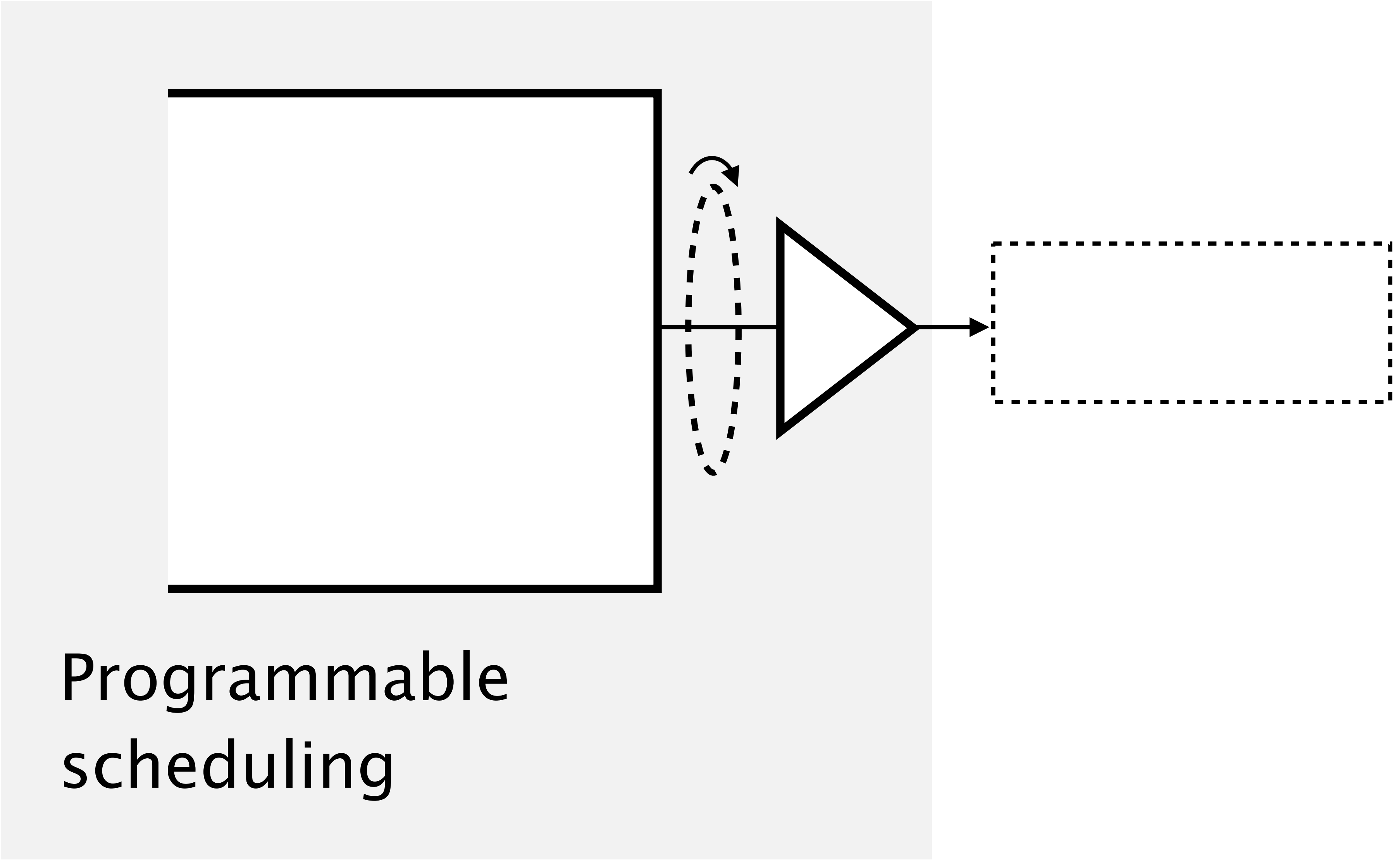
programmable scheduling



Incoming packets

link capacity

Traffic-signature inference

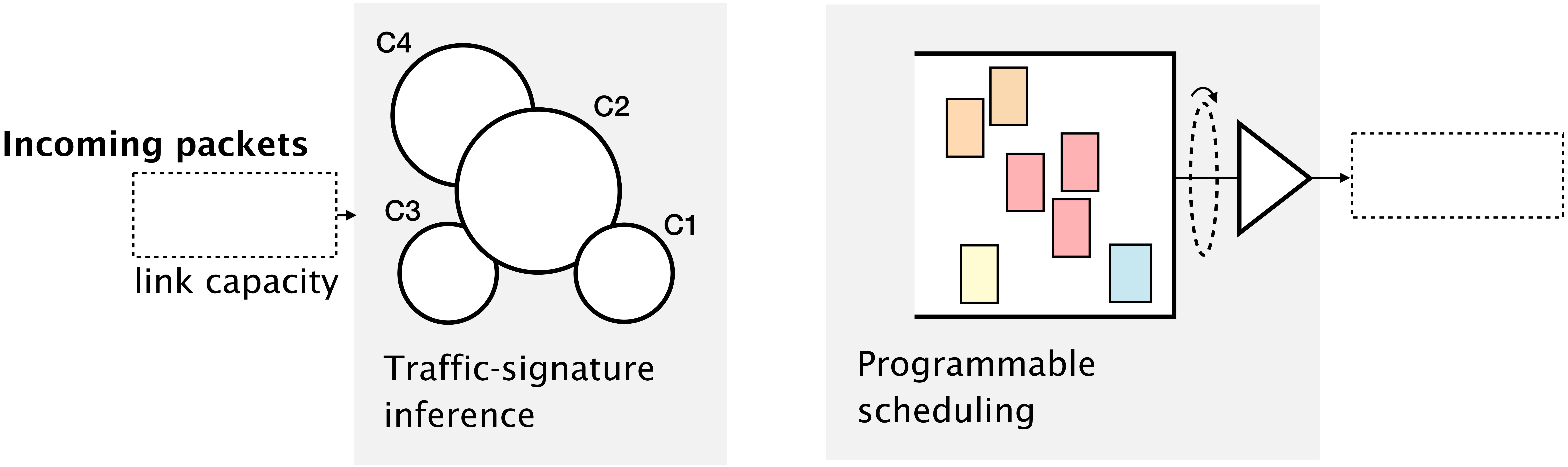


Programmable scheduling

DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

programmable scheduling



