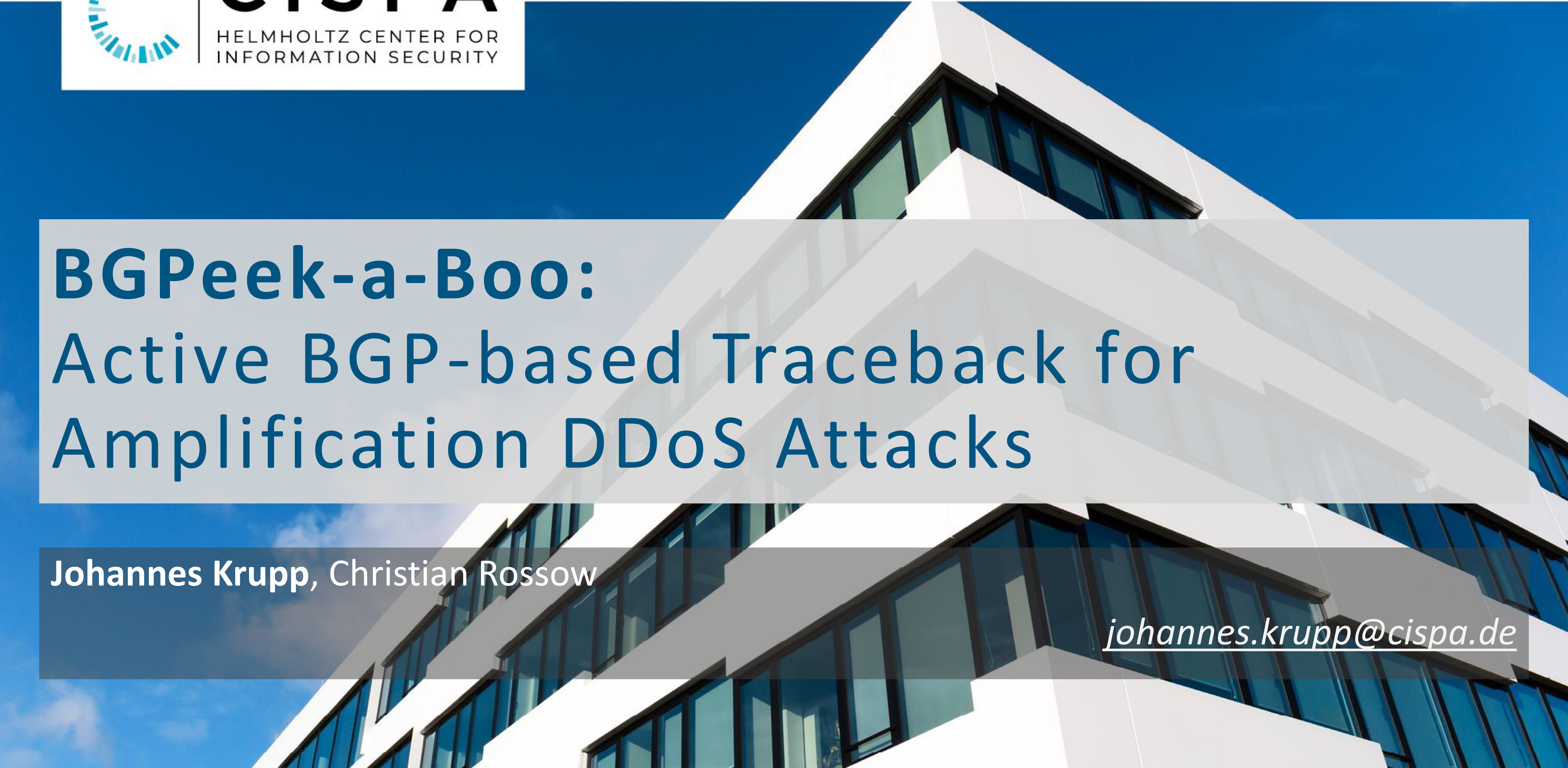
CISPA
HELMHOLTZ CENTER FOR
INFORMATION SECURITY

