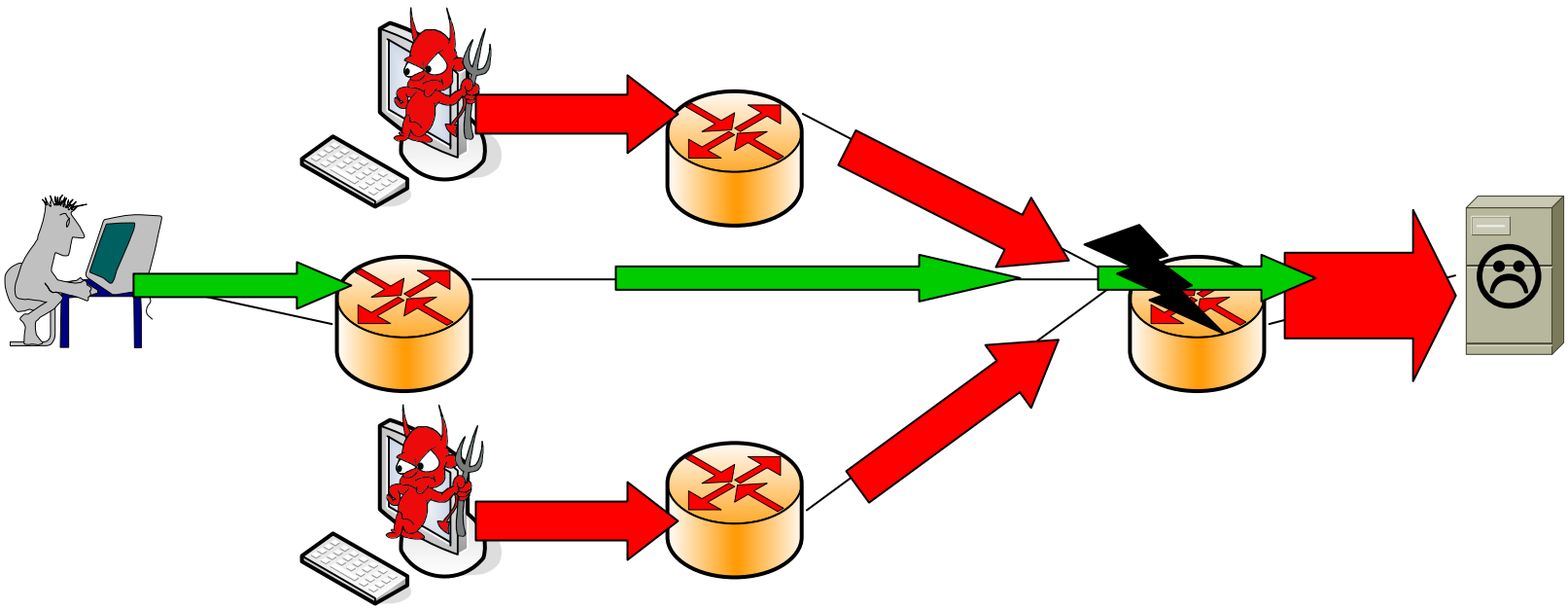
To Filter or to Authorize: Network-Layer DoS Defense against Multimillion-node Botnets

Xiaowei Yang
Duke Unversity

Denial of Service (DoS) flooding attacks



- Send packet floods to a targeted victim
- Exhaust shared resources
  - Bandwidth, memory, or CPU time

Most newsworthy weakness of the Internet

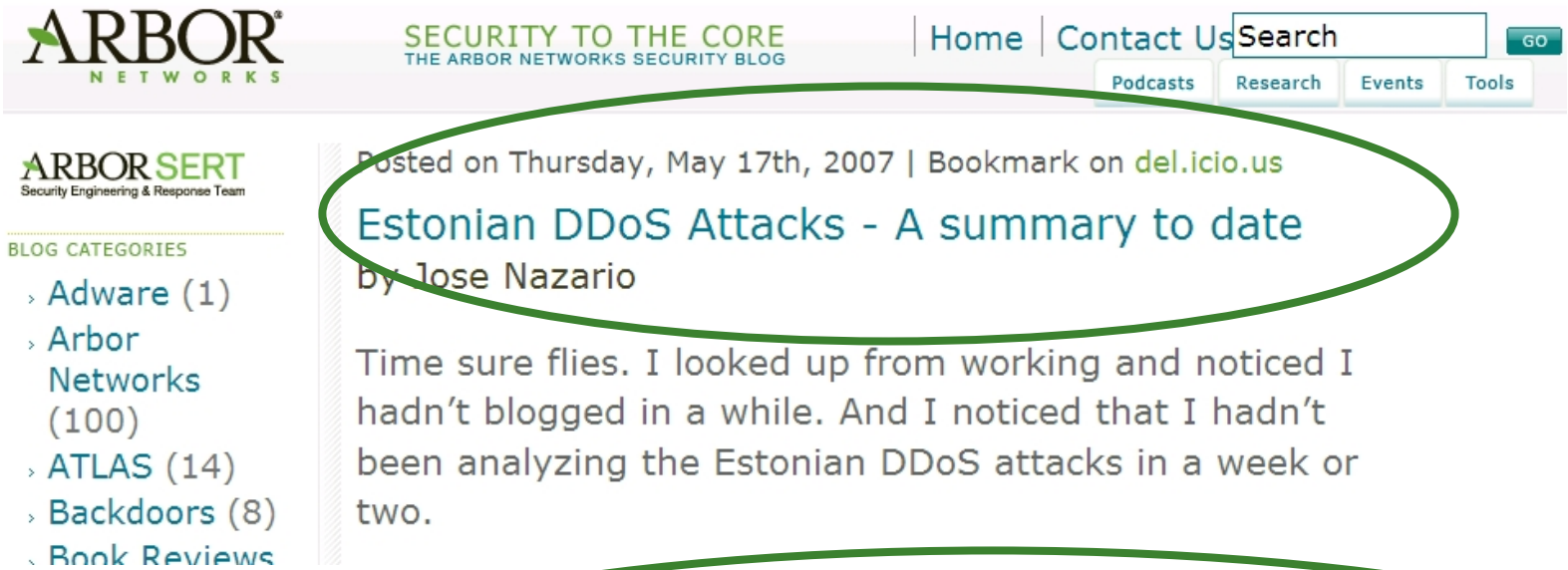
WikiLeaks Under Fire

Posted by Soulskill on Monday February 18, @08:15AM
from the **also-possibly-on-fire** dept.

kan0r writes

> "The transparency group WikiLeaks.org currently seems to be under heavy fire. The main WikiLeaks.org DNS entry is unavailable, reportedly due to a restraining order relating to a series of articles and documents released by WikiLeaks about off-shore trust structures in the Cayman Islands. The WikiLeaks whistle blower, allegedly former vice president of the Cayman Islands branch of swiss bank Julius Baer, states in the WikiLeaks documents that the bank supported tax evasion and money laundering by its clients from around the world. WikiLeaks alternate names remained available until Saturday, when there seems to have been a heavy DDoS attack and a fire at the ISP. The documents in question are still available on other WikiLeaks sites, such as wikileaks.be, and are also mirrored on Cryptome. Details of the court documents have also been made available."

Anyone can be a victim



Lucrative

BBC News 24

Last Updated: Friday, 19 March, 2004, 12:35 GMT

E-mail this to a friend Printable version

Bookies suffer online onslaugh

By Mark Ward
BBC News Online technology correspondent

The extent to which British betting websites are being attacked by criminals using the net to bring down a site unless a ransom is paid has been revealed by a BBC News Online investigation.

again.

Whitepapers Guides and Reports Webcasts Podcasts Videos

NetworkWorld.com > Security >

Extortion via DDoS on the rise

By Denise Pappalardo and Ellen Messmer, Network World, 05/16/05

Start a discussion Print article

Criminals are increasingly targeting corporations with distributed denial-of-service attacks designed not to disrupt business networks but to extort thousands of dollars from the companies.

Those targeted are increasingly deciding to pay the extortionists rather than accept the consequences, experts say. While reports

Other stories on this topic

Russian hosting network runs a protection racket
2/20/2008

May 2005

The Extortion Problem

We know this about online extortion: It happens. Evidence of its prevalence or damage is speculative and anecdotal but useful nonetheless in guiding CSOs to understand the nature of the crime. Anecdotally, experts from law enforcement and information security consultants believe that perhaps one in 10 companies has been threatened with online extortion; in one survey by Carnegie Mellon University researchers, 17 out of 100 small and midsize businesses reported being targeted. Interviews with security consultants and industry players suggest that as many as three