Incoming packets

link capacity

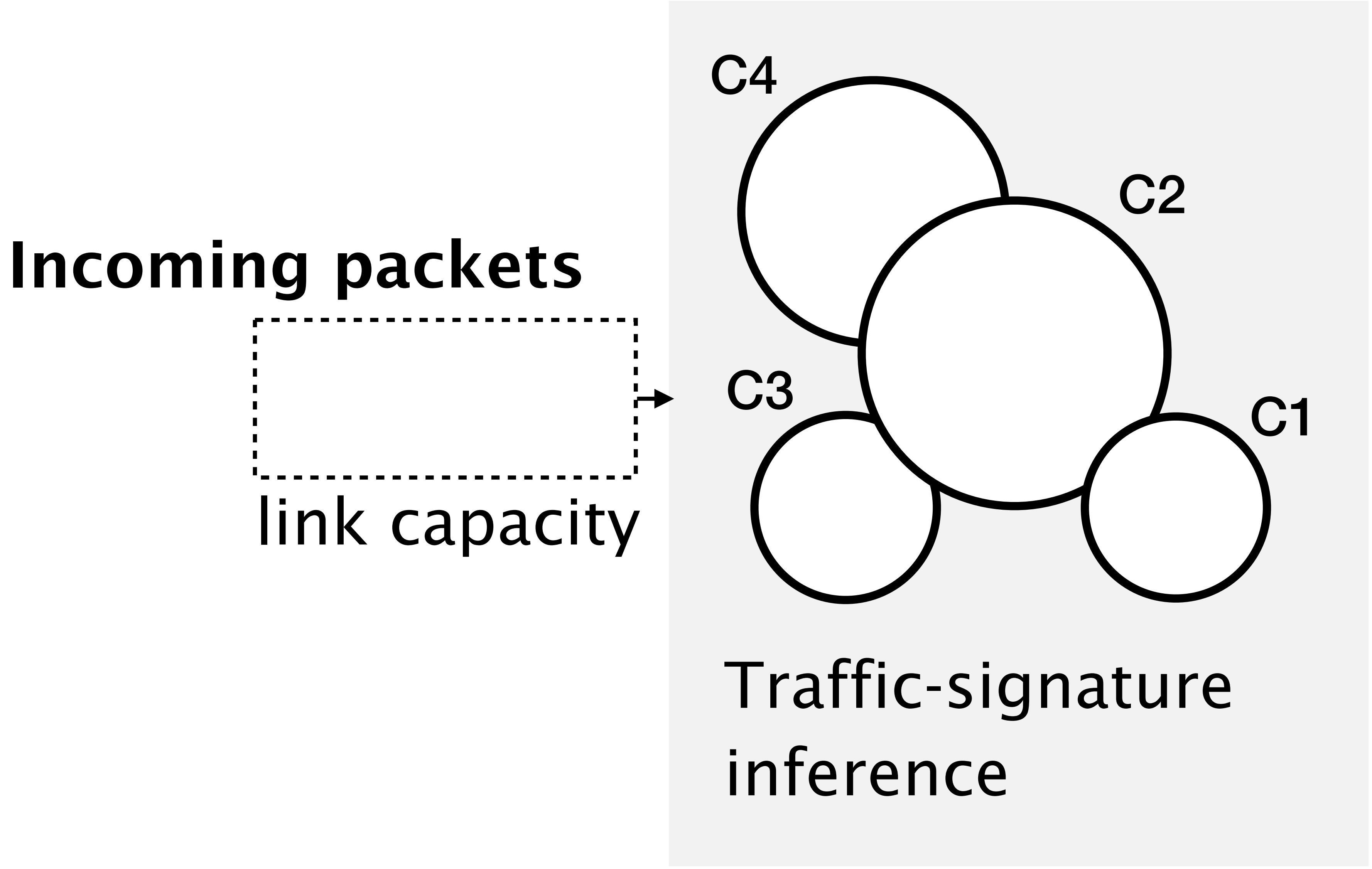
Traffic-signature inference

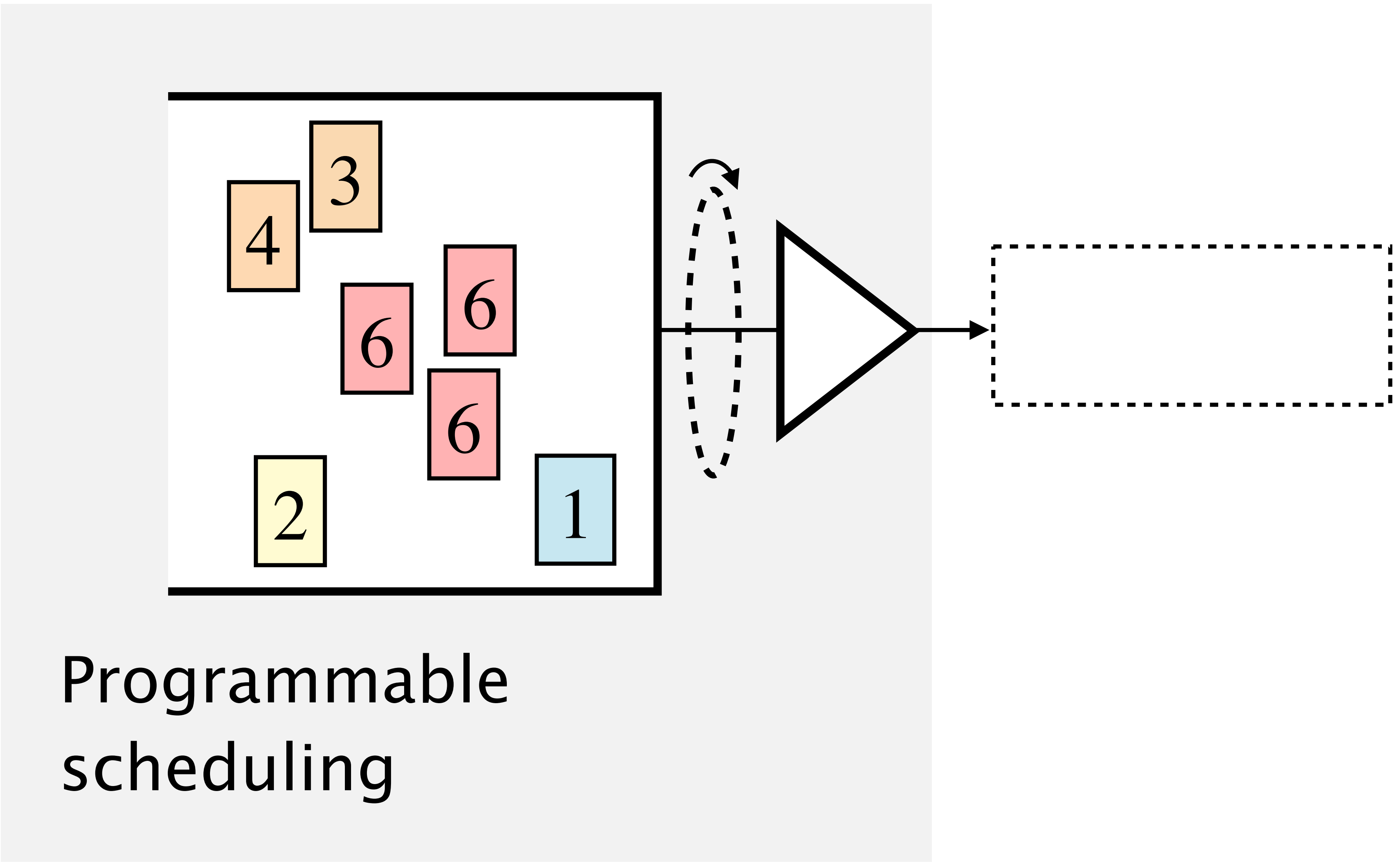
Programmable scheduling

DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

programmable scheduling

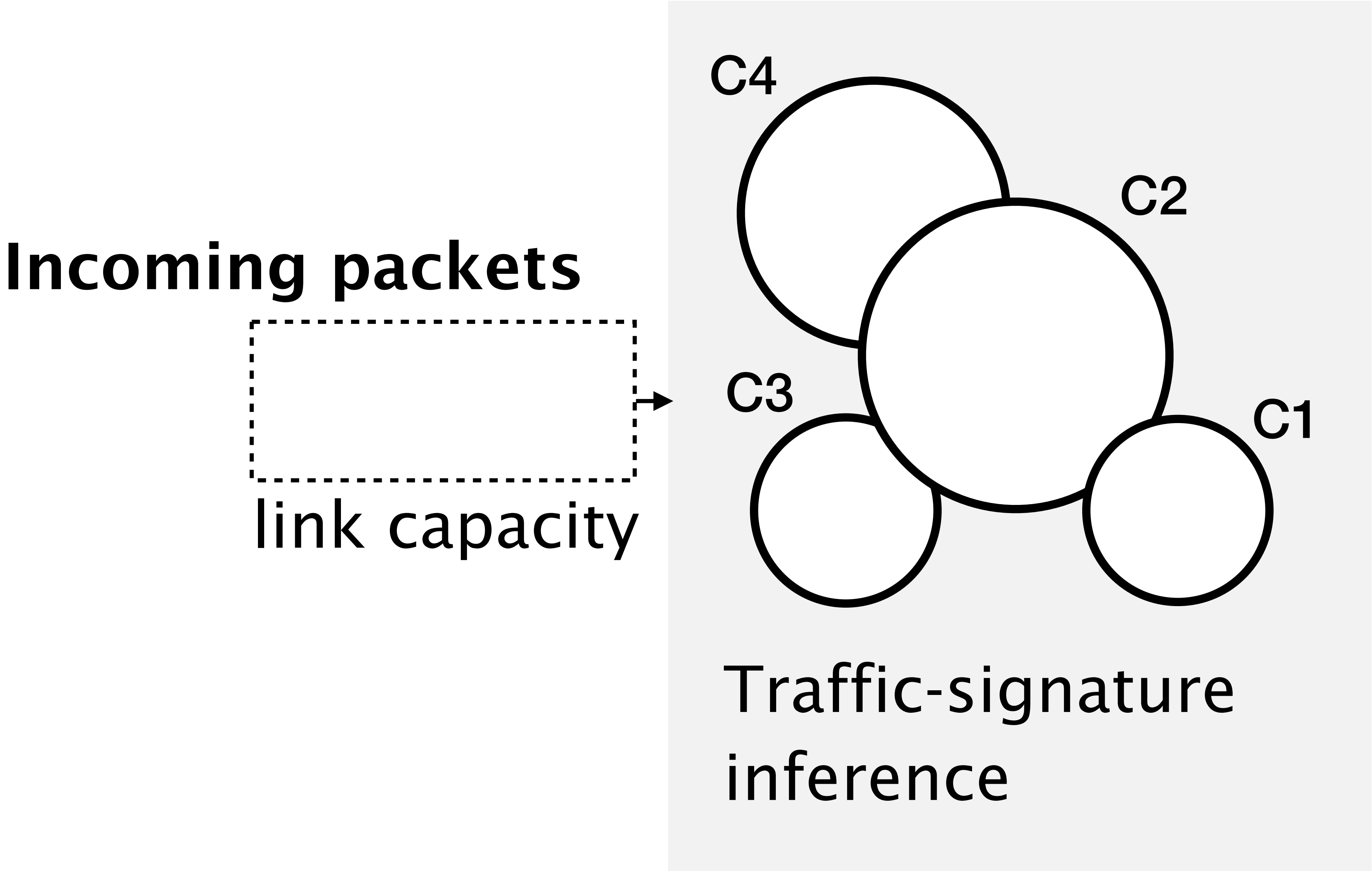