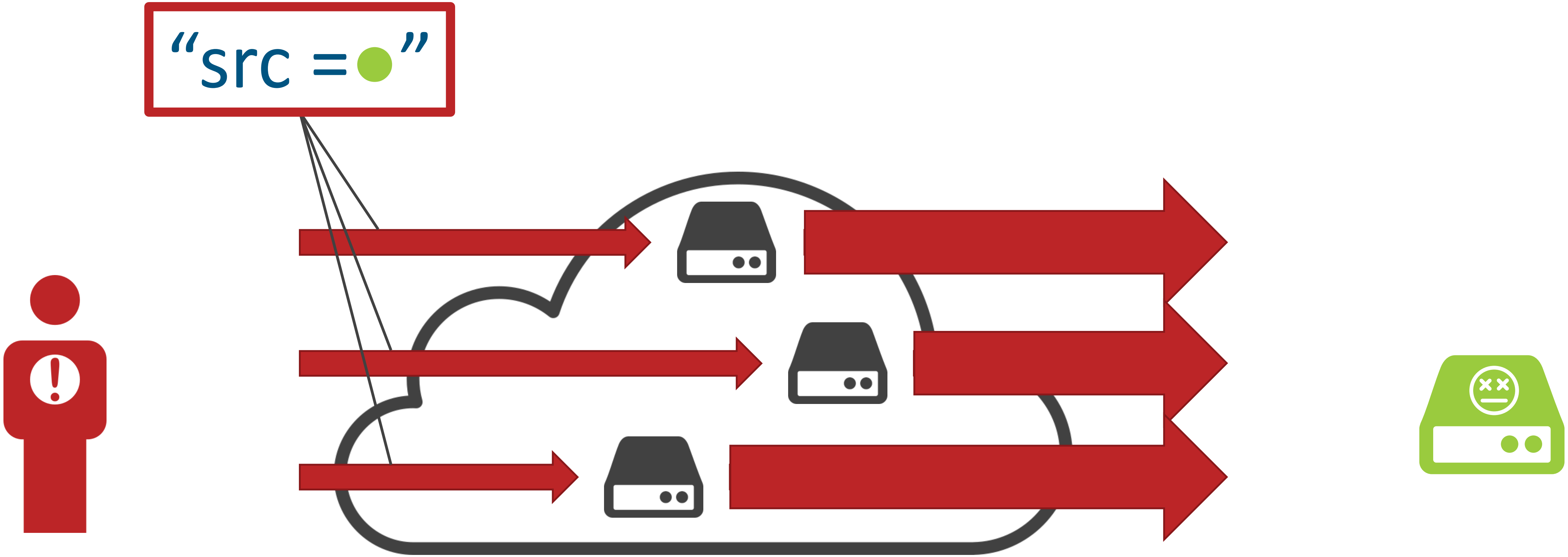


BGPeek-a-Boo: Active BGP-based Traceback for Amplification DDoS Attacks

Johannes Krupp, Christian Rossow

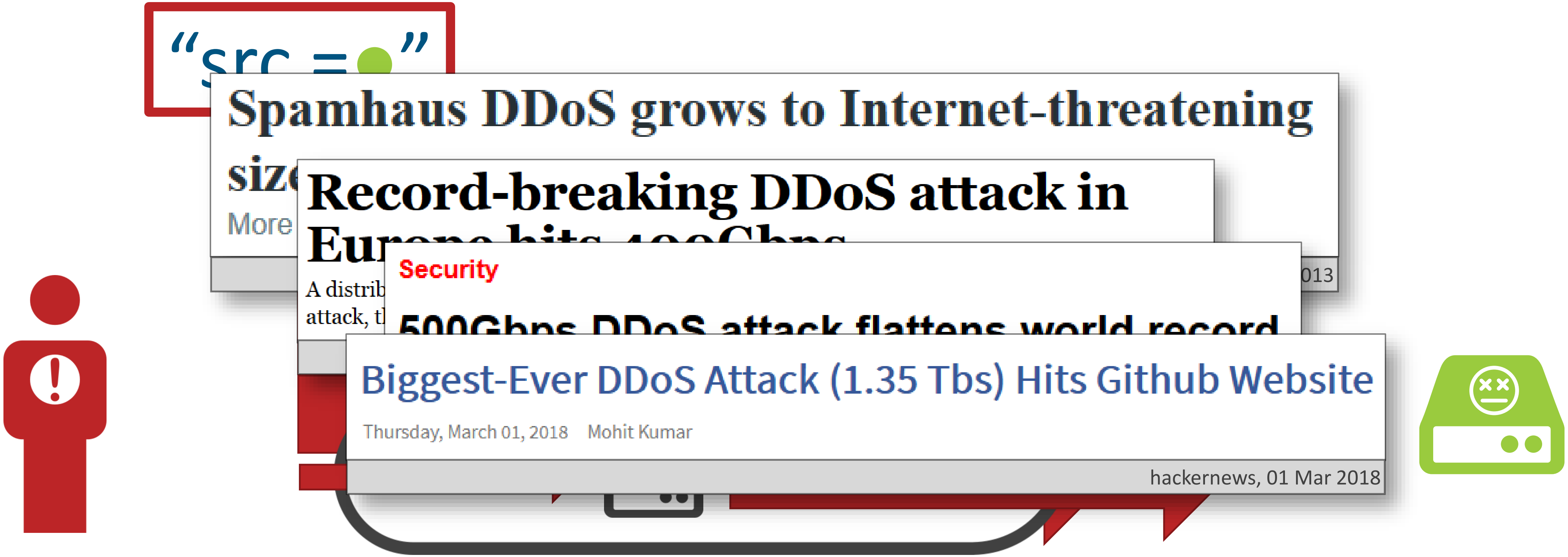
johannes.krupp@cispa.de

Reflective DDoS



“src = ●”

Reflective DDoS



Spamhaus DDoS grows to Internet-threatening siz

More

013

Record-breaking DDoS attack in Europe hits 400Gbps

A distrib

attack, t

Security

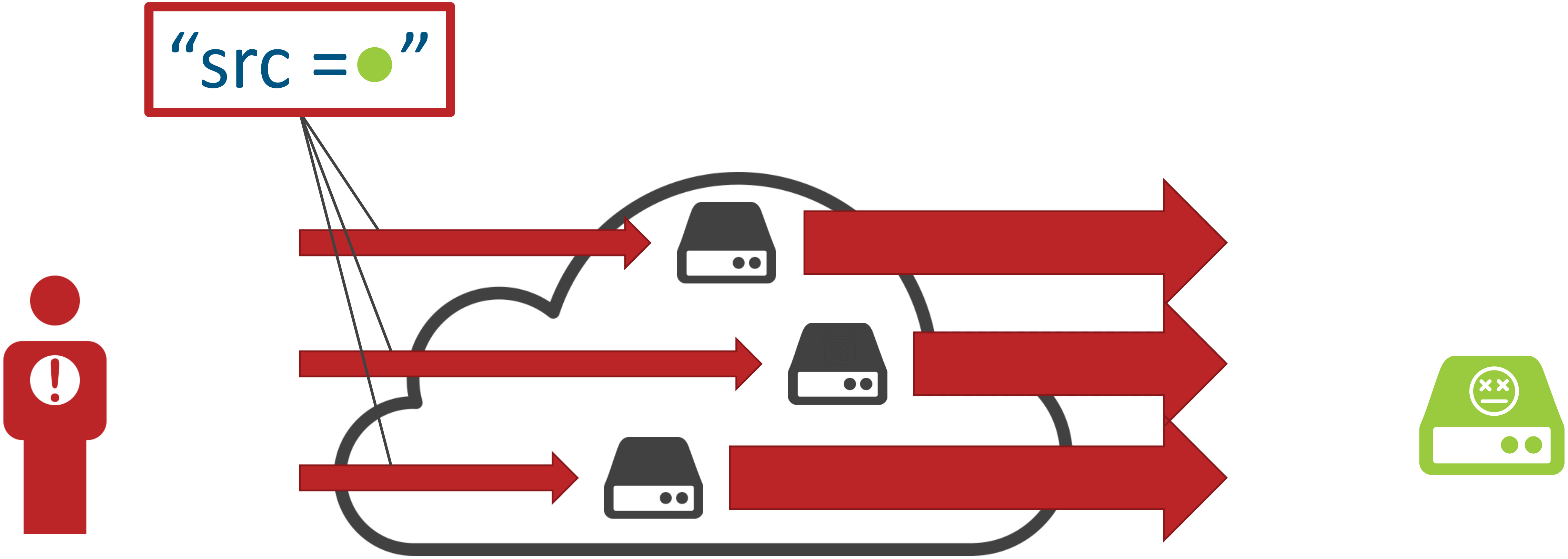
500Gbps DDoS attack flattens world record

Biggest-Ever DDoS Attack (1.35 Tbs) Hits Github Website

Thursday, March 01, 2018 Mohit Kumar

hackernews, 01 Mar 2018

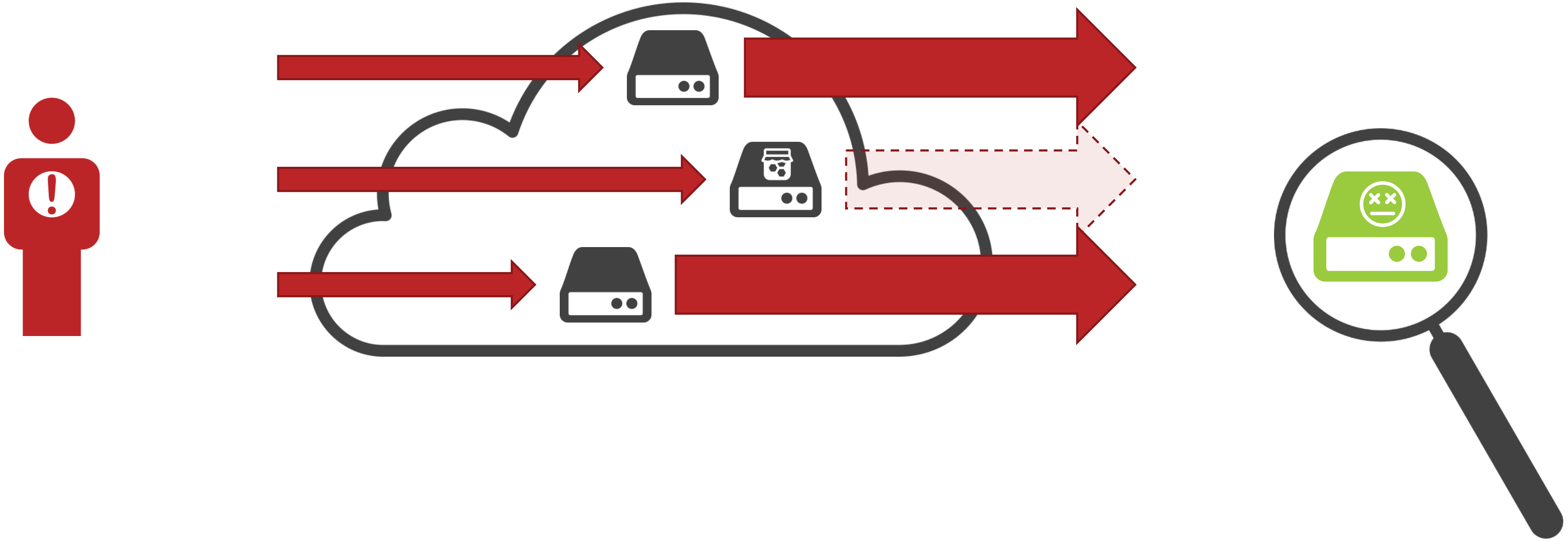
Reflective DDoS - Honeypot



where do attacks come from?