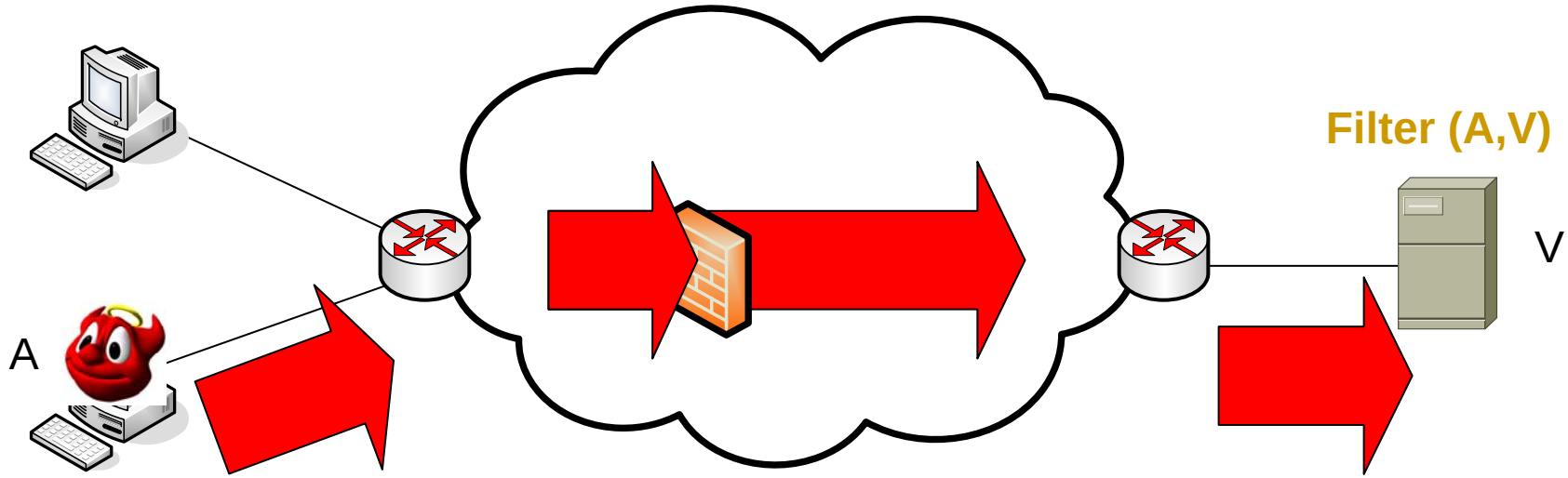
- Buyout is cheaper

No Consensus on How to Combat DoS

- Many proposals to mitigate DoS flooding attacks
  - Mayday, AITF, Flow-Cookies, Phalanx, SOS, Pushback, dFence, Portcullis, OverDoSe, CenterTrack, Defense-by-Offense, FastPass, SIFF, TVA, …
- Two intriguing schools of thought
  - Filters
  - Capabilities

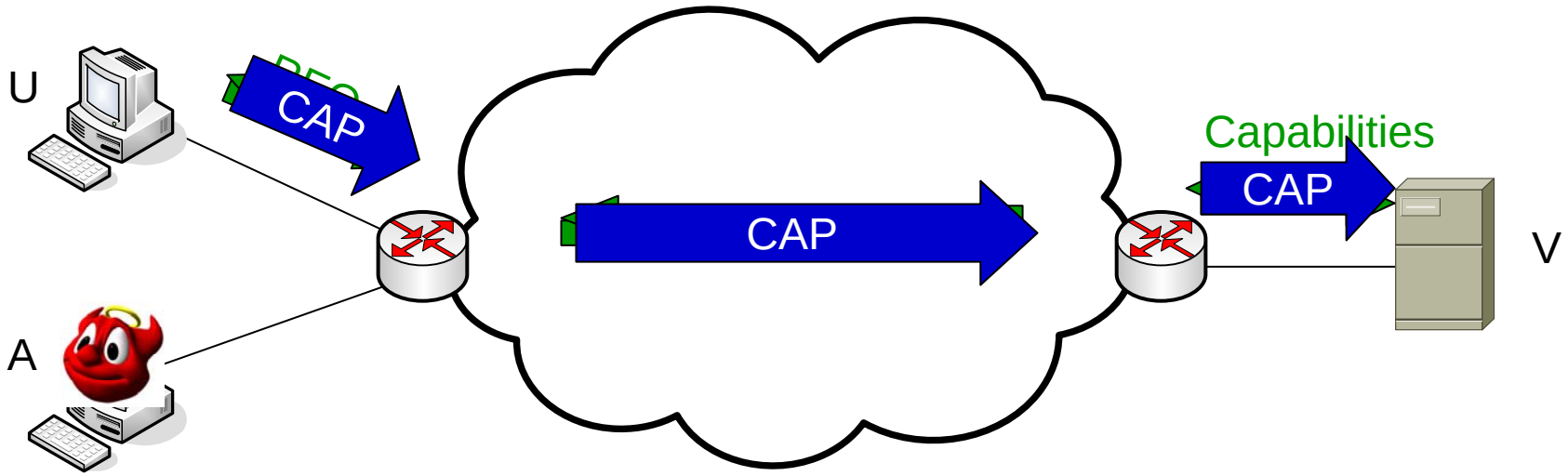
Filter-based Approach

1. Anyone can send to anyone by default
2. A receiver requests the network to install filters



Capability-based Approach [TVA]

1. Source requests permission to send
2. Destination authorizes source for limited transfer
3. Source places capabilities on packets and sends them
4. Network filters packets based on capabilities



Goal of This Work

"…capabilities are neither sufficient nor necessary to combat DoS."

by K. Argyraki, et al.

"We strongly disagree: … a simple and highly efficient network-based defense … can prevent DoC attacks."

by A. Perrig, et al.

To design a DoS-resistant network architecture, should we use filters, capabilities, neither, or both?

Our Approach

“We believe in: rough consensus and running code.”
-- David Clark

1. Design an effective filter-based system
   - Existing filter systems have several limitations
     - Loss of control messages
     - Filter exhaustion attacks
     - Damage when filters fail to install
2. Compare the effectiveness of filter-based and capability-based systems under various attacks

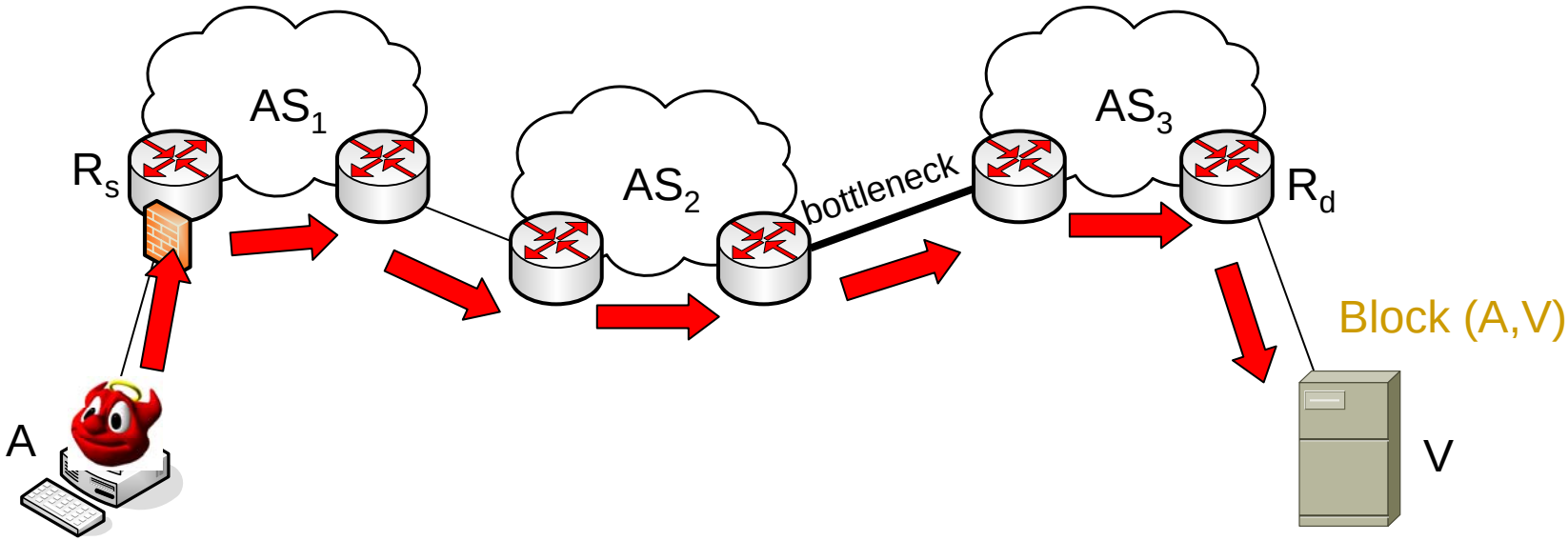
Design Goals of StopIt

- Effective with little collateral damage
  - Do not block legitimate communications
- Resilient to a wide range of strategic attacks
  - E.g.: impersonation attacks, filter exhaustion attacks
- Fail-safe
  - Limit the damage when filters fail to install
- Incentivizing deployment
  - Early adopters should benefit immediately

Design Premises

- Similar to capability-based systems
- Simplifying assumptions
  - End systems can distinguish attack traffic
  - Both routers and hosts can be upgraded
  - Securable intra-AS communications
- Practical constraints
  - No special hardware
    - E.g.: no tamper-proof hardware, no line-speed per-packet public key operations
  - Both hosts and routers may be compromised

Overview of an Ideal Filter System



Scalable: no per-flow state in the network core

Secure the Basic Design

<table>
<tr><th>Problems</th><th colspan="2">Solutions</th></tr>
<tr><td>Source address spoofing attacks</td><td colspan="2">Authenticate source addresses with Passport [NSDI’08]</td></tr>
<tr><td>Impersonation attacks</td><td>Authenticate filter requests with standard authentication techniques</td><td rowspan="2">Closed control channel</td></tr>
<tr><td>Filter exhaustion attacks</td><td>Confirm attacks before accepting filter requests; avoid filters against compliant sources; catch and punish misbehaving sources</td></tr>
<tr><td>Control channel DoS attacks</td><td rowspan="3" colspan="2">Source-based fair queuing</td></tr>
<tr><td>Filters fail to install</td></tr>
<tr><td>Incentives to deploy</td></tr>
</table>