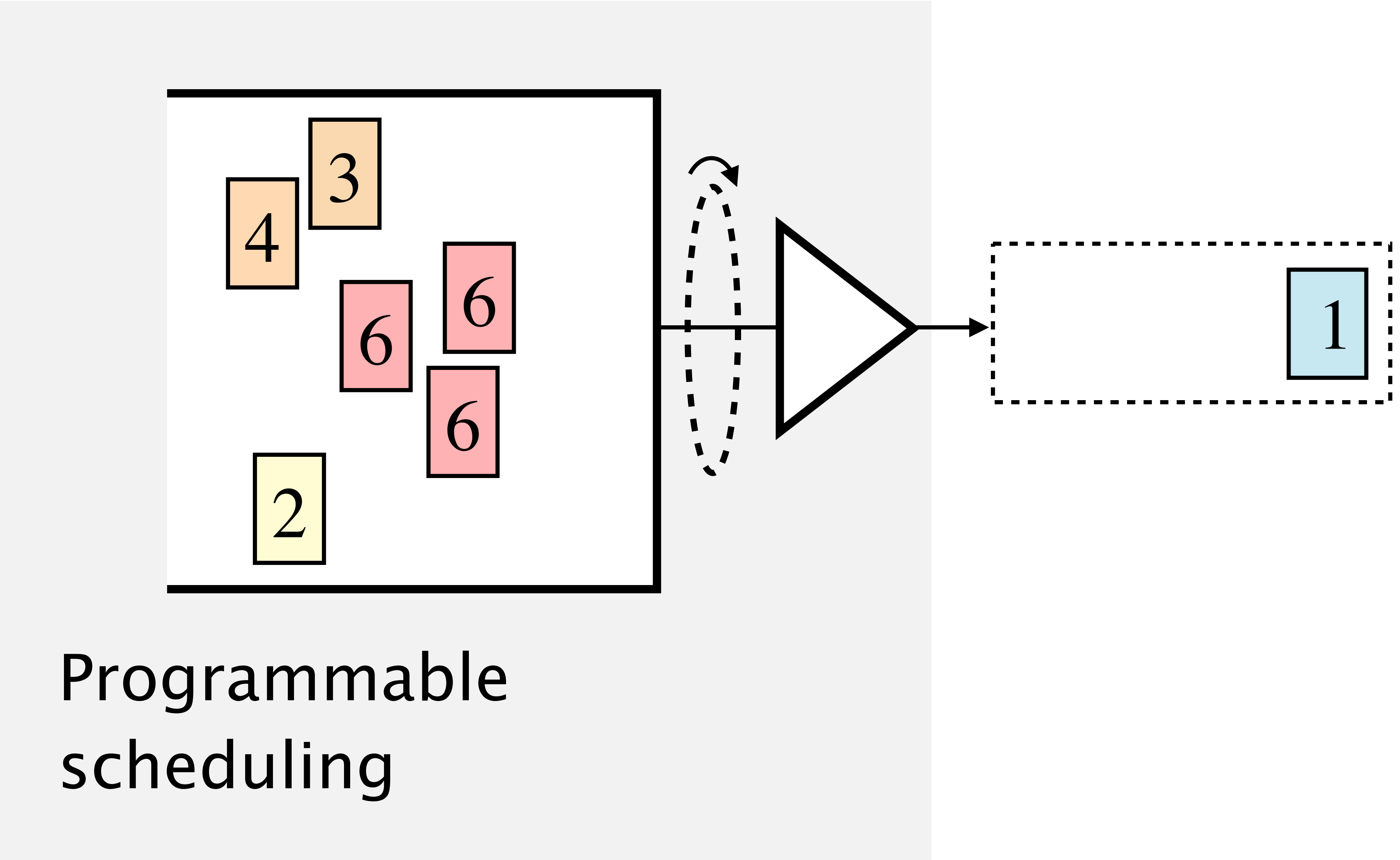


DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

programmable scheduling

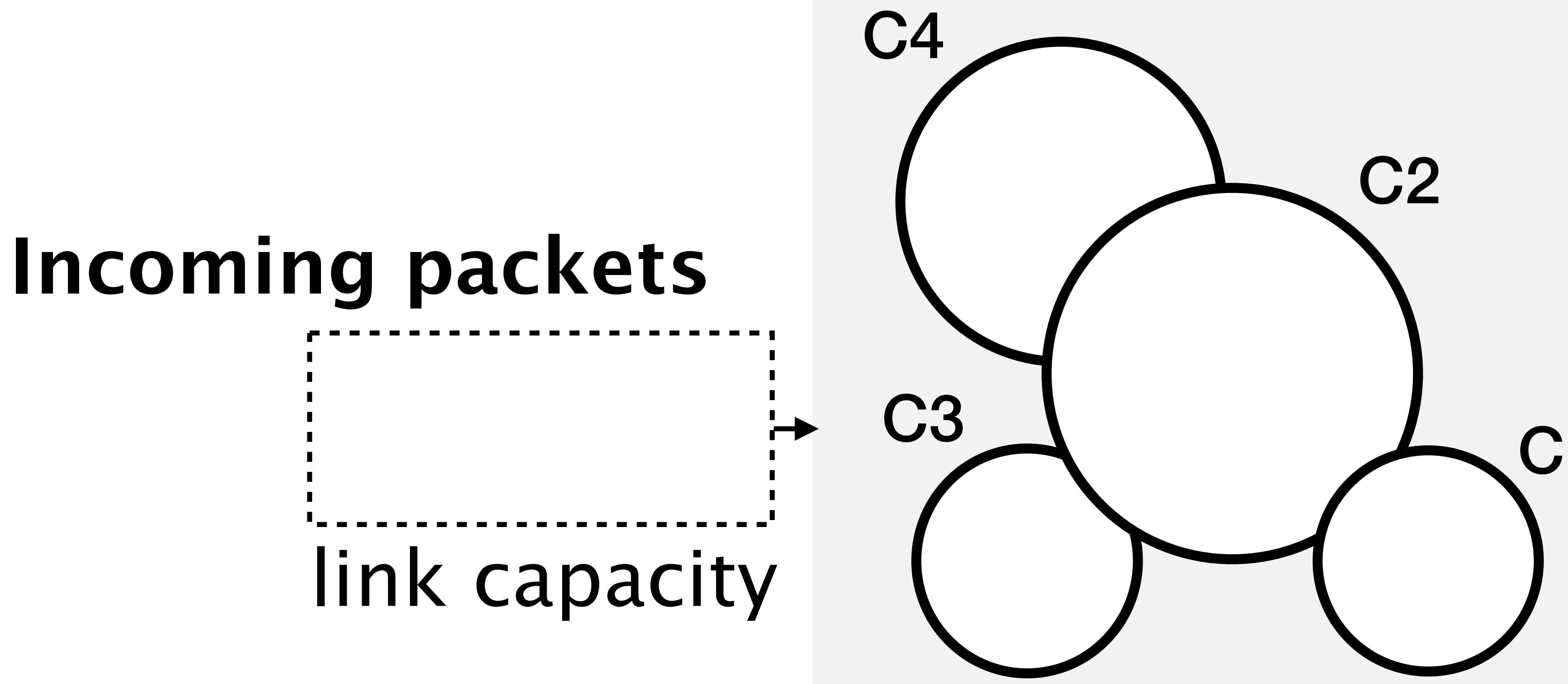




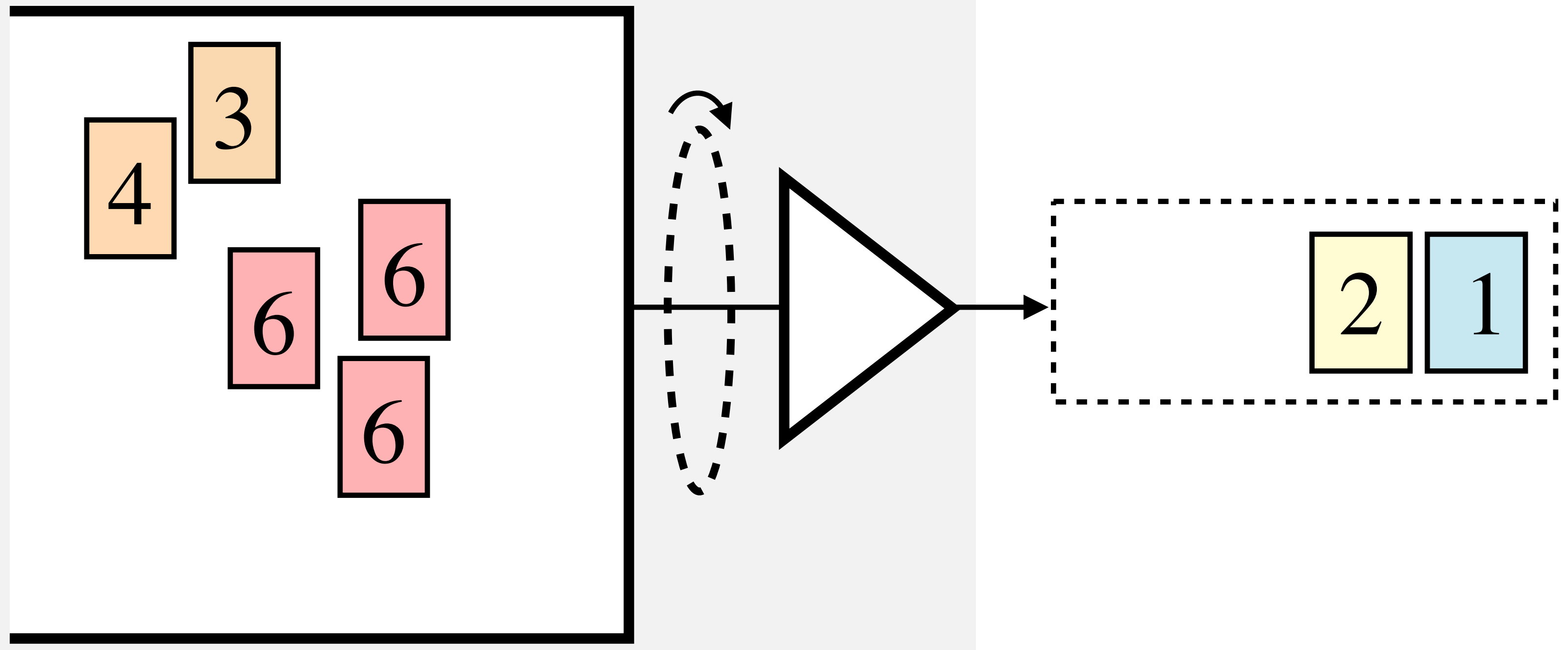
DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

