Resilient Networking
Module 5: Denial of Service

Thorsten Strufe – *This module prepared in cooperation with Günter Schäfer, Mathias Fischer, and the members of the Chair.*

Winter Term 2021 – KIT/TUD

KASTEL Security Research Labs

Denial of Service

- Classification
- DoS examples
  - Exploiting IP fragmentation and assembly
  - Abusing ICMP: Smurf attack
  - TCP SYN-Flood attack
  - DDoS
  - Botnets
  - DRDoS
- Countermeasures against DoS
  - Crypto Puzzles
  - Stateless Protocols
  - Avoid IP address spoofing / identifying malicious nodes
  - Filtering attack traffic
  - Industry solutions to DDoS mitigation