Main challenges of Passport

	Secure	Lightweight	Adoptable
Ingress filtering	✘	✓	✘
Digital signature	✓	✘	✓
Passport	✓	✓	✓

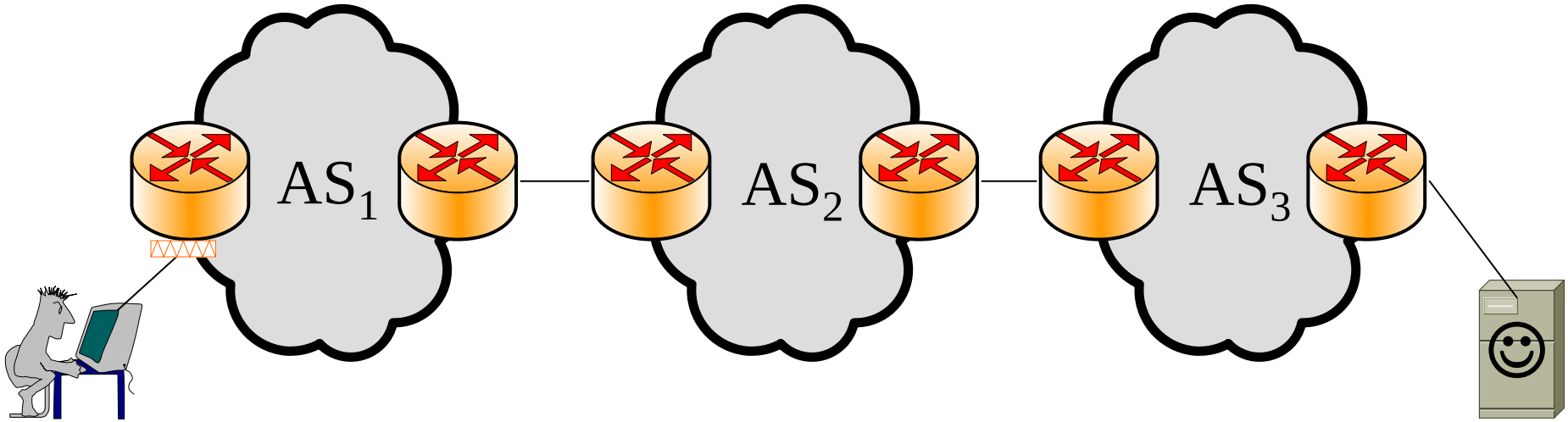
- Ingress filtering
  - One weak link allows spoofing
    - Spoofer shows ~20% of the Internet can spoof
  - An early adopter can't protect its own address space

- Digital signature
  - PKI, time-consuming to stamp and verify, large header overhead

Passport mechanisms

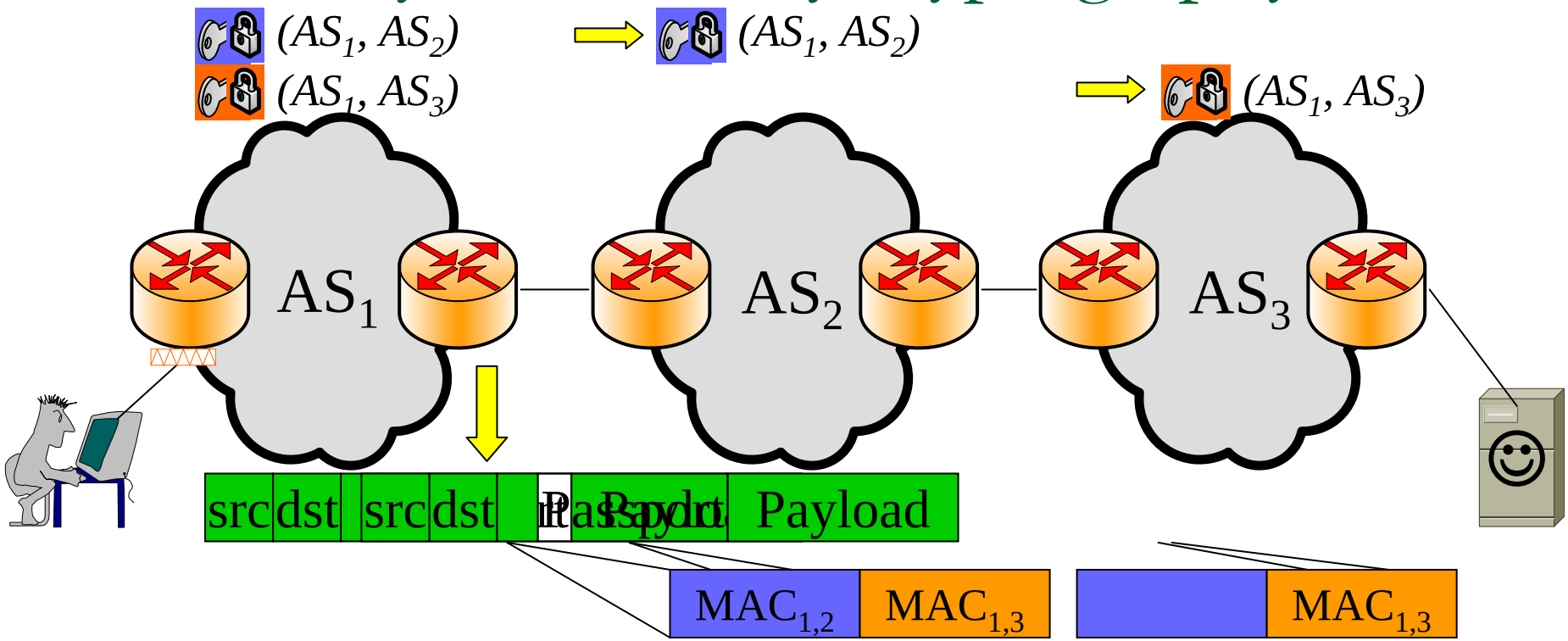
- Symmetric key cryptography
  - Efficient, secure
- Use routing to distribute keys
  - Bootstrap, efficient, simple
- AS-level (autonomous system) fate sharing
  - Scalable, incentive compatible

AS-level fate sharing



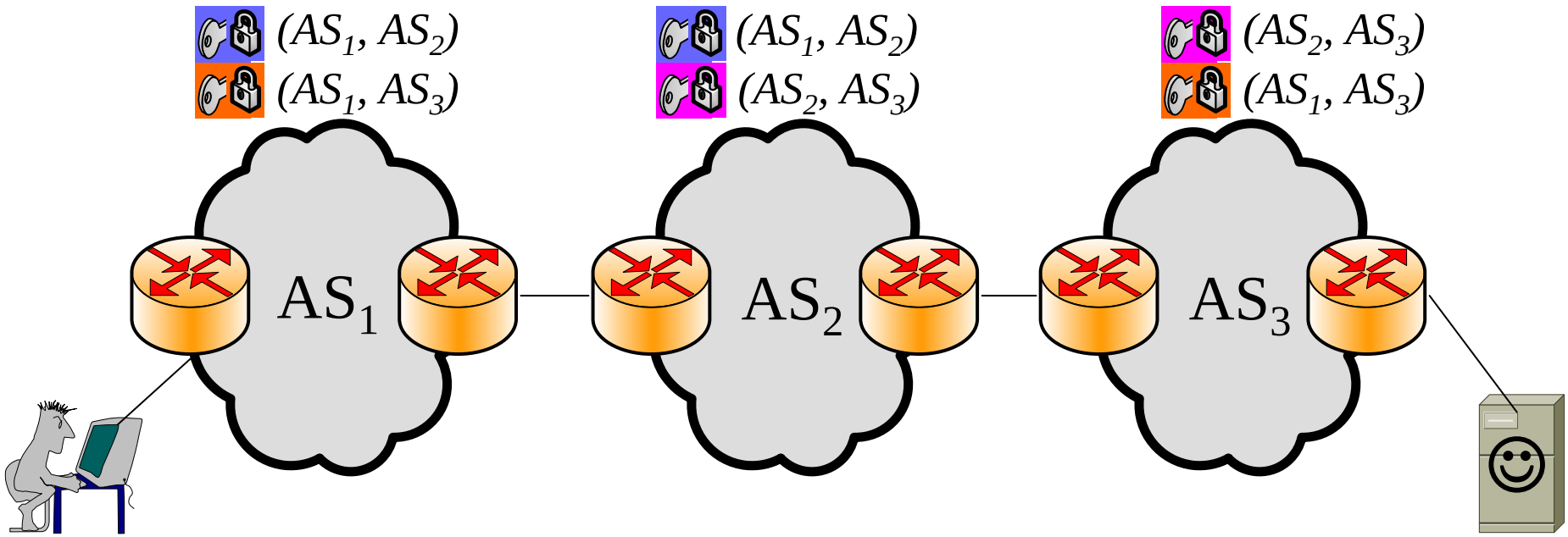
- Passport prevents AS-level spoofing
  - One AS cannot spoof other ASes' addresses
- An AS is responsible to prevent internal spoofing
  - Ingress filters
  - An irresponsible AS only harms its own hosts
- Scalable, incentive compatible