programmable scheduling



Traffic-signature inference

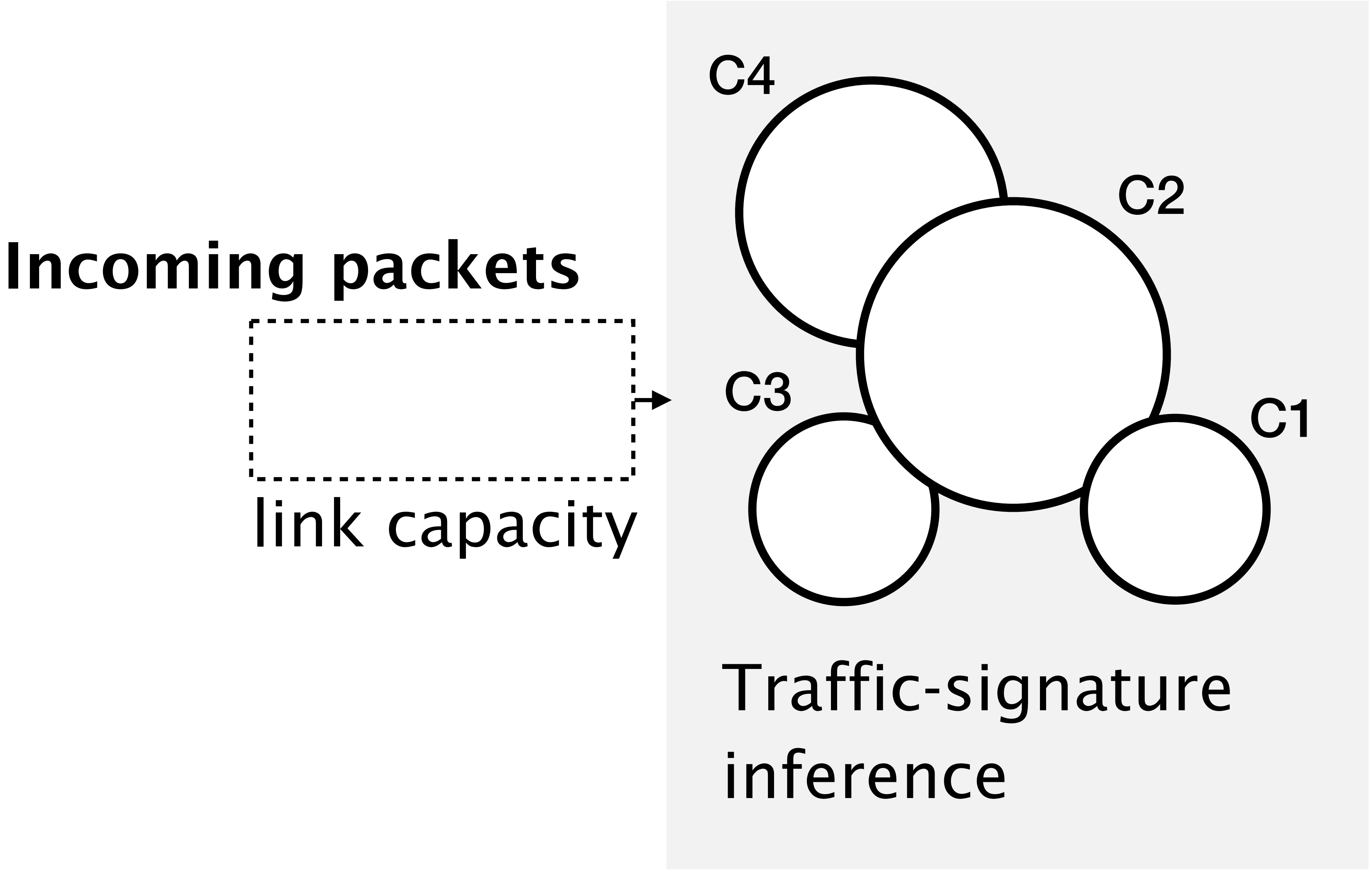


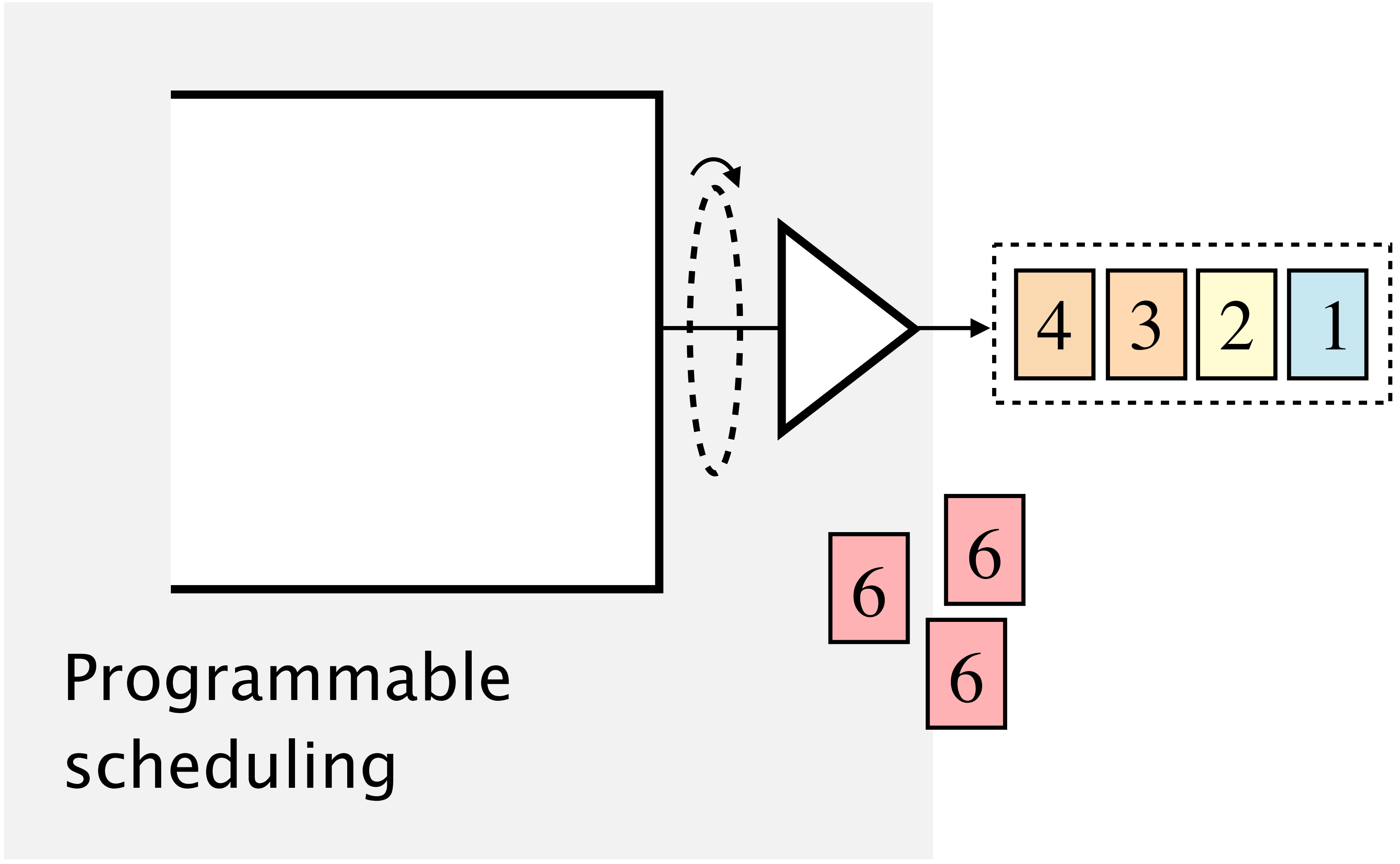
Programmable scheduling

DDoS-AID combines in-network online-clustering with programmable scheduling

online-clustering techniques
directly **in the network**

programmable scheduling





DDoS-AID combines in-network online-clustering with programmable scheduling

Identify unexpectedly-high rates of very-similar packets

online-clustering techniques
directly **in the network**

- ✔ Fully-automated detection
- ✔ Covers new attacks

- ✔ Absorb high rates
- ✔ Analyze all traffic with no latency increase

Automatically throttle identified traffic

programmable scheduling

- ✔ Non-binary assessment
- ✔ Only drop under congestion
- ✔ Starts dropping the most malicious
- ✔ Minimizes the impact of false positives

DDoS-AID: Automated In-Network DDoS Mitigation as a First Line of Defense

1 Key insights

How does it work

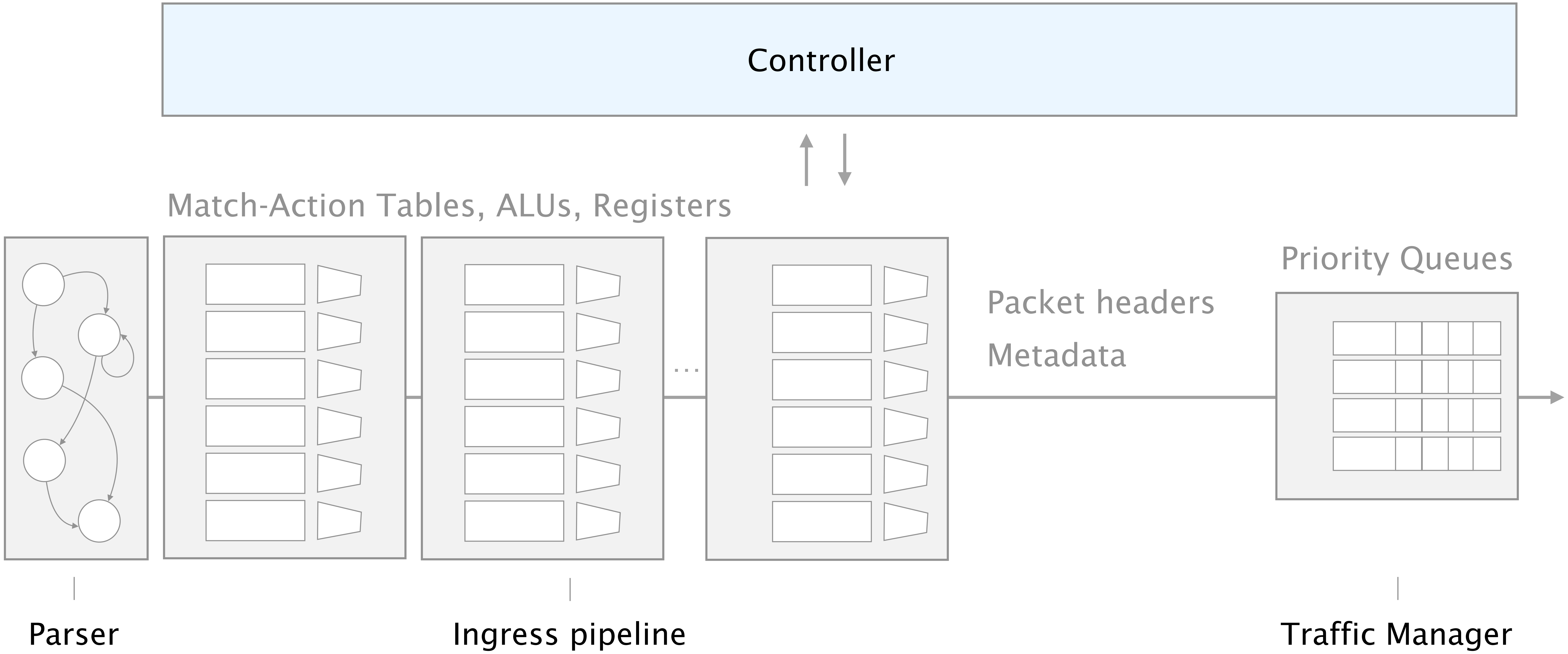
2 Implementation

How can it be deployed

3 Evaluation

How well does it perform

DDoS-AID runs at line-rate on existing programmable hardware



DDoS-AID runs at line-rate on existing programmable hardware

Challenges

“Off-the-shelf” online clustering provides coarse results and no guarantees

Two-step mitigation

Extract info about the clusters, analyze their quality and *only then* mitigate

DDoS-AID runs at line-rate on existing programmable hardware

Challenges

“Off-the-shelf” online clustering provides coarse results and no guarantees

Two-step mitigation

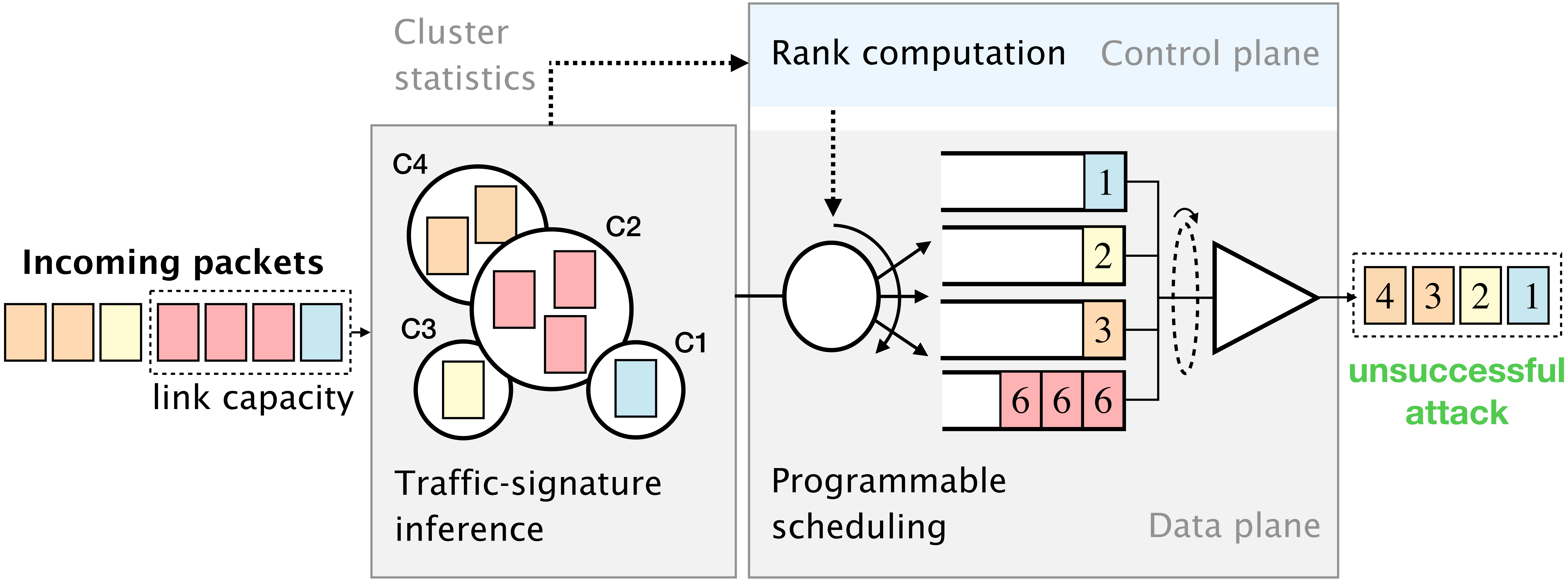
Extract info about the clusters, analyze their quality and *only then* mitigate