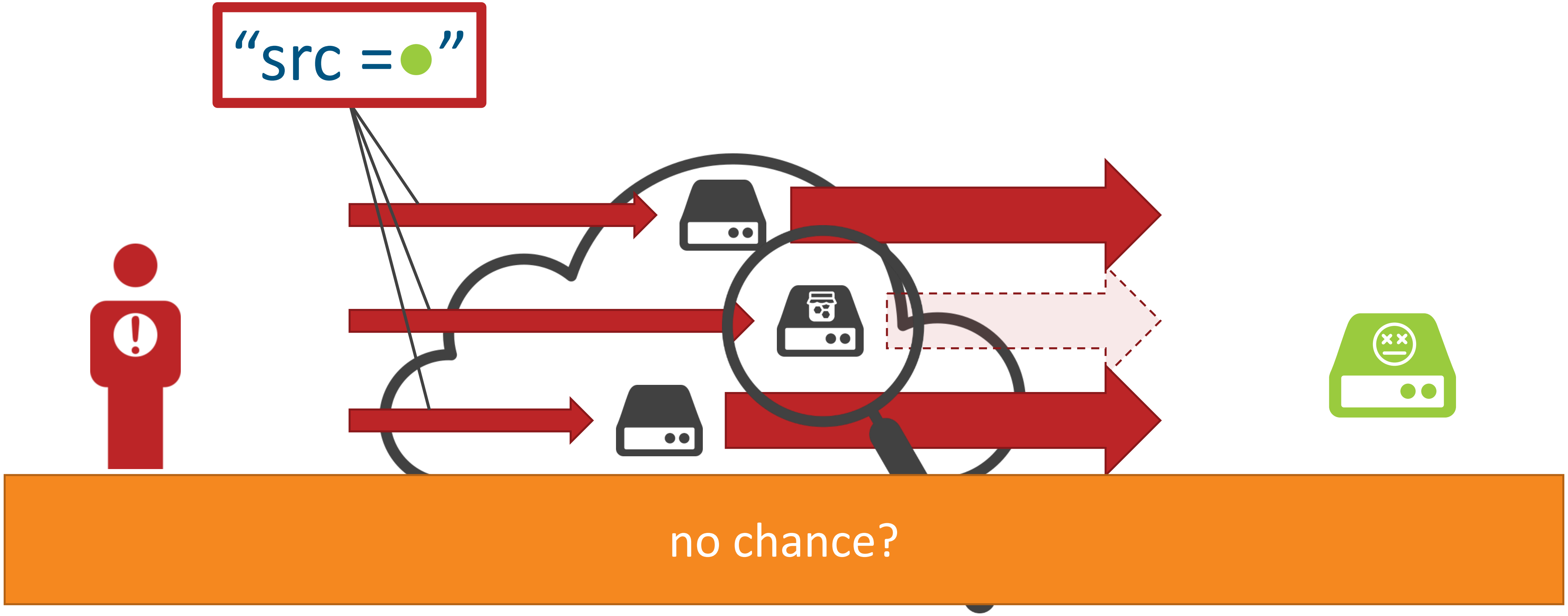
Victim’s Perspective

- Traffic from amplifiers only
- No direct contact with attacker



Amplifier’s Perspective

- Traffic from attacker
- …but with spoofed source only

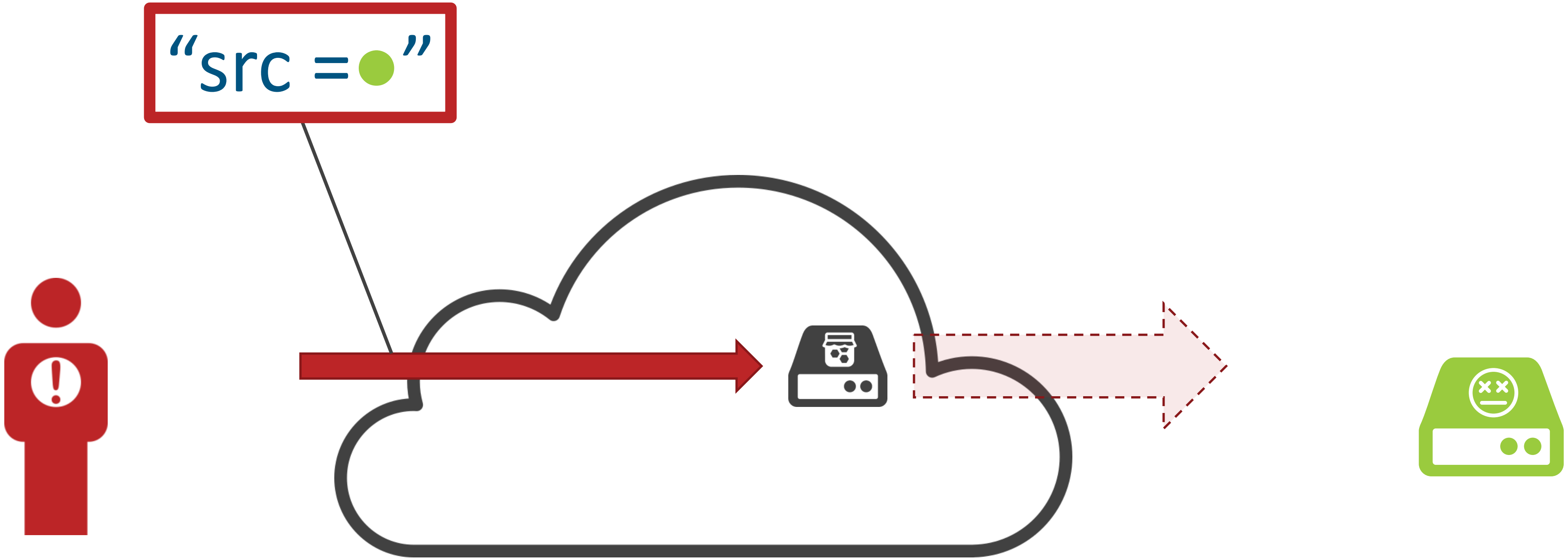


“src =●”

no chance?