Efficient symmetric key cryptography



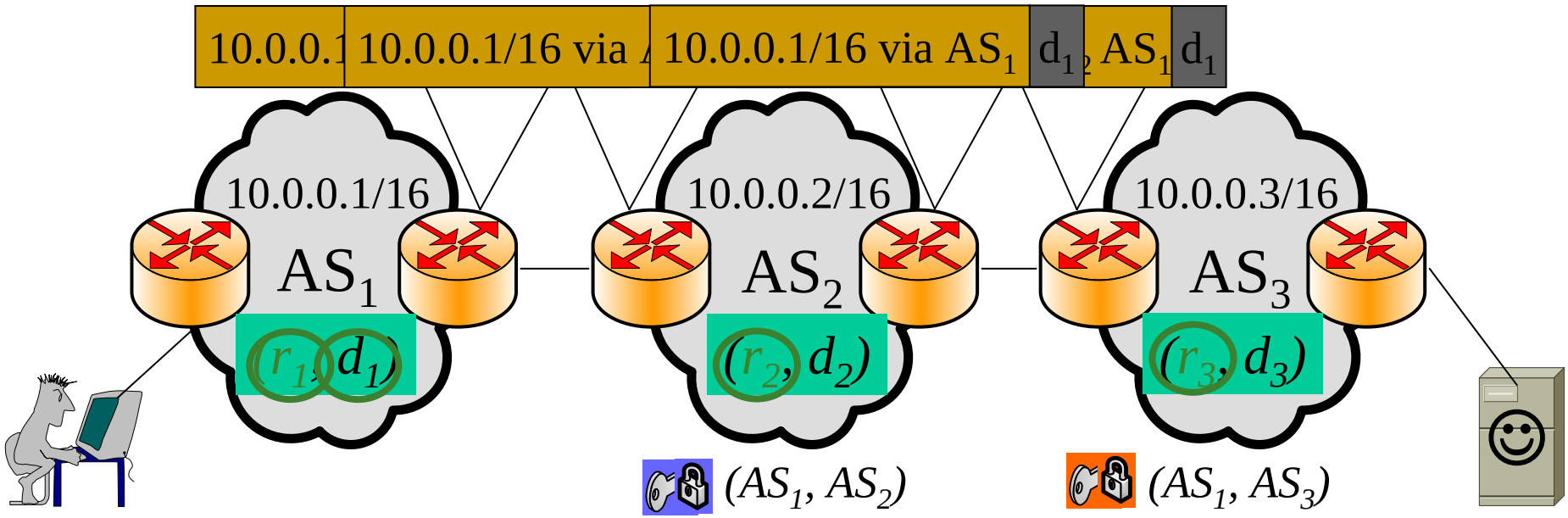
- Source border router stamps Message Authentication Codes (MACs) into a Passport header
  - Obtain AS paths from BGP
- Other border routers verify corresponding MACs
  - Demote or discard invalid Passports

How to obtain shared secret keys



- Problems
  - Bootstrap: chicken-and-egg
  - Efficiency: must obtain shared keys with ~30K ASes

A Diffie-Hellman key exchange via routing

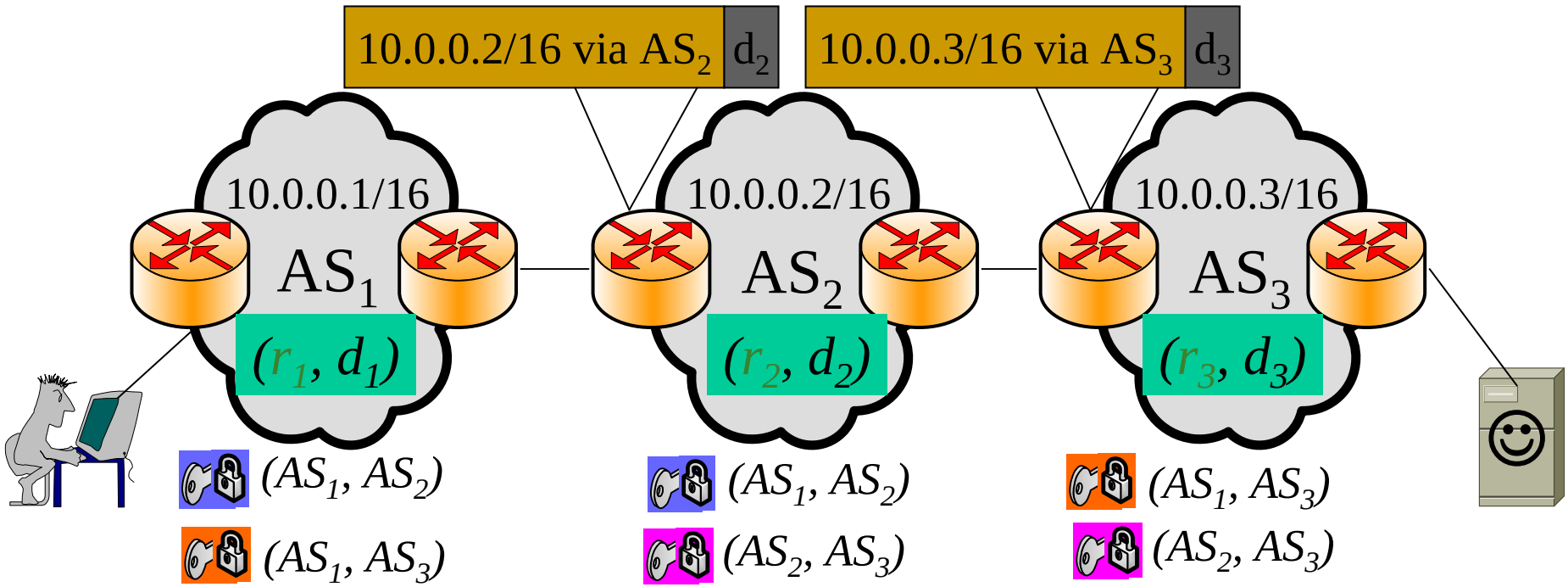


$d_i = g^{r_i} \bmod p$ g, p are system-wide parameters

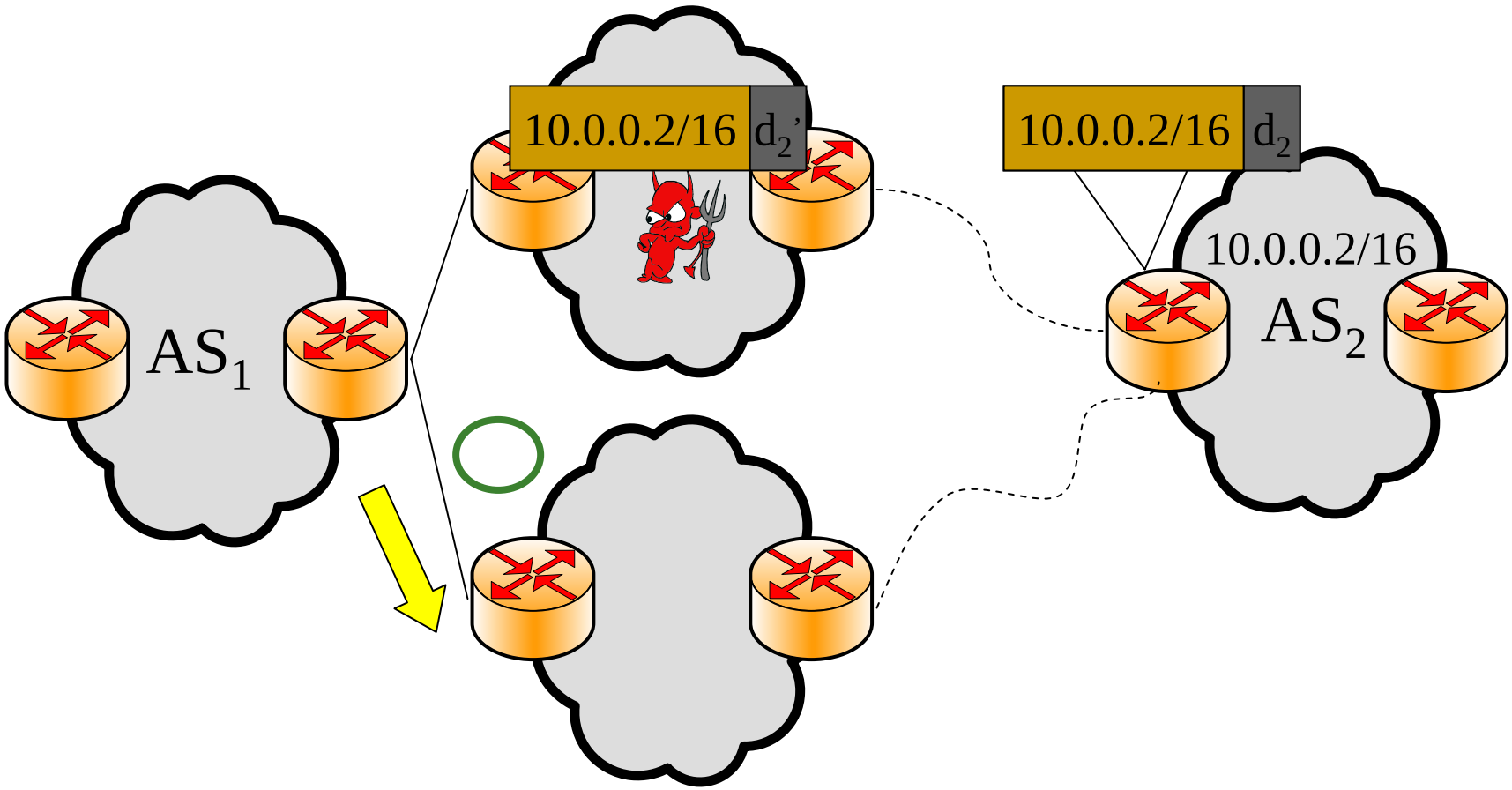
$(AS_1, AS_2) = (d_1)^{r_2} \bmod p = (d_2)^{r_1} \bmod p$