Fit both, clustering algorithm and programmable scheduler, in hardware

Hybrid design

Rank computation and queue mapping offloaded to control plane

DDoS-AID runs at line-rate on existing programmable hardware



DDoS-AID: Automated In-Network DDoS Mitigation as a First Line of Defense

1 Key insights

How does it work

2 Implementation

How can it be deployed

3 Evaluation

How well does it perform

Evaluation

Disclaimer

Performance depends on the characteristics of benign and attack traffic

Evaluation in paper

Performance evaluation on CICDDoS2019 dataset

Behavior analysis on a morphing attack

Measurement impact of the design decisions

Reaction-time evaluation on hardware testbed

Comparison with state of the art solutions*

Evaluation

Disclaimer

Performance depends on the characteristics of benign and attack traffic

Evaluation in paper

Performance evaluation on CICDDoS2019 dataset

Behavior analysis on a morphing attack

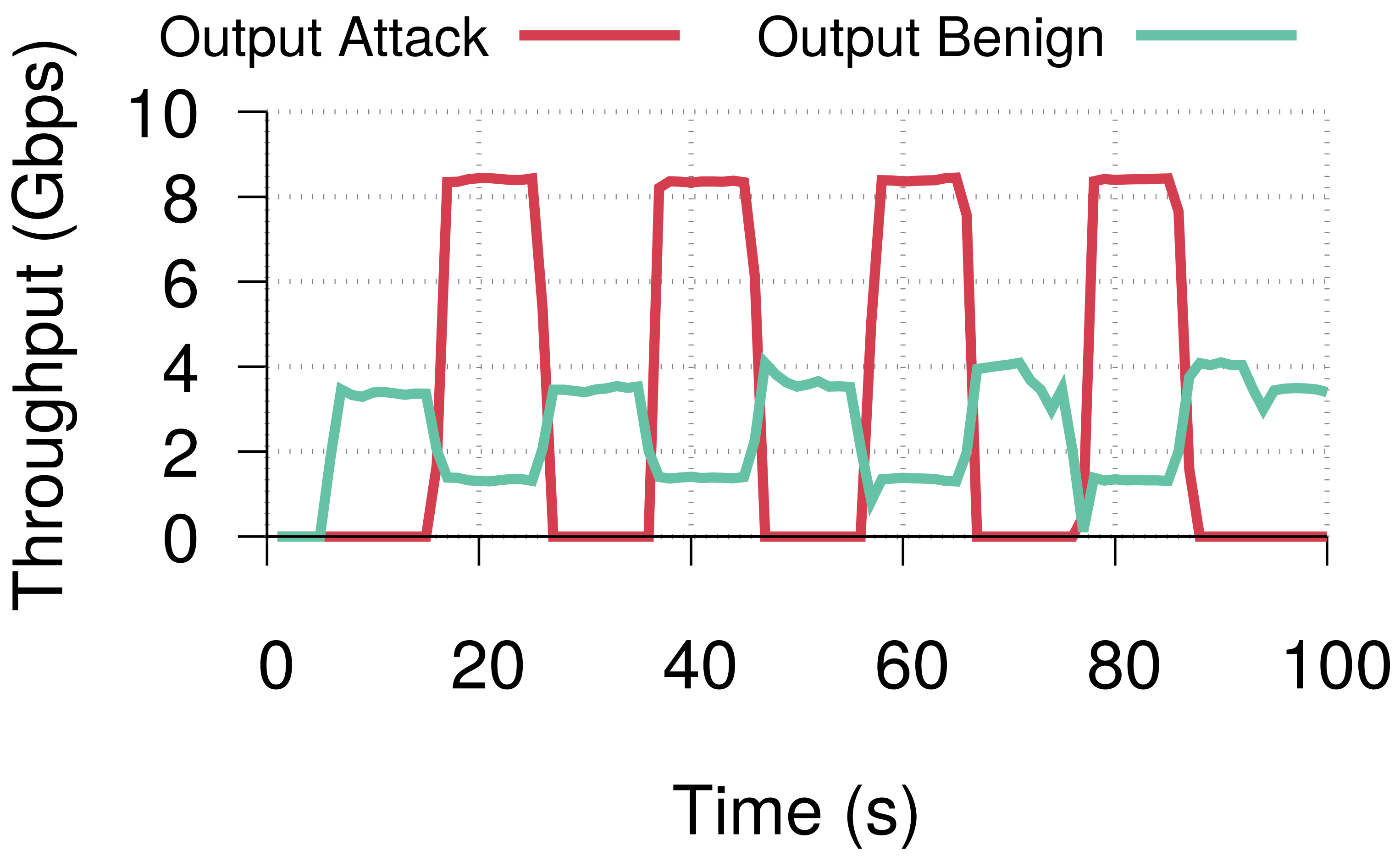
Measurement impact of the design decisions

Reaction-time evaluation on hardware testbed

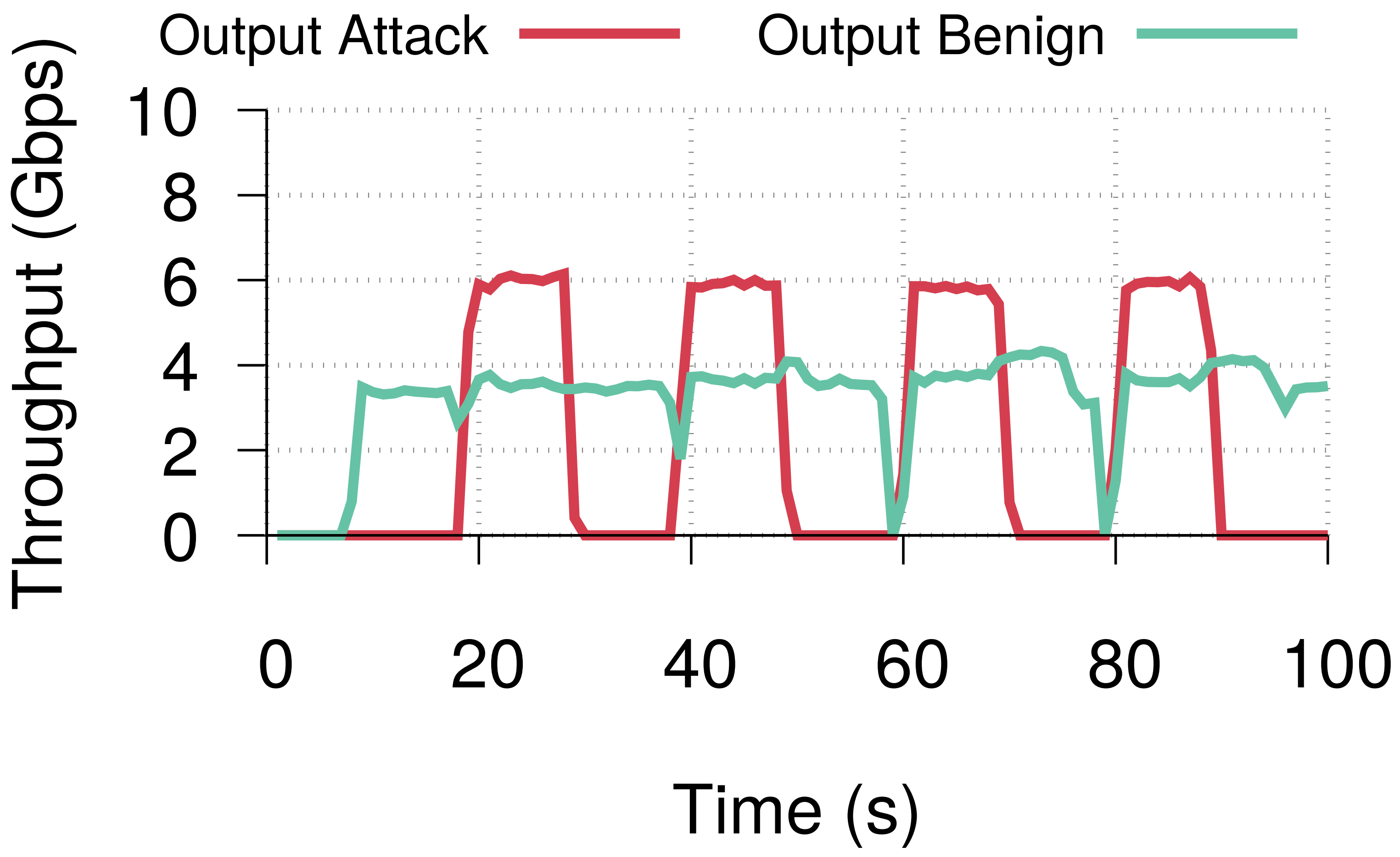
Comparison with state of the art solutions*

DDoS-AID achieves sub-second reaction times

FIFO



DDoS-AID (4 clusters, 4 features)



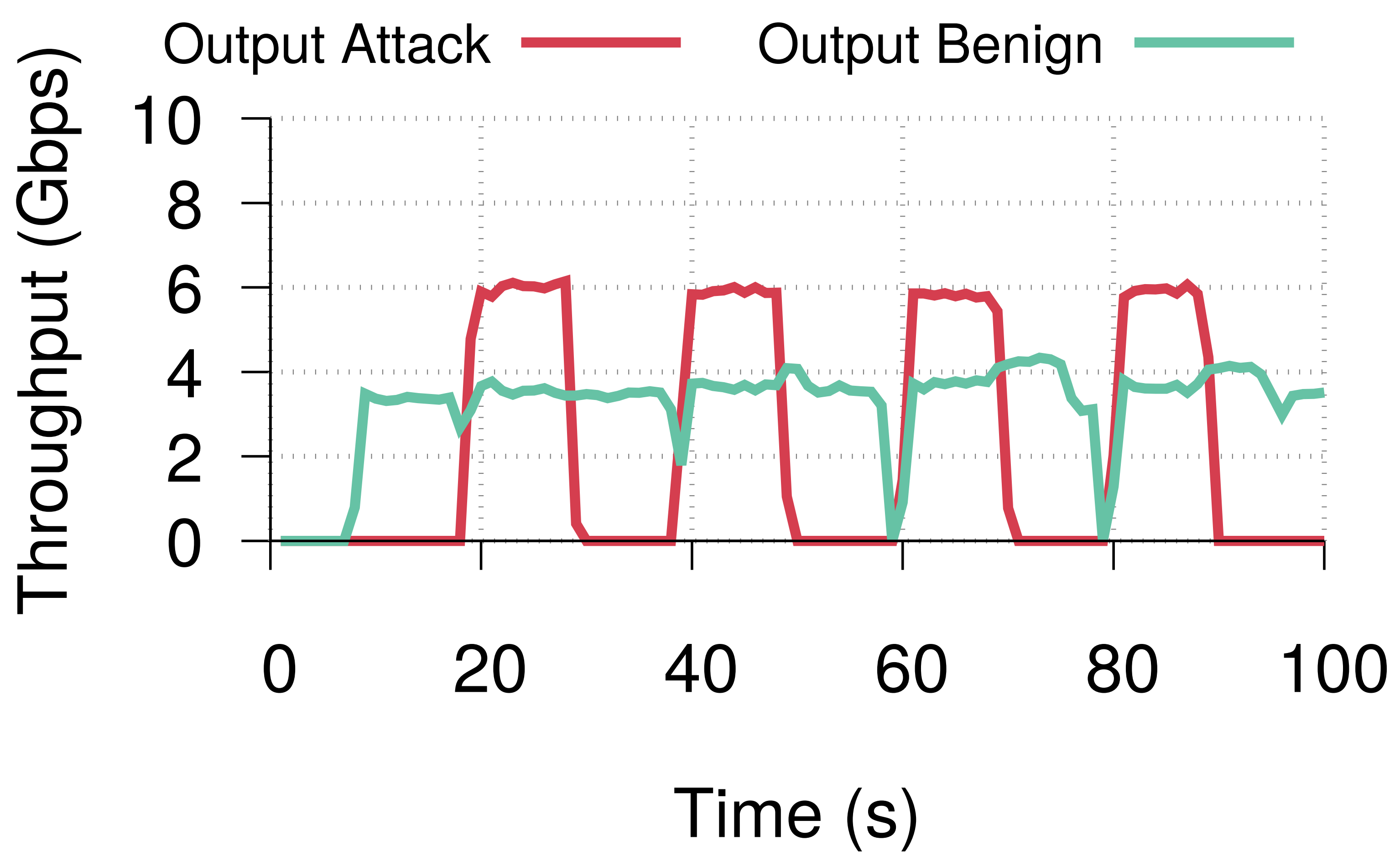
DDoS-AID achieves sub-second reaction times

Reaction time

1. Poll throughput statistics
2. Update cluster ranks (priorities)
3. Deploy them to data plane

(~1s with unoptimized controller)

DDoS-AID (4 clusters, 4 features)



DDoS-AID achieves sub-second reaction times

DDoS-AID

1. Poll throughput statistics
2. Update cluster ranks (priorities)
3. Deploy them to data plane

(~1s with unoptimized controller)

Jaqen (State of the art)

1. Detect attack
2. Compute mitigation module
3. Orchestrate rerouting legitimate traffic
4. Replicate switch state to controller
5. Reprogram switch with mitigation module

DDoS-AID achieves sub-second reaction times

DDoS-AID

1. Poll throughput statistics
2. Update cluster ranks (priorities)
3. Deploy them to data plane

(~1s with unoptimized controller)