Amplifier’s Perspective (network view)

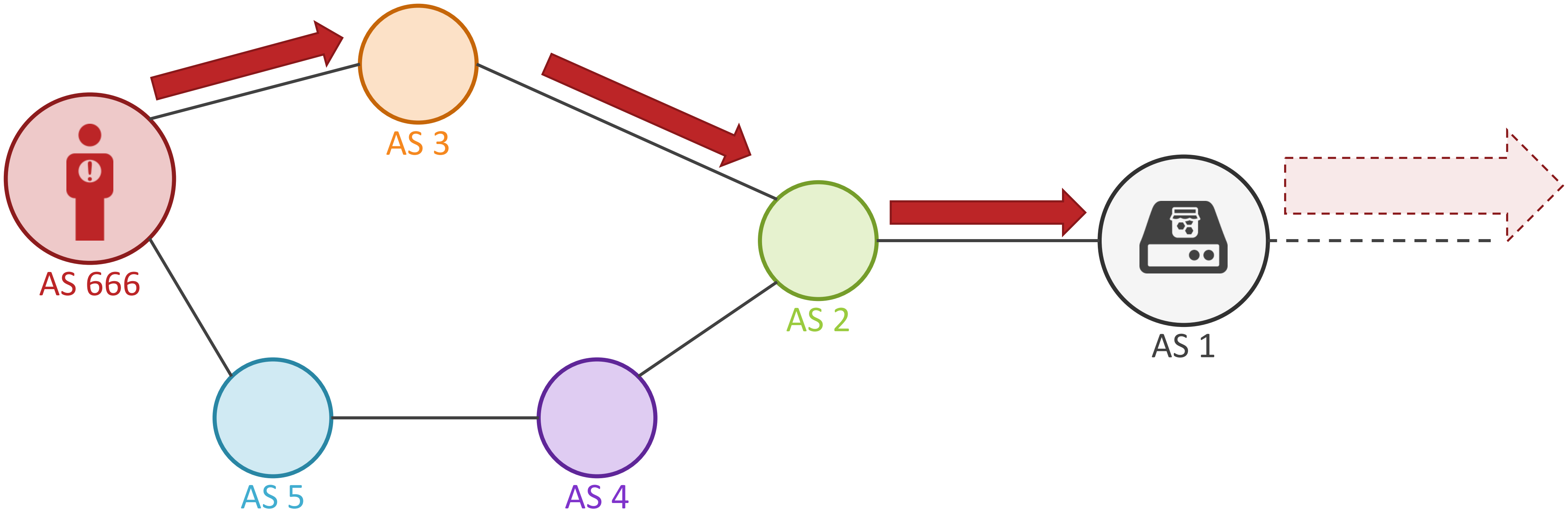
- Traffic from attacker
- …but with spoofed source only
- **but still originating from attacker**



“src =●”

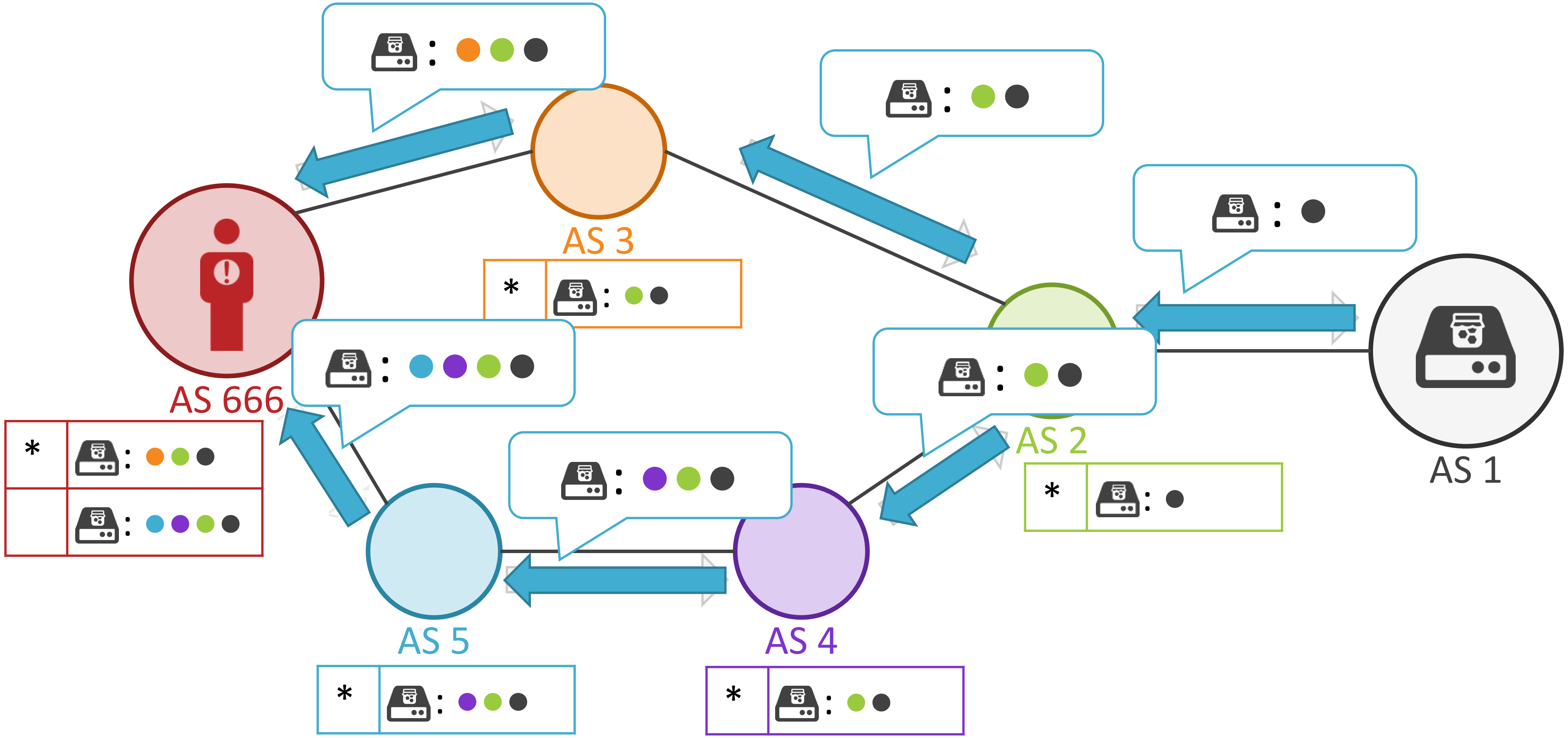
Amplifier’s Perspective (network view)

- Traffic from attacker
- …but with spoofed source only
- **but still originating from attacker**