$(AS_1, AS_3) = (d_1)^{r_3} \bmod p = (d_3)^{r_1} \bmod p$

A Diffie-Hellman key exchange via routing



Secure key distribution via routing



- Accept *d* received from the next hop AS
- Secure routing → secure source authentication

Routing helps a lot

- Bootstrap and secure key exchange
- Efficient
  - Send one announcement, establish all pair keys
- DoS-resistant
  - High priority forwarding

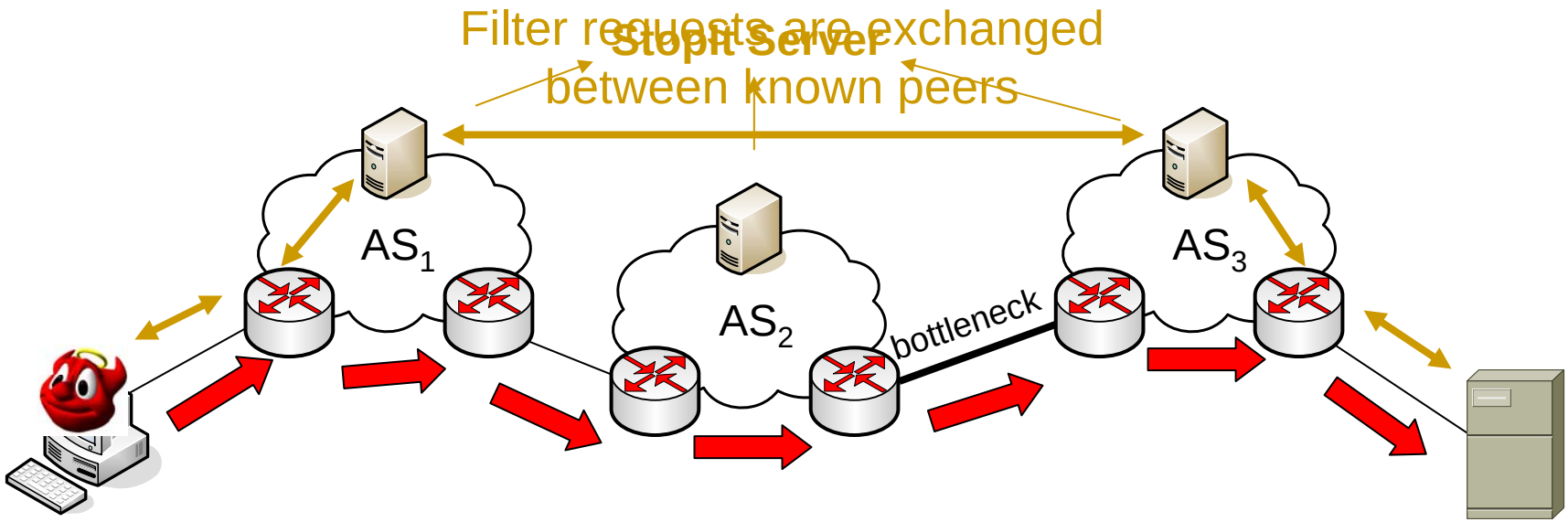
Other design issues

- Incremental deployable
  1. Transparent to hosts
  2. Inter-operate with legacy ASes
  3. Downstream legacy ASes can also benefit
     - BGP optional and transitive attributes
     - A shim layer
     - Encapsulation
- Secure under host, monitor, and router attackers
  - Seamless rekey
  - Resistant to sniff-and-replay: bound to a path
- Handle path changes
  - Demote at the intermediate ASes

Secure the Basic Design

<table>
<tr><th>Problems</th><th colspan="2">Solutions</th></tr>
<tr><td>Source address spoofing attacks</td><td colspan="2">Authenticate source addresses with Passport [NSDI’08]</td></tr>
<tr><td>Impersonation attacks</td><td>Authenticate filter requests with standard authentication techniques</td><td rowspan="2">Closed control channel</td></tr>
<tr><td>Filter exhaustion attacks</td><td>Confirm attacks before accepting filter requests; avoid filters against compliant sources; catch and punish misbehaving sources</td></tr>
<tr><td>Control channel DoS attacks</td><td colspan="2" rowspan="3">Source-based fair queuing</td></tr>
<tr><td>Filters fail to install</td></tr>
<tr><td>Incentives to deploy</td></tr>
</table>