Jaqen (State of the art)

1. Detect attack
2. Compute mitigation module
3. Orchestrate rerouting legitimate traffic
4. Replicate switch state to controller
5. Reprogram switch with mitigation module

(This step alone is already
2x slower than *all* DDoS-AID)

DDoS-AID: A *fully automated, and-yet-safe* in-network DDoS defense

Most DDoS attacks are composed of unexpectedly-high rates of very-similar packets

DDoS-AID captures this characteristic by relying on in-network online clustering

DDoS-AID mitigates attacks safely by relying on programmable packet scheduling

DDoS-AID: Automated In-Network DDoS Mitigation as a First Line of Defense

Albert Gran Alcoz(1), Martin Strohmeier(2), Vincent Lenders(2), Laurent Vanbever(1)

Cyber-Alp Retreat

July, 01 2020

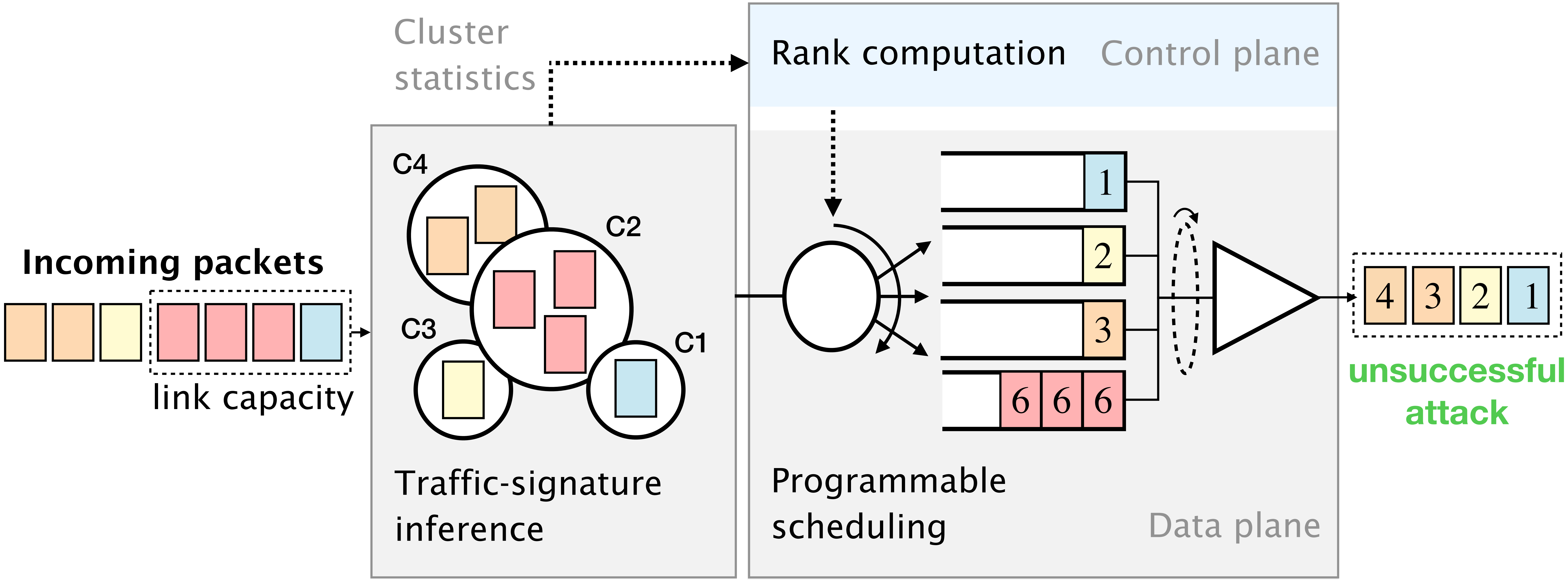
(1) ETH zürich

(2) Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

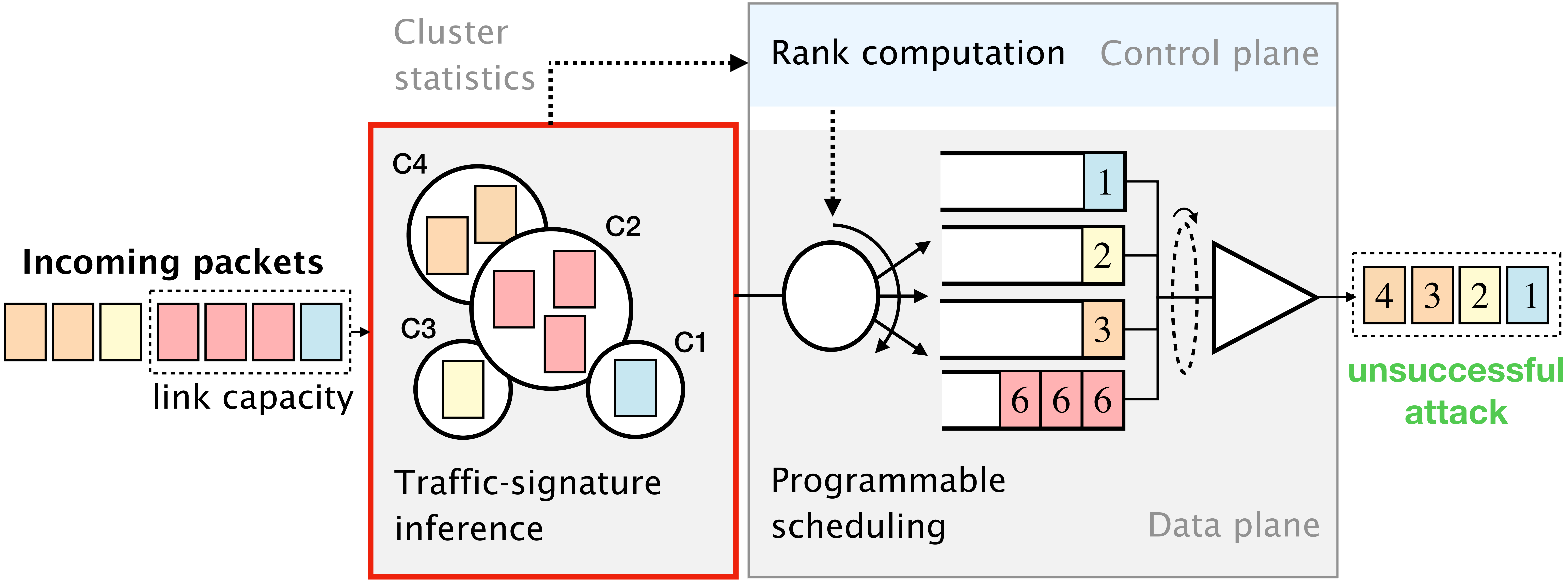
armasuisse

Additional slides for Q&A

DDoS-AID runs at line-rate on existing programmable hardware

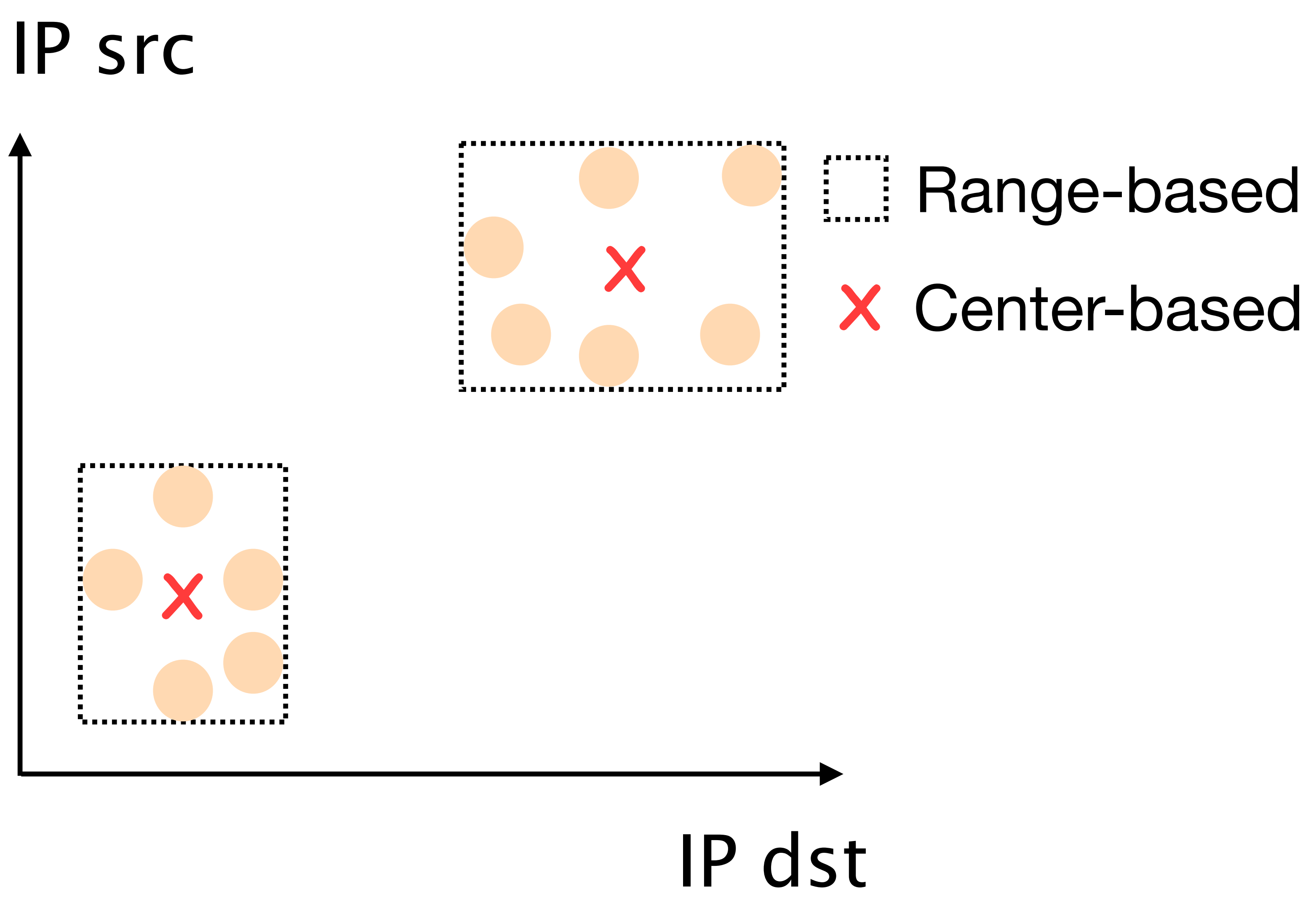


DDoS-AID runs at line-rate on existing programmable hardware



DDoS-AID clusters packets based on their header space representations

Each packet is a point in the header space



Packet headers are the clustering features

Two representations:

Distance-based (e.g., IP src, IP dst, TTL)

ranges per field [min_f, max_f]