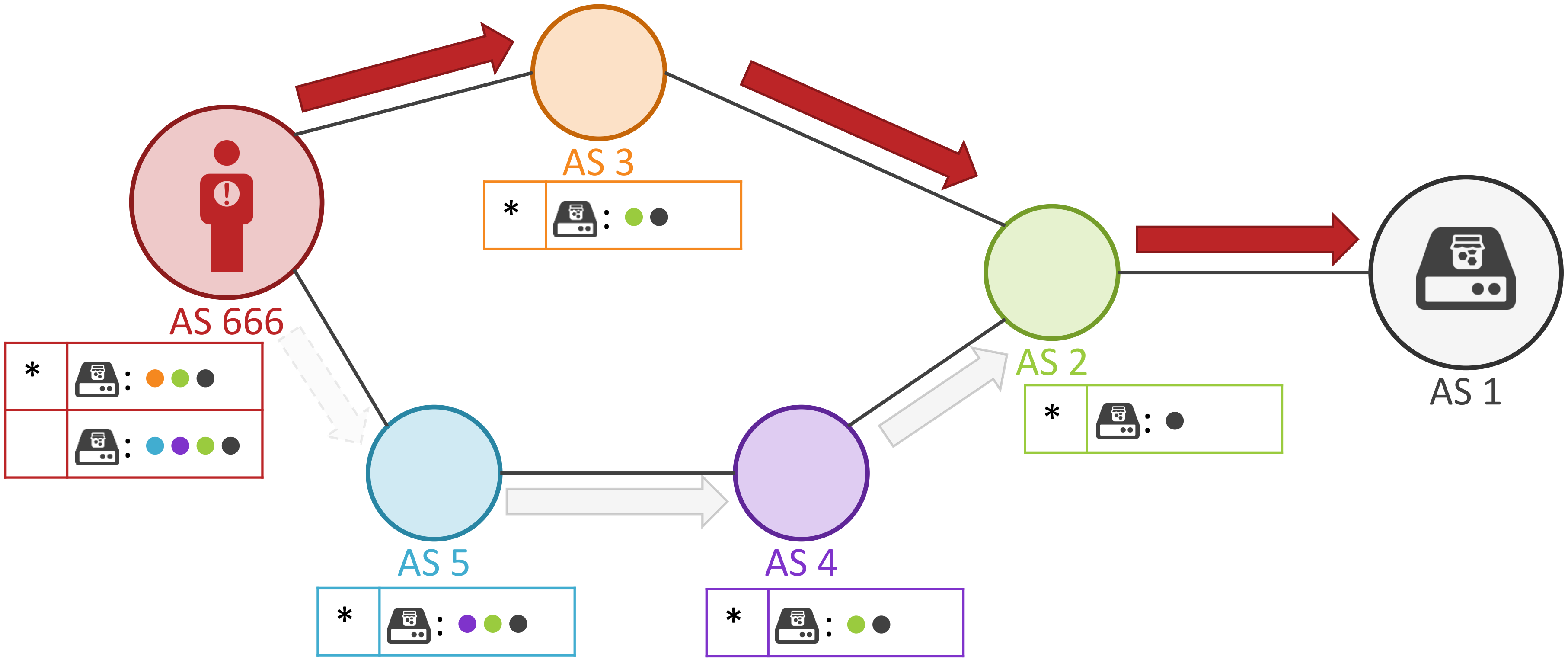
BGP Path Propagation

- How does the attacker system know where to forward traffic to?
  => through BGP



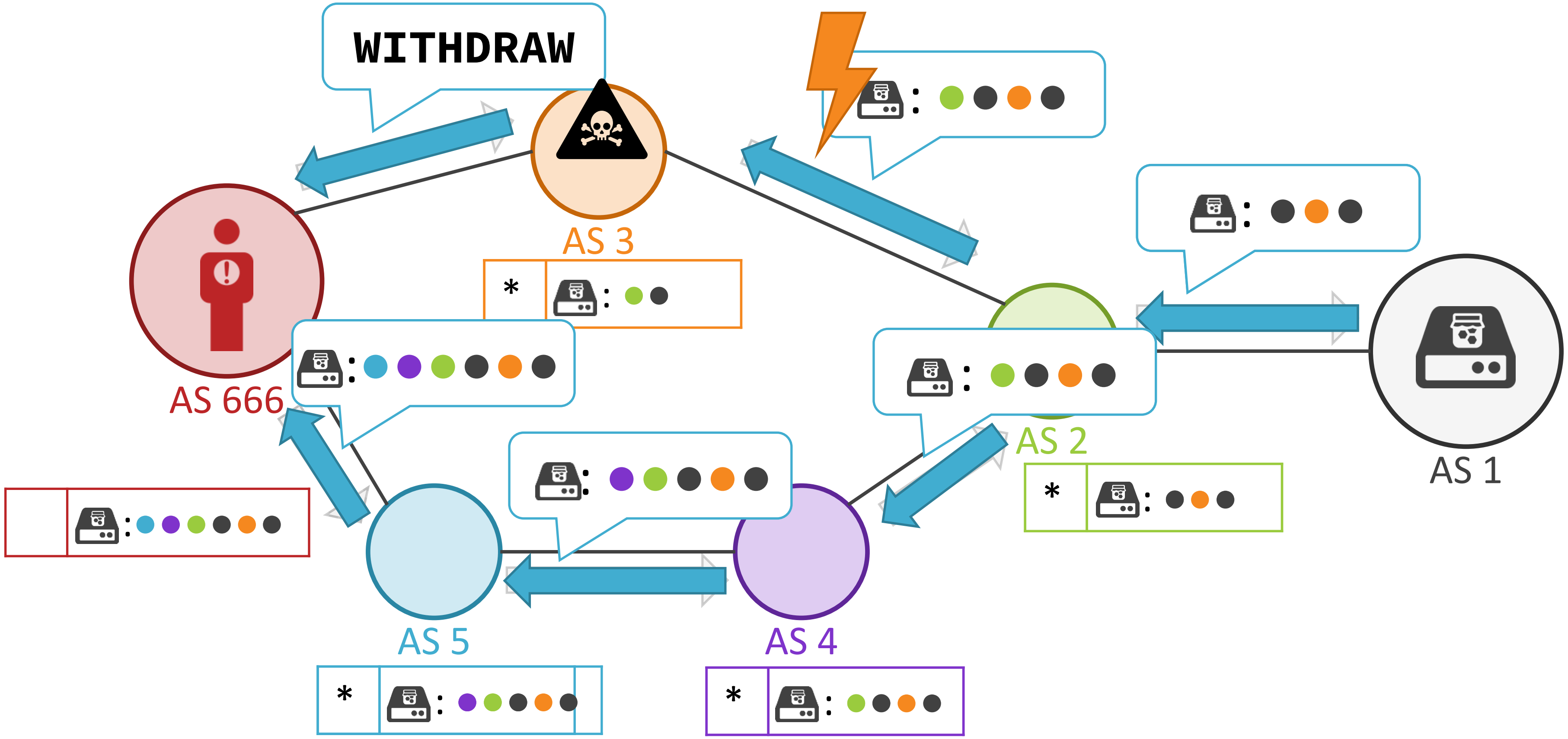
BGP Path Propagation

- How does the attacker system know where to forward traffic to?
  => through BGP