Closed Control Channel

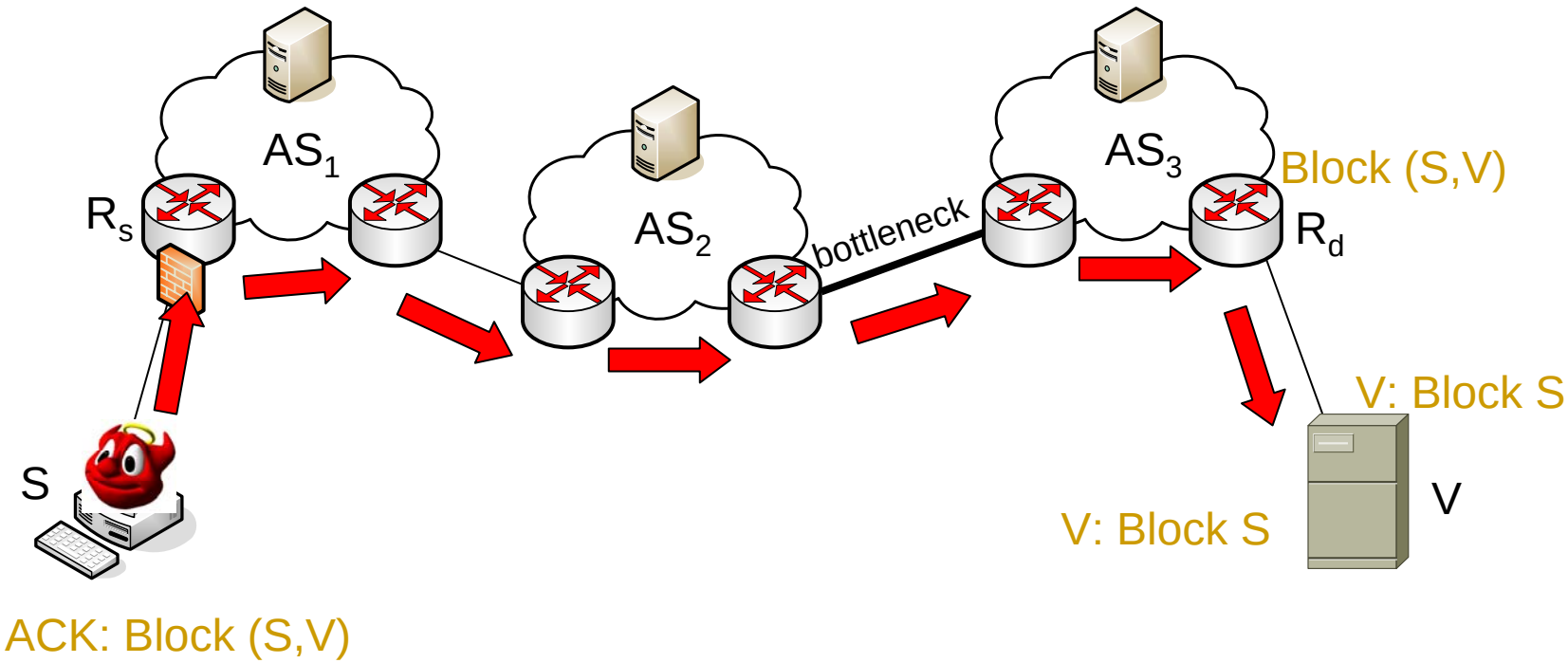


StopIt Server addresses are published in BGP

BGP Prefix Announcement

10.1.0.0/16		StopIt Server Address	

Steps to Block Attack Traffic



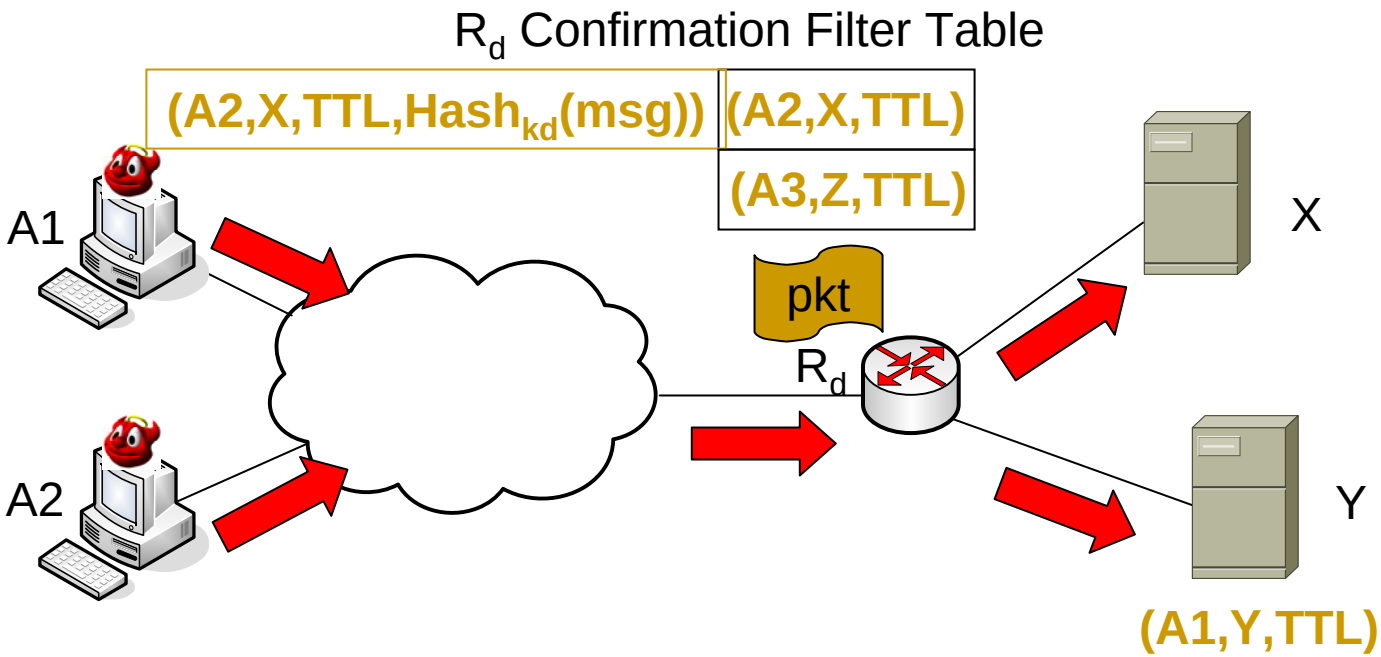
End-to-end requests before submitting filter requests
Attack confirmation on R_d to mitigate filter exhaustion attacks
Use source address and IP-ASN mapping to locate source AS
Request-ACK between S and R_s to mitigate filter exhaustion attacks

Confirm that Attack Traffic Exists

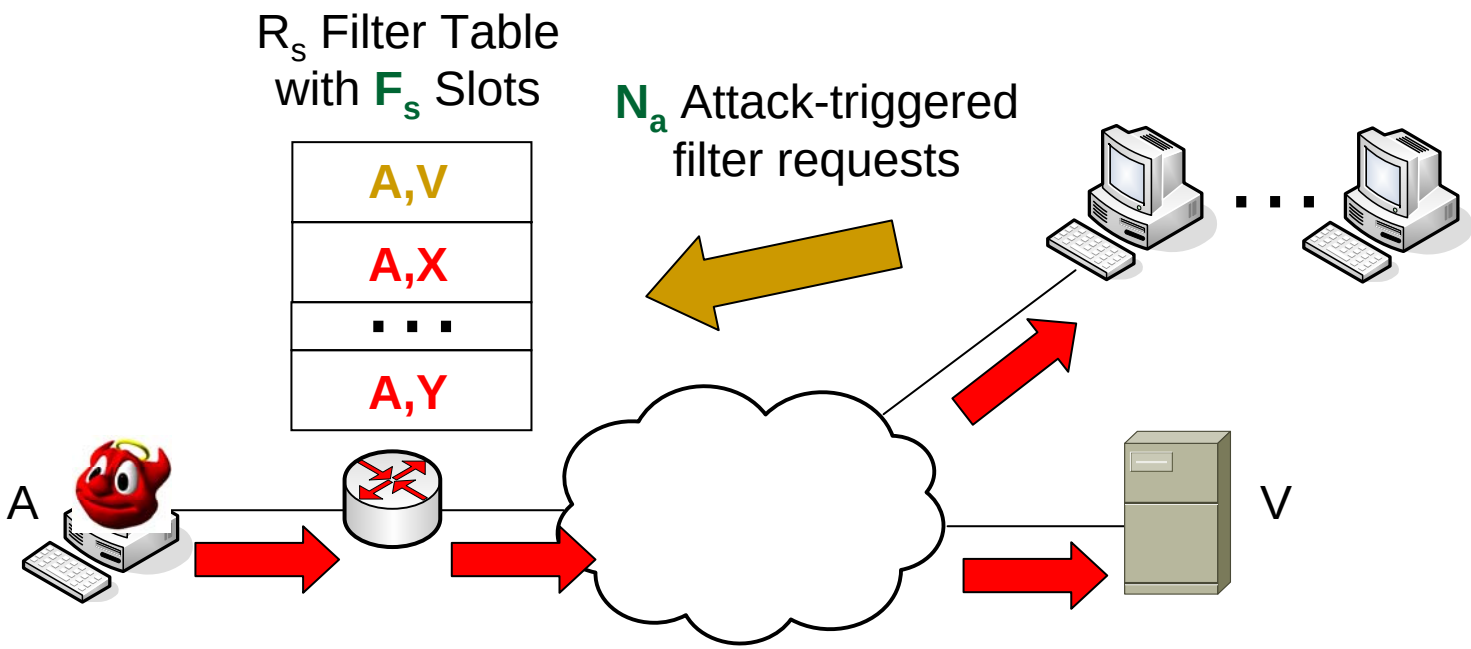
- Goal: prevent attackers installing filters against non-existent traffic
- Confirm attack traffic with flow cache
  - Access routers use flow cache to record recent src-dst pairs
  - Filter requests against traffic not in the flow cache are discarded

Confirm Source is Non-compliant

- Goal: prevent malicious destinations installing filters against compliant sources on source access routers
- Mitigate filter exhaustion: secure filter swapping



Source-side Filter Exhaustion Attack



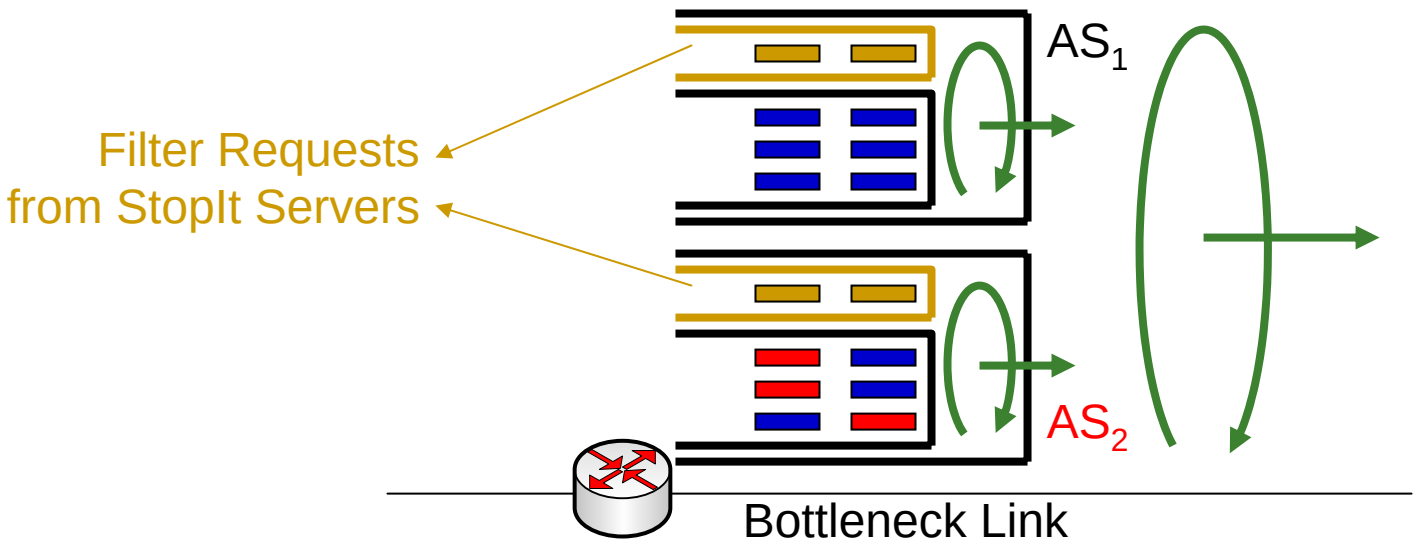
- Random filter replacement: $P_{caught}=(1-1/F_s)^{Na}$
  - E.g.: if Fs=1k and Na=1k, Pcaught=36.8%
- Aggregate misbehaving sources' filters
- Quota on filter requests to limit attacker capacity

Secure the Basic Design

<table>
<tr><th>Problems</th><th colspan="2">Solutions</th></tr>
<tr><td>Source address spoofing attacks</td><td colspan="2">Authenticate source addresses with Passport [NSDI’08]</td></tr>
<tr><td>Impersonation attacks</td><td>Authenticate filter requests with standard authentication techniques</td><td rowspan="2">Close the control channel</td></tr>
<tr><td>Filter exhaustion attacks</td><td>Confirm attacks before accepting filter requests; avoid filters against compliant sources; catch and punish misbehaving sources</td></tr>
<tr><td>Control channel DoS attacks</td><td colspan="2" rowspan="3">Source-based fair queuing</td></tr>
<tr><td>Filters fail to install</td></tr>
<tr><td>Incentives to deploy</td></tr>
</table>

Two-level Hierarchical Fair Queuing

- First-level fair queuing: source AS
  - Limit damage of attack traffic when filters fail to install
  - Incentivize deployment
- Second-level fair queuing: source address
  - Give inter-domain filter requests guaranteed bandwidth