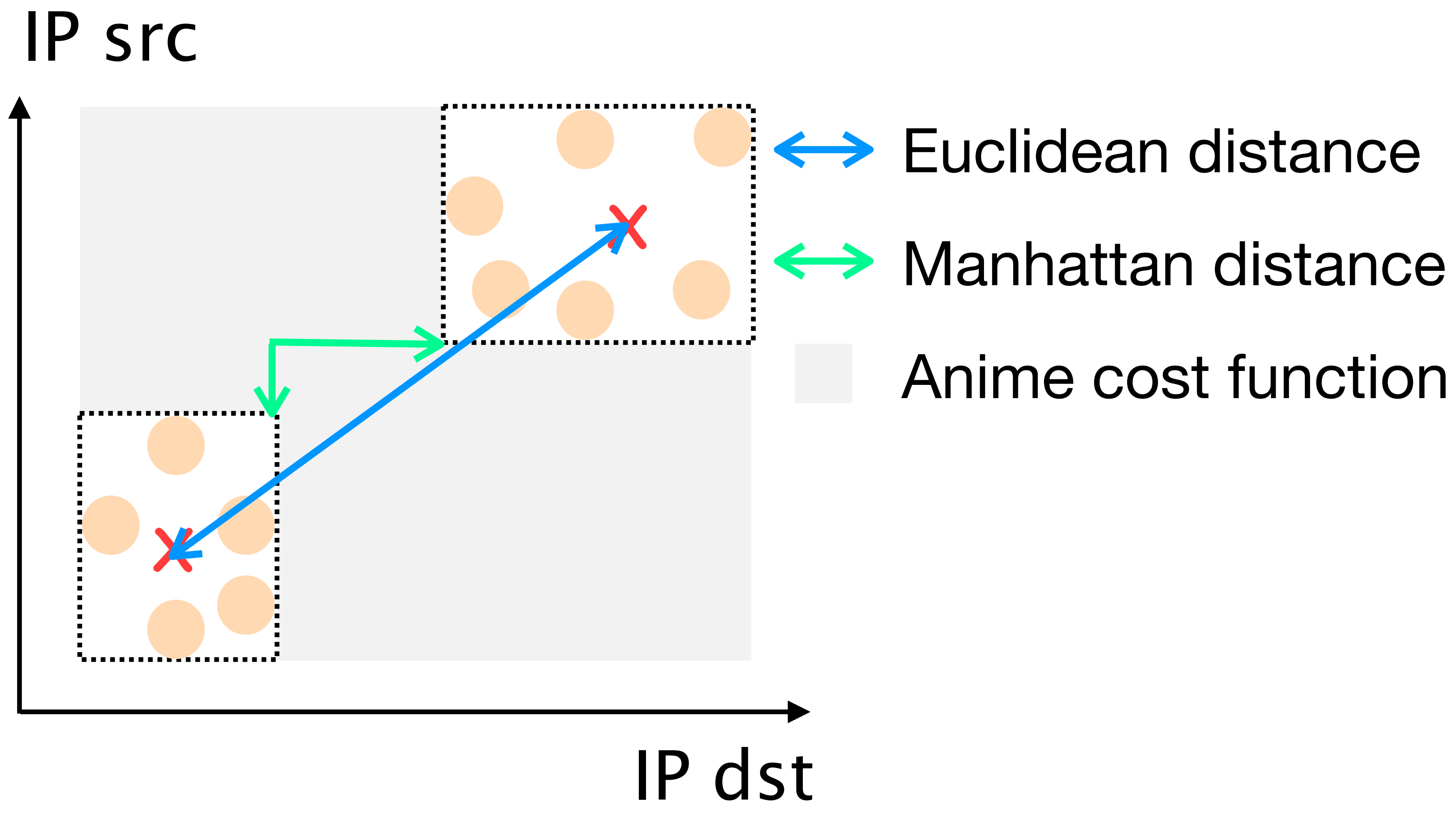
Distance-independent (e.g., sport, proto, flags)

count_distinct [f]

Objective Find *k* clusters that minimize the represented area while covering all packets observed

DDoS-AID clusters packets based on their header space representations

Algorithm



For each new packet:

Compute (adapted-)Manhattan distance from packet to all clusters

Select cluster with smallest distance

Objective Find *k* clusters that minimize the represented area while covering all packets observed

DDoS-AID clusters packets based on their header space representations

Advantages

Online-clustering has same requirements as programmable switches

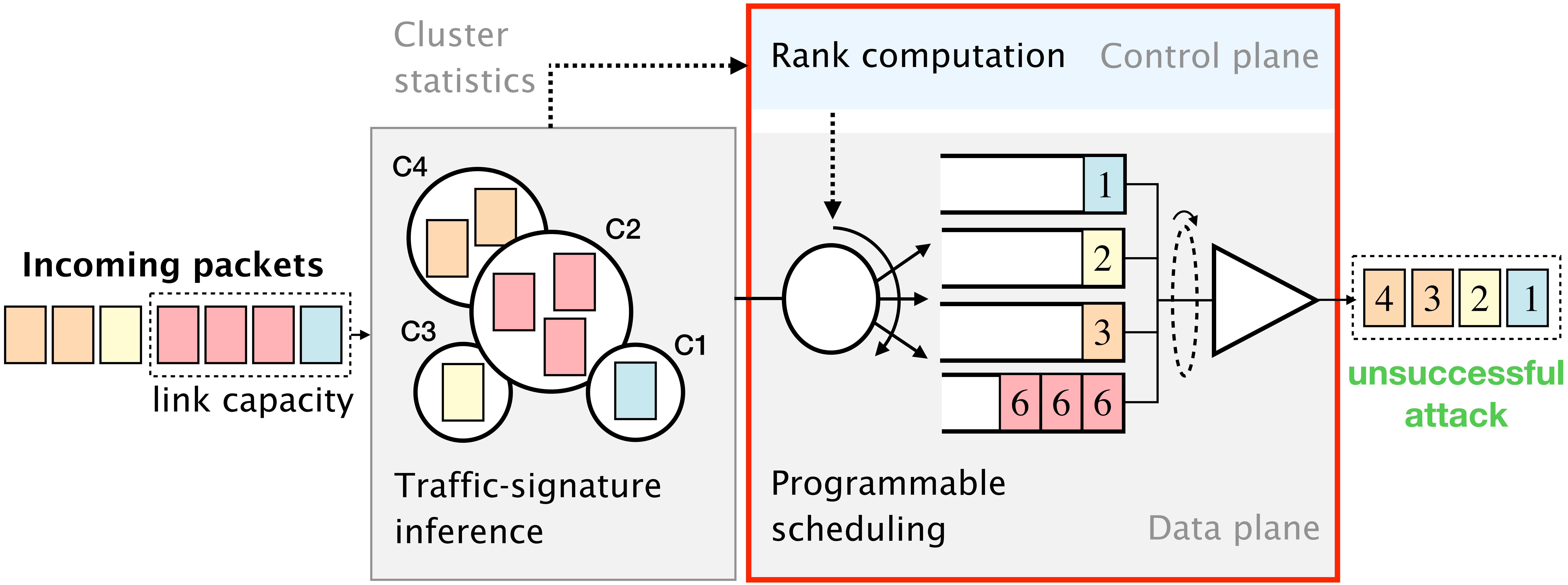
Ranges can be easily updated (max, min operations)

Range-representation allows us to extract information about cluster sizes

Cluster size can be used to measures similarity of packets represented: rank computation

Manhattan distance’s output space is tractable

DDoS-AID runs at line-rate on existing programmable hardware



DDoS-AID runs at line-rate on existing programmable hardware

Flexible scheduling

Flexible rank computation in the control plane

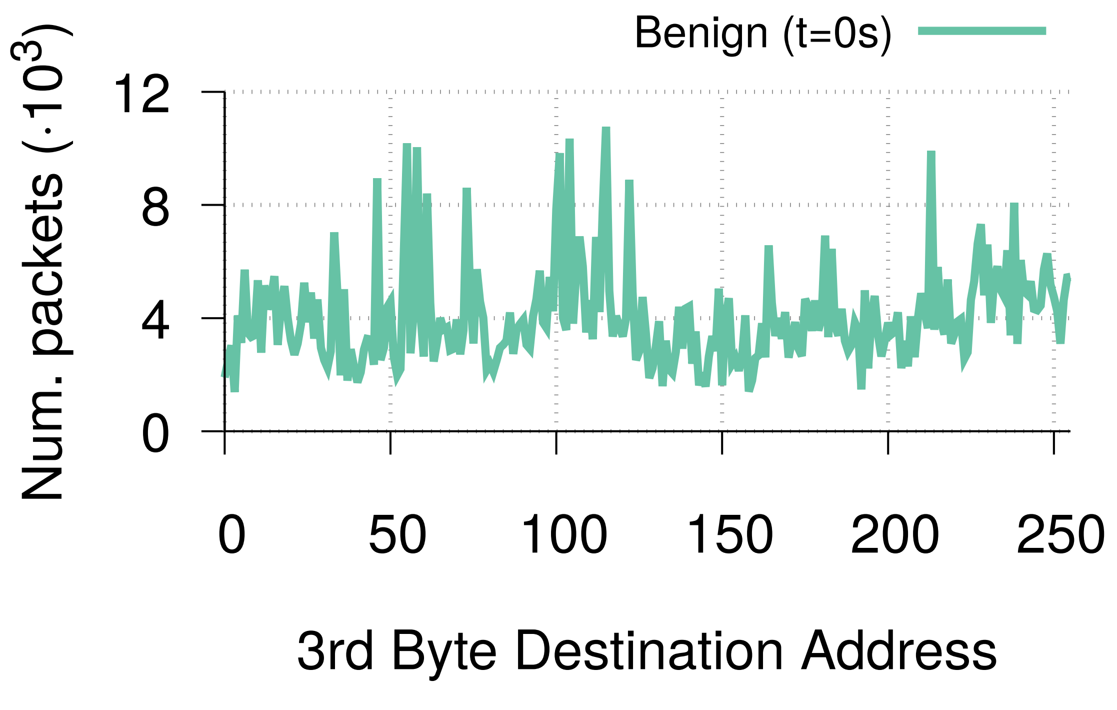
throughput(c_selected)/size(c_selected)

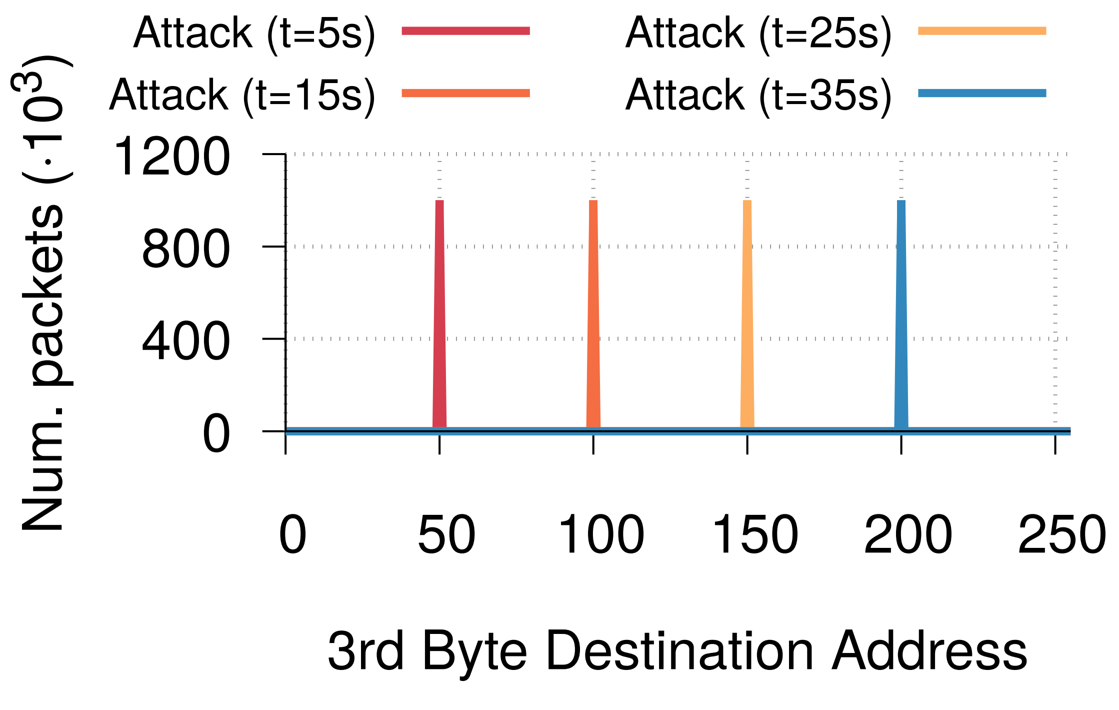
All data plane resources can be dedicated to clustering