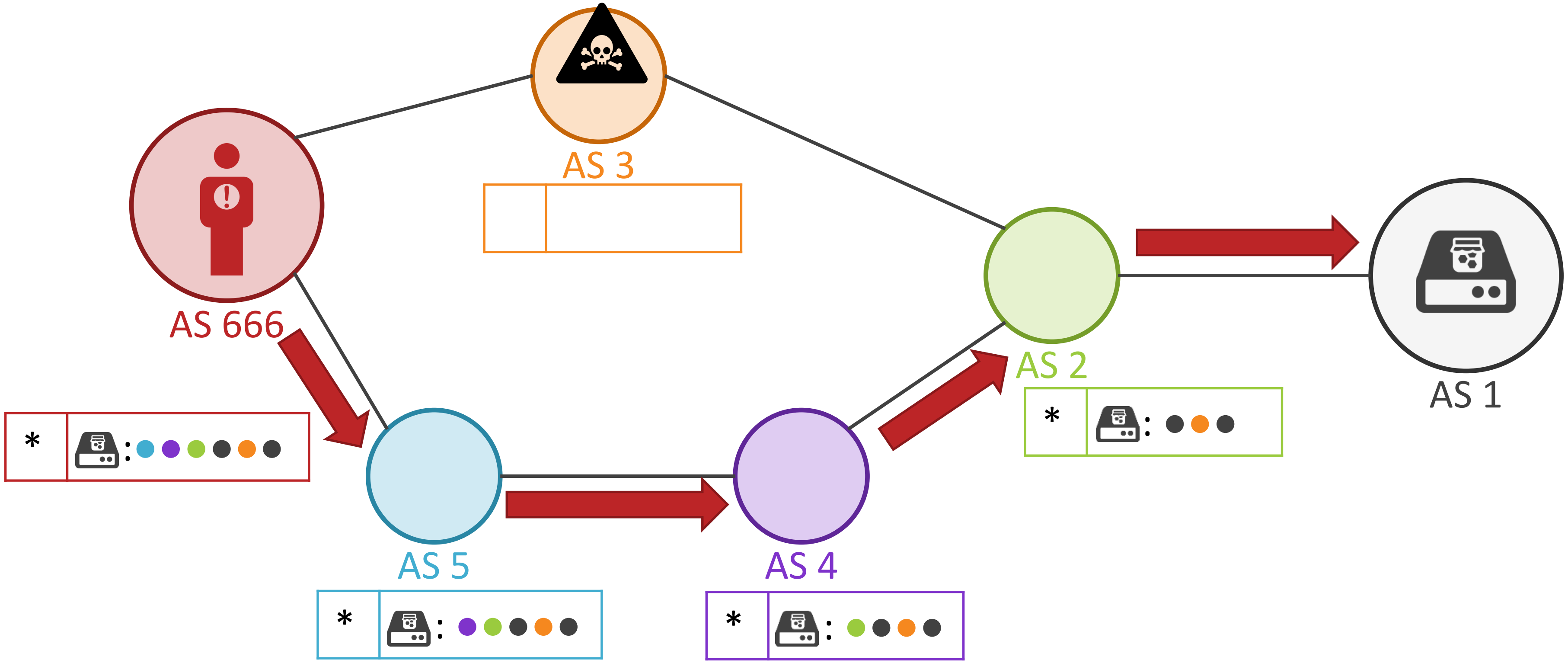
BGP Path Propagation under Poisoning

- How does the attacker system know where to forward traffic to?
  => through BGP
- Can we influence the attacker?
  => Yes, through BGP Poisoning

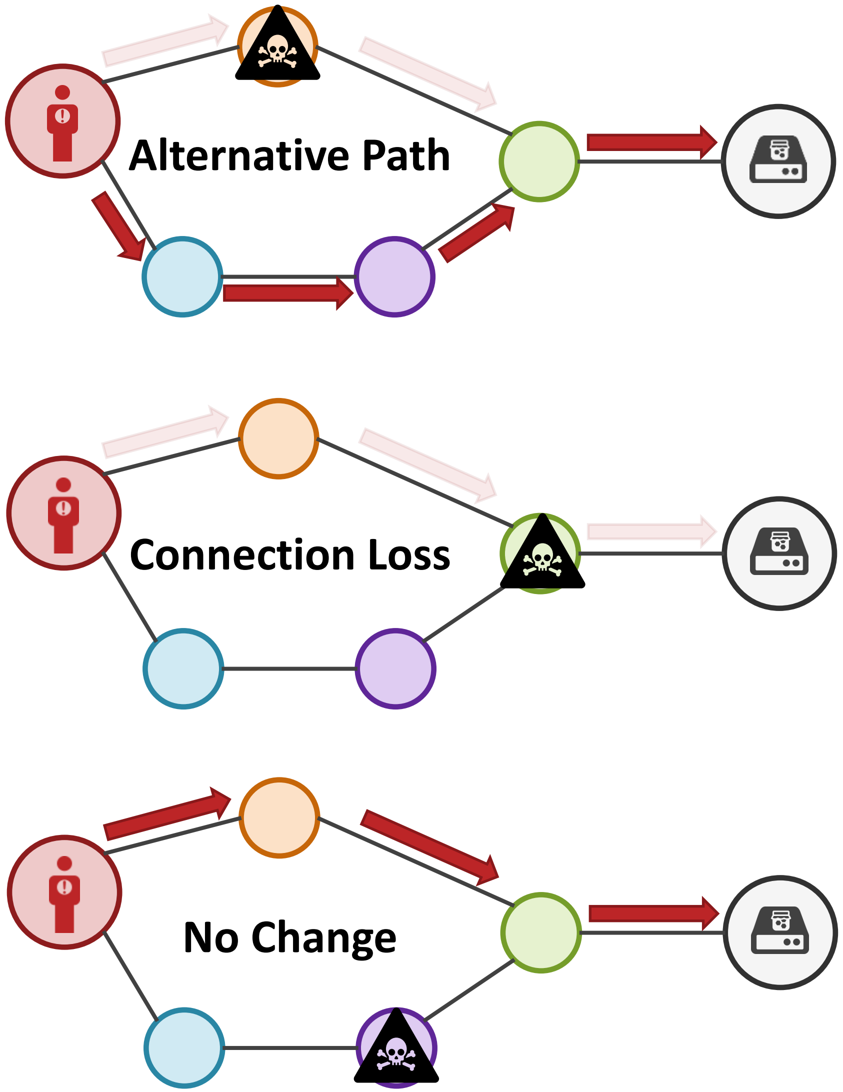


BGP Path Propagation under Poisoning

- How does the attacker system know where to forward traffic to?
  => through BGP
- Can we influence the attacker?
  => Yes, through BGP Poisoning



BGP Poisoning for Attack Traceback



Observable Effect?

Yes (if TTL change)

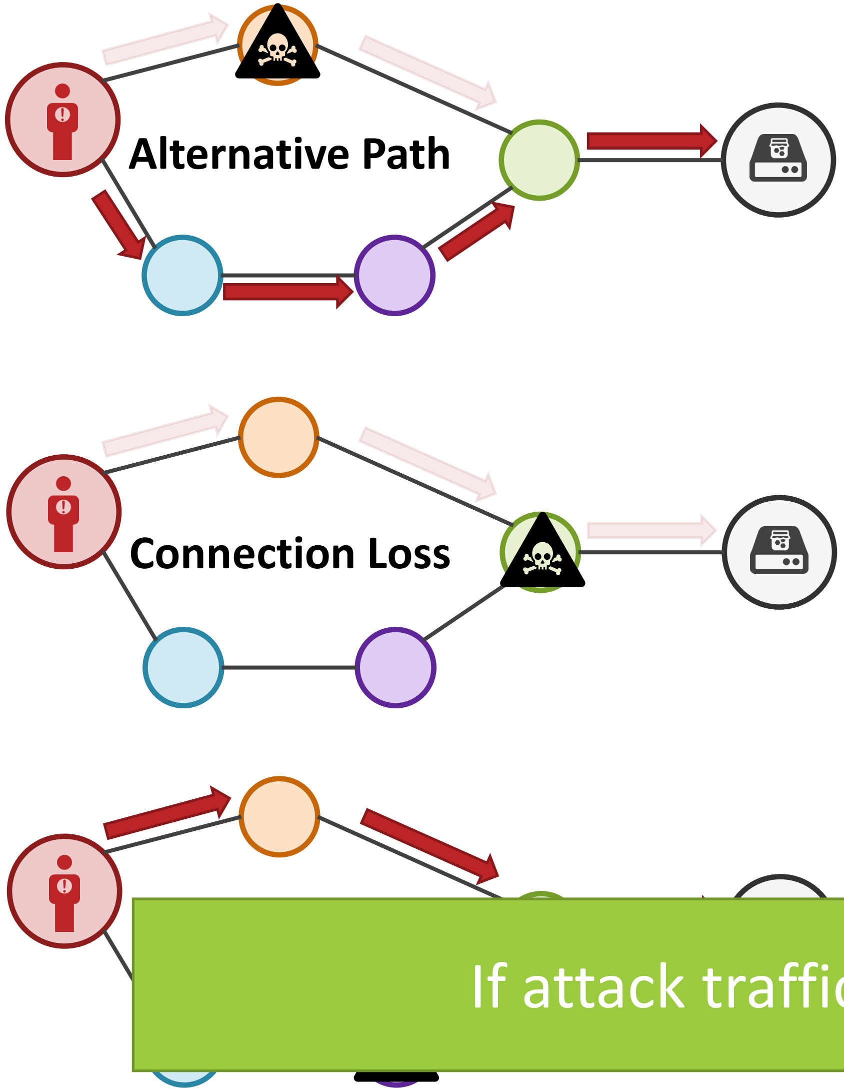
Yes (traffic stops)

No

only if poisoned AS was on original path

BGP Poisoning for Attack Traceback

Observable Effect?