Evaluate StopIt

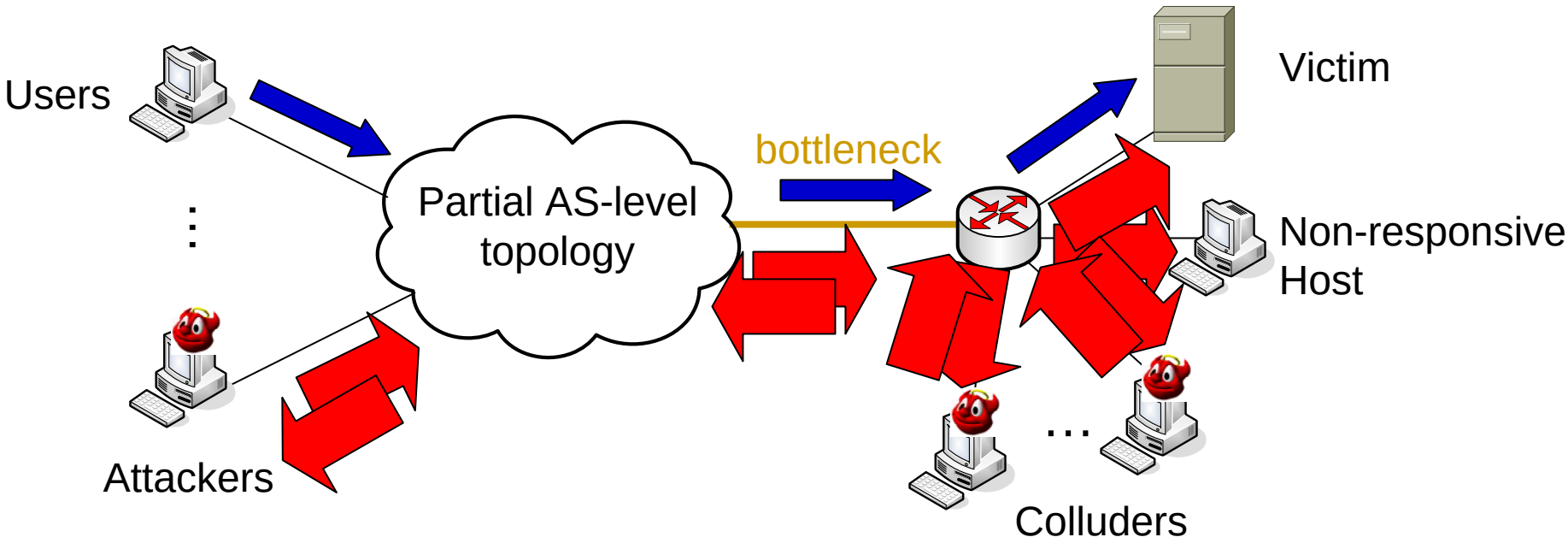
- Prototype implemented on Linux using Click
- Evaluated on Deterlab
  - Block various number of attackers with destination-side filter exhaustion
  - Source-side filter exhaustion attack
- Main Results
  - Block 10M attackers in 1658 seconds
  - With 10M filter slots and 10M daily quota on filter requests, on average an attacker can at most attack a victim 2.4 times per day

Compare Filters & Capabilities: Settings

- DoS Mitigation Systems
  - Filter-based: StopIt, AITF, Pushback
  - Capability-based: TVA, TVA+(Passport), Portcullis
- Topology
  - a branch of AS-level topology from RouteViews
- Scale-down factor: 1/20
  - E.g., bottleneck bandwidth: 1Gbps(simulated) = 50Mbps(real)
- Metrics of effectiveness
  - Ratio of successful file transfers
  - Average file transfer time

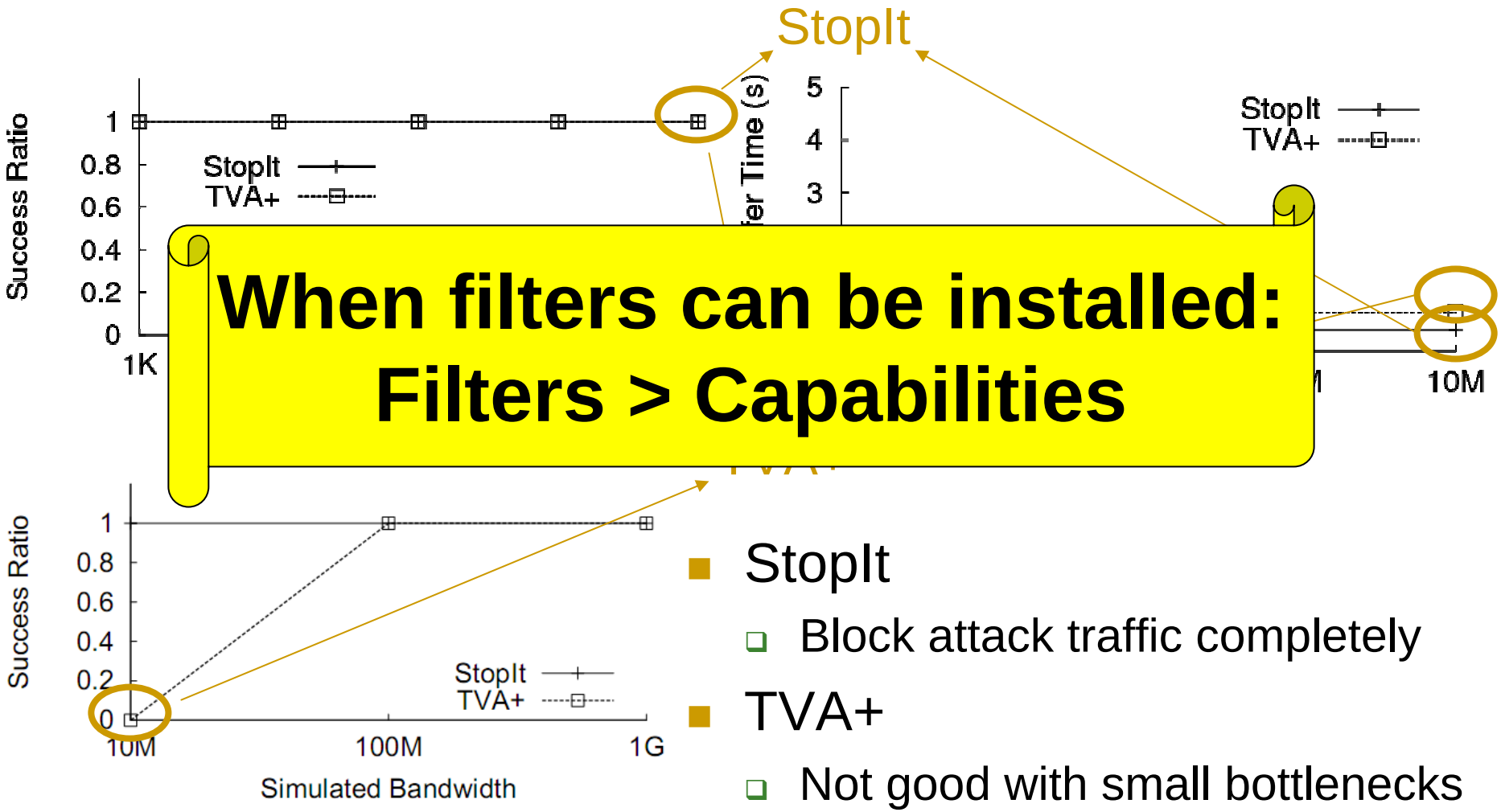
  Default file size: 20KB
- Default simulated bottleneck bandwidth: 1Gbps