Still sub-second reaction time,

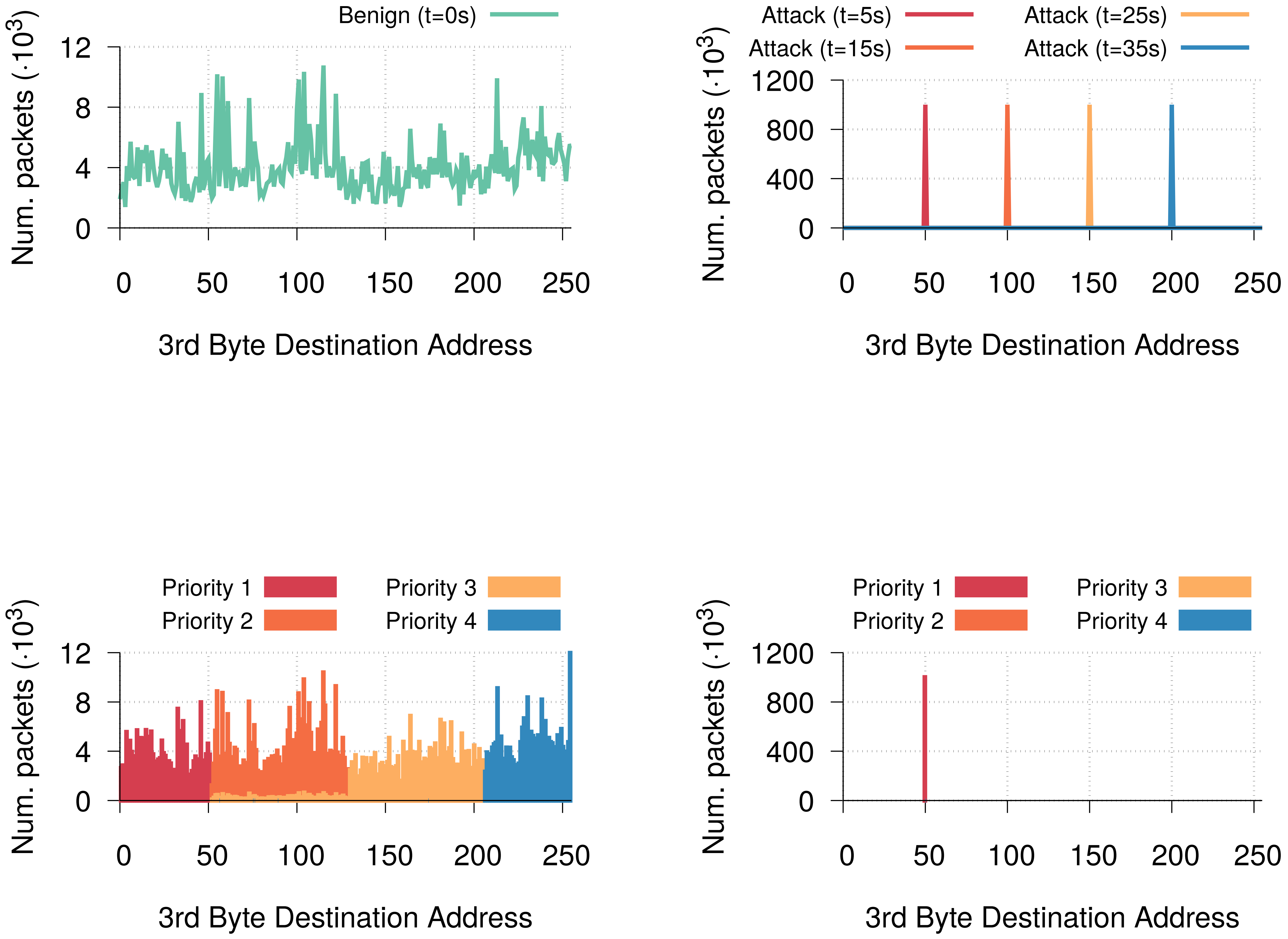
faster than state-of-the-art

DDoS-AID performance increases with the number of clusters



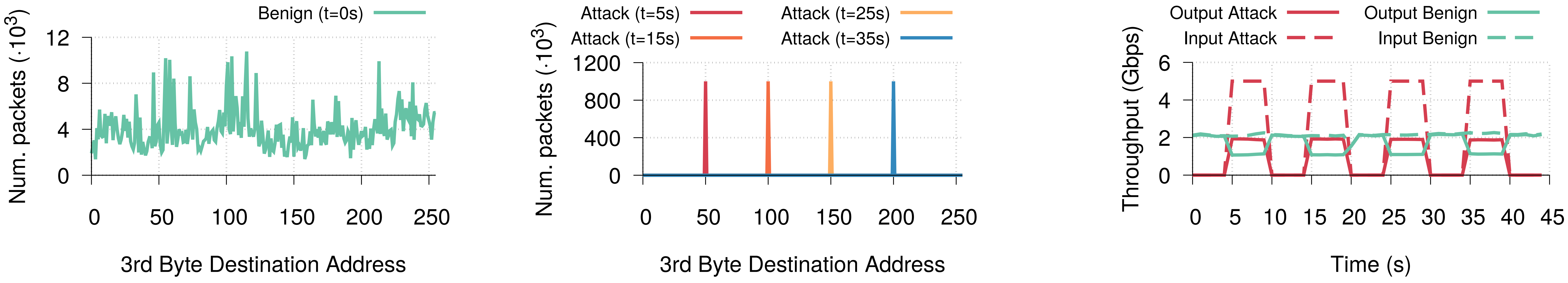


DDoS-AID performance increases with the number of clusters

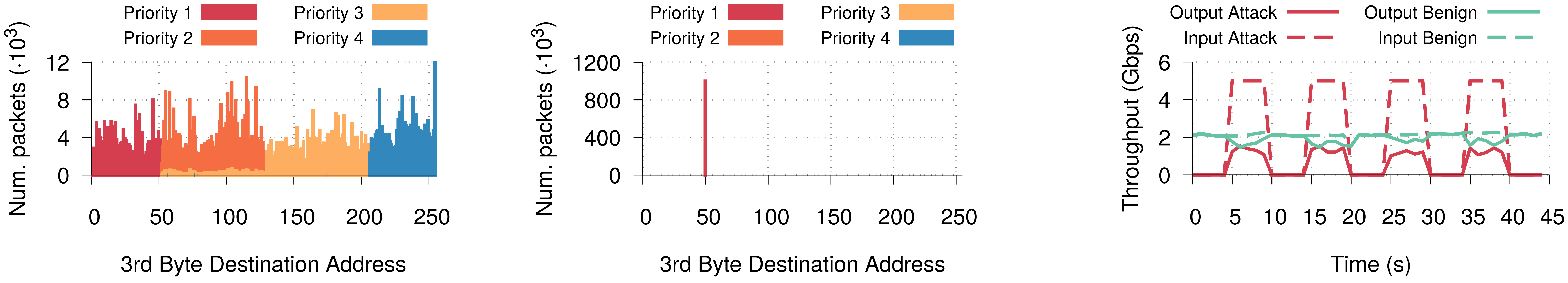


DDoS-AID performance increases with the number of clusters

FIFO

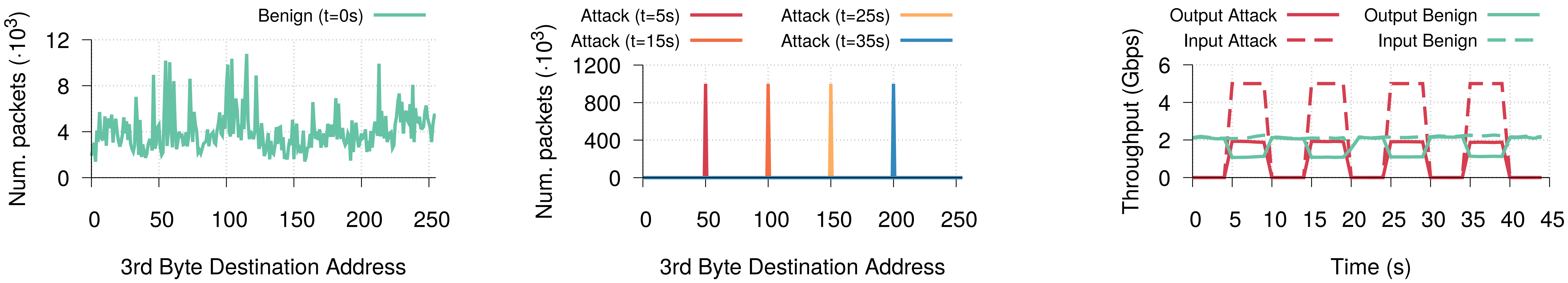


DDoS-AID (4 clusters, 1 feature)

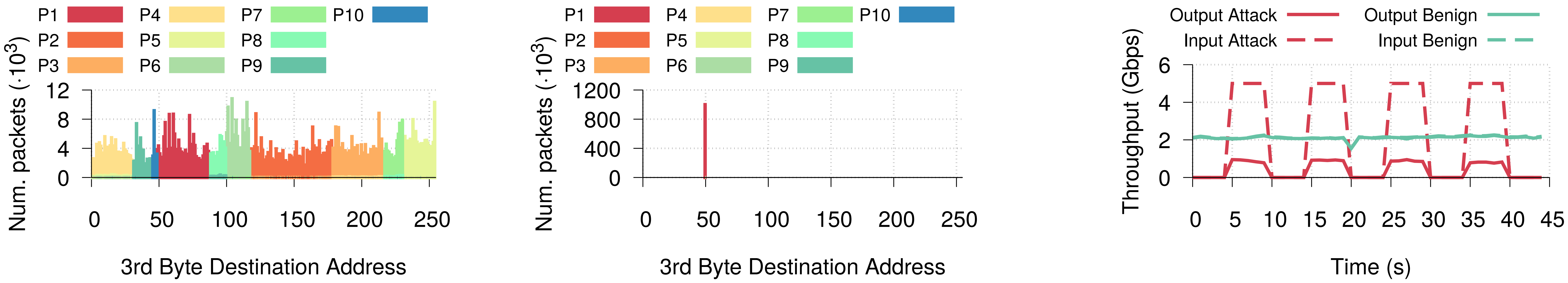


DDoS-AID performance increases with the number of clusters

FIFO



DDoS-AID (10 clusters, all features)


P1
P2
P3
P4
P5
P6
P7
P8
P9
P10
Num. packets ($\cdot 10^3$)
3rd Byte Destination Address
P1
P2
P3
P4
P5
P6
P7
P8
P9
P10
Num. packets ($\cdot 10^3$)
3rd Byte Destination Address
Output Attack
Input Attack
Output Benign
Input Benign
Throughput (Gbps)
Time (s)