Yes (if TTL change)

Yes (traffic stops)

only if poisoned AS was on original path

If attack traffic changes => poisoned AS was on path

Naive Traceback

```
for every AS A:
  poison A
  if has_effect():
    candidates.add(A)
```

- ~ 70,000 active Ases
- max rate: 6/h

11,667 hours
= 486 days
= 1.3 years

Naive Traceback

```
for every block of ASes P:
  poison P
  if has_effect():
    split P in two parts
    & recurse
```

- shortcut: stop if a stub-AS shows an effect (no customers => must be traffic origin)

- ~ 70,000 active Ases
- max rate: 6/h
- poison 128 ASes in parallel
  - logarithmic split&recurse overhead

91.1 hours
= 3.8 days

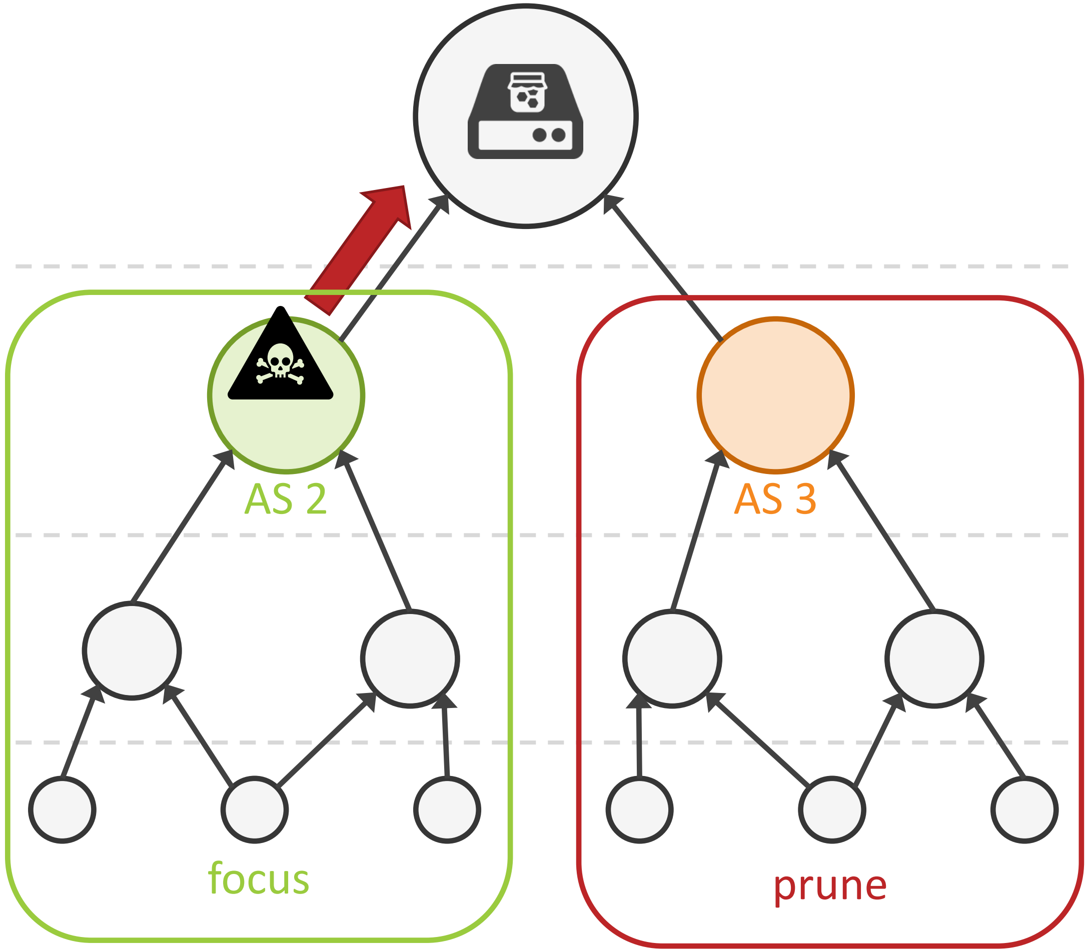
Feasible

Can we do even better?

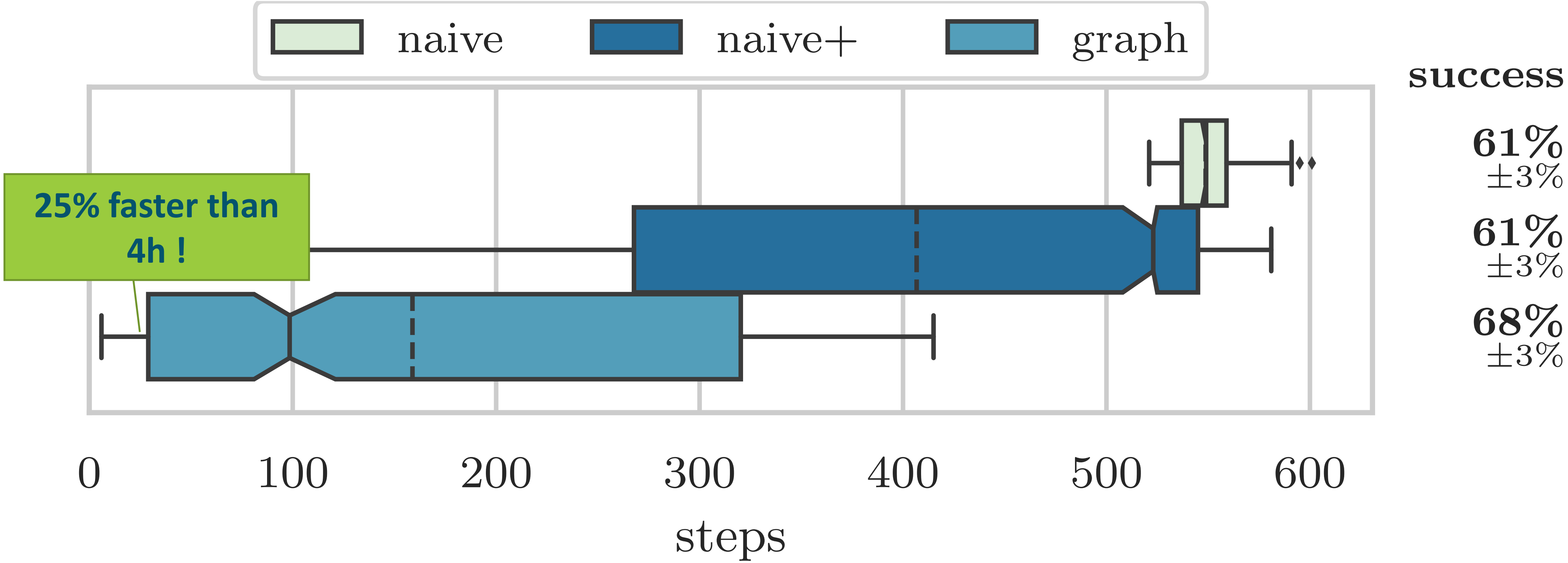
Graph-based Traceback

- create rooted directed graph over ASes
  - root: 🖴
  - edge AS1 → AS2:
AS1 can have AS2 as next-hop
- use graph to
  - search in layers
  - prune search
- **requires accurate AS relationship data**