Compare Filters & Capabilities: Attacks



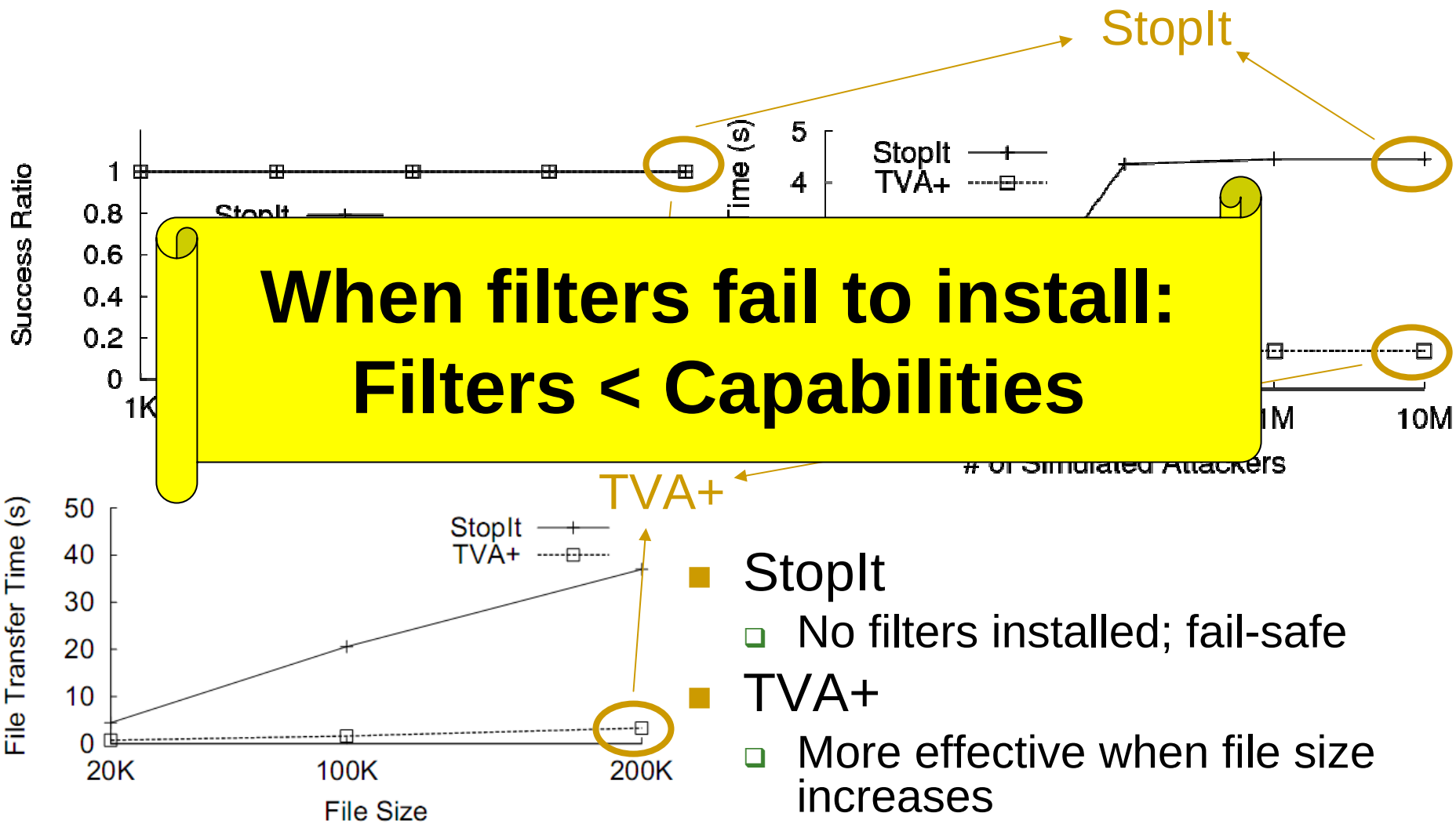
- Destination flooding attacks
- One-way link flooding attacks
- Two-way link flooding attacks

Destination Flooding Attacks



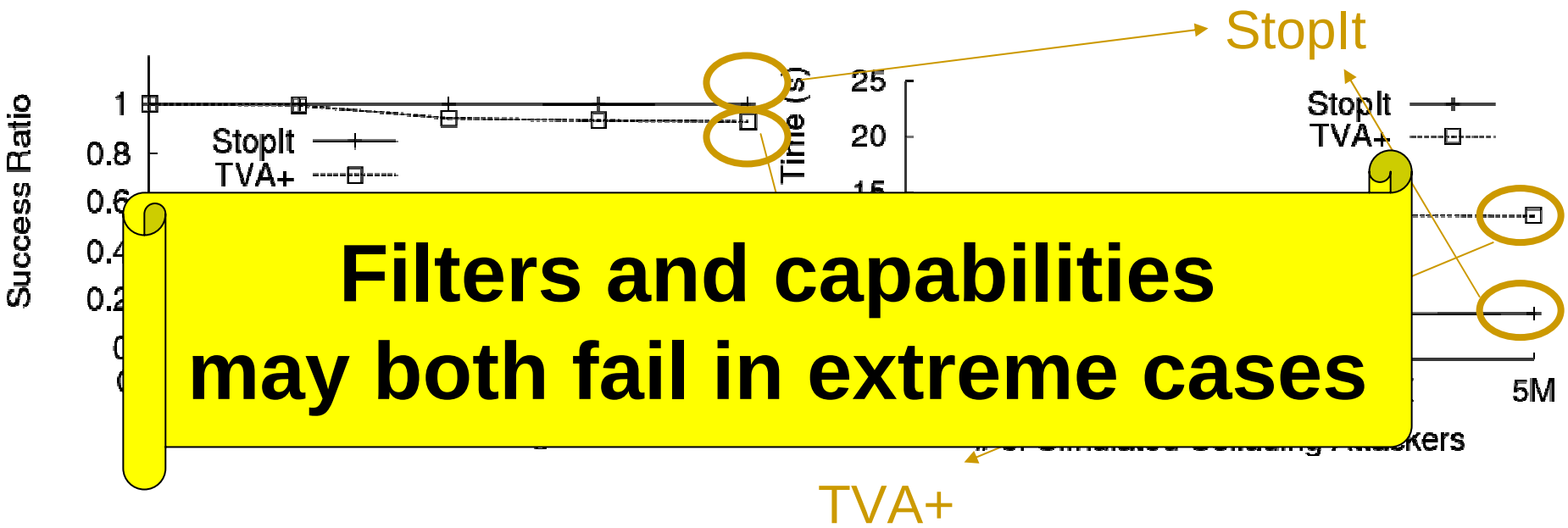
- StopIt
  - Block attack traffic completely
- TVA+
  - Not good with small bottlenecks

One-Way Link Flooding Attacks



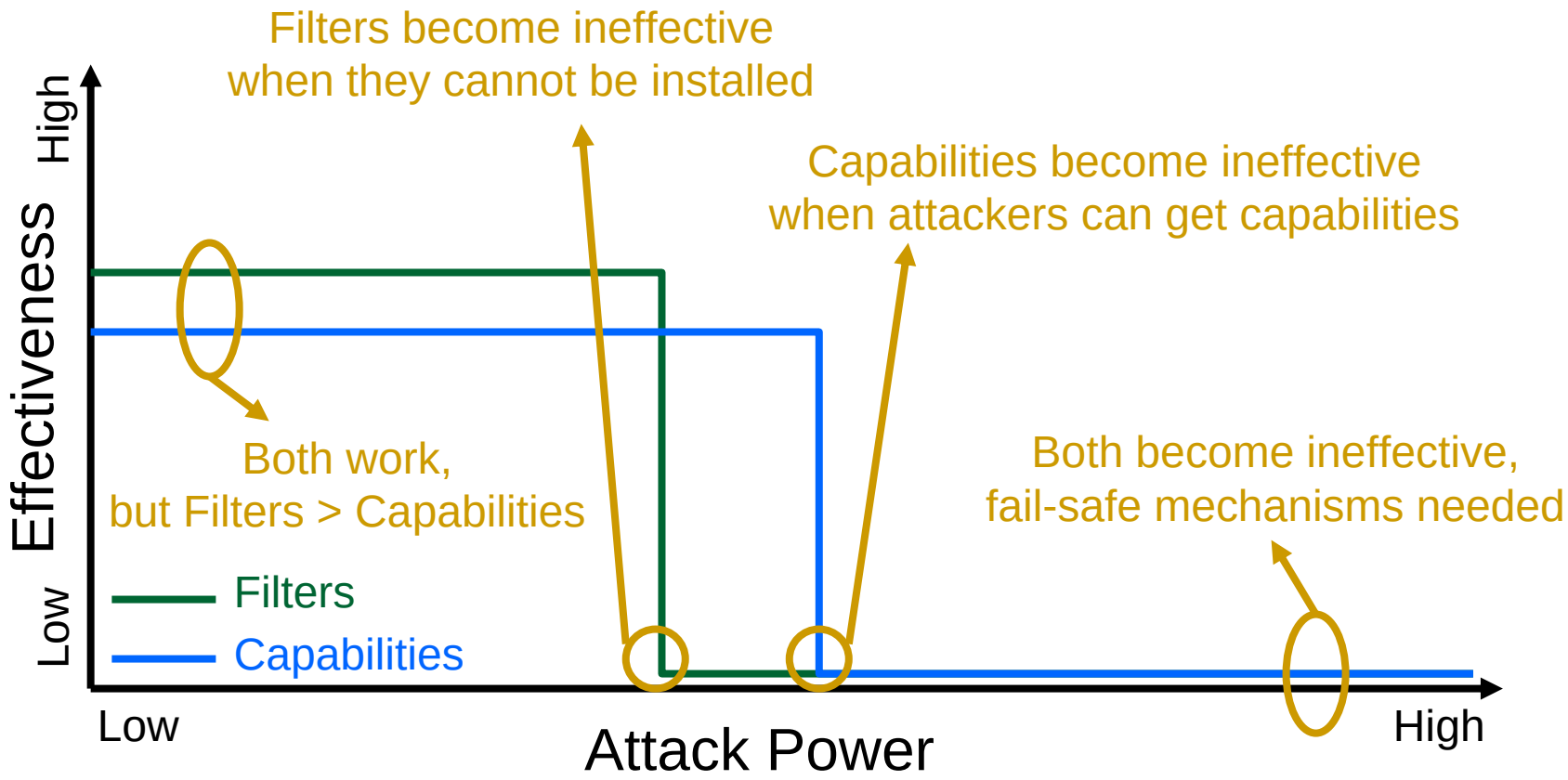
- StopIt
  - No filters installed; fail-safe
- TVA+
  - More effective when file size increases

Two-Way Link Flooding Attacks



- StopIt
  - No filters installed; degraded to per-source FQ
- TVA+
  - Attackers get capabilities; degraded to per-destination FQ
- Under the specific settings, per-src FQ > per-dst FQ

Compare Filters & Capabilities: Summary



Conclusion

- It's feasible to design an effective filter system
  - Resilient to various attacks
  - Fail-safe
- Filters v.s. Capabilities
  - Filters are more effective if they can be installed
  - Capabilities are more robust against attacks
  - Capability systems tend to be simpler
- Capabilities + Per-AS fairness: might be the most cost-effective solution