large parts pruned
= dramatic speed-up

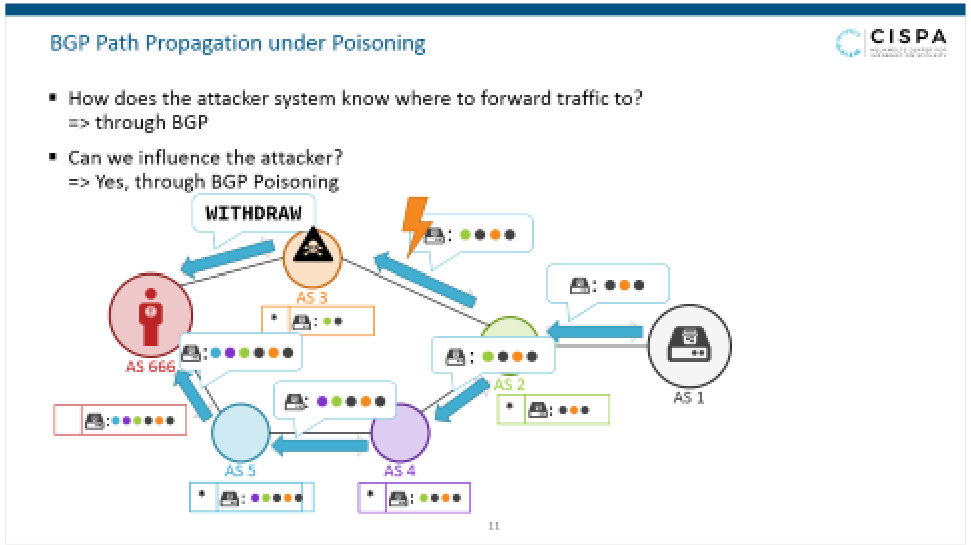


Evaluation – Results



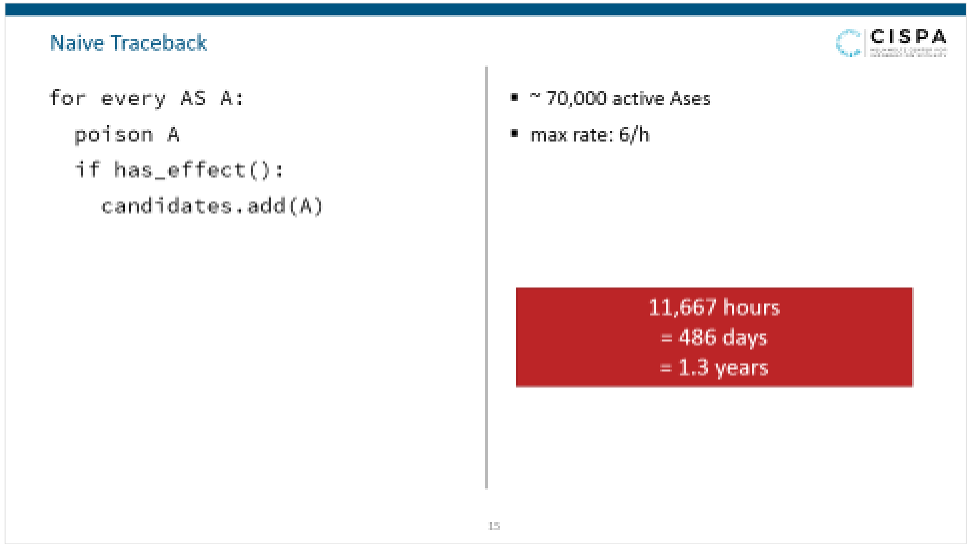
25% faster than 4h !

Conclusion



BGP Path Propagation under Poisoning

- How does the attacker system know where to forward traffic to?
  => through BGP
- Can we influence the attacker?
  => Yes, through BGP Poisoning

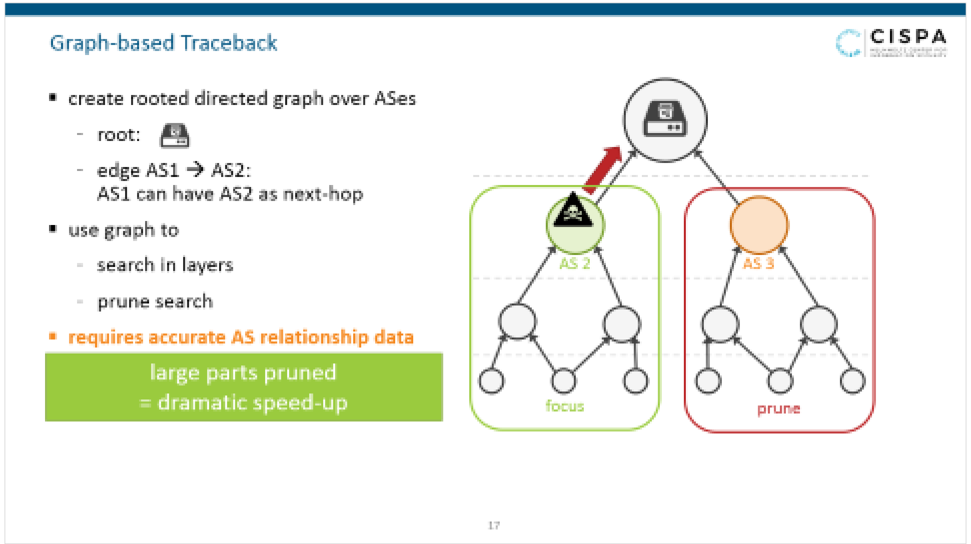


Naive Traceback

```
for every AS A:
  poison A
  if has_effect():
    candidates.add(A)
```

- ~ 70,000 active Ases
- max rate: 6/h

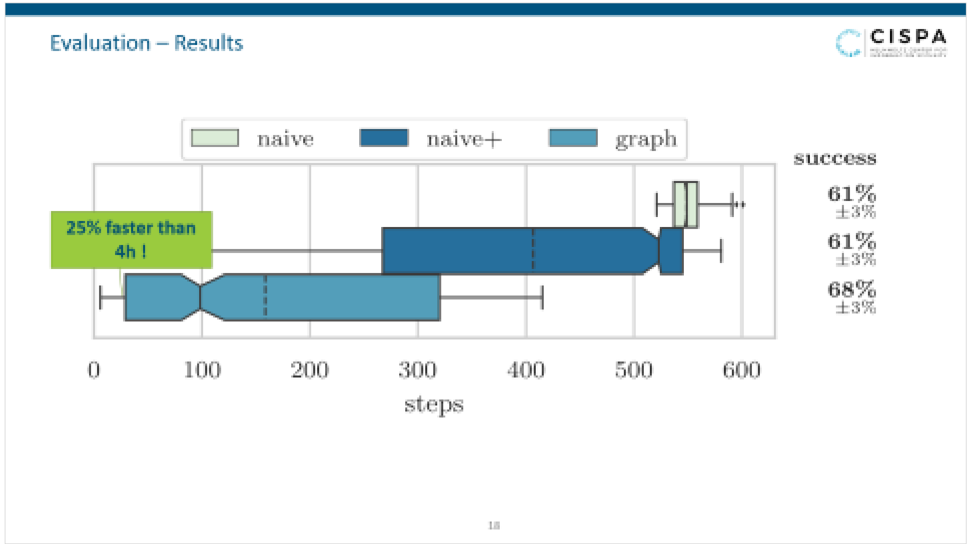
11,667 hours
= 486 days
= 1.3 years



Graph-based Traceback

- create rooted directed graph over ASes
  - root:
  - edge AS1 → AS2:
    AS1 can have AS2 as next-hop
- use graph to
  - search in layers
  - prune search
- requires accurate AS relationship data

large parts pruned
= dramatic speed-up



Evaluation – Results

25% faster than 4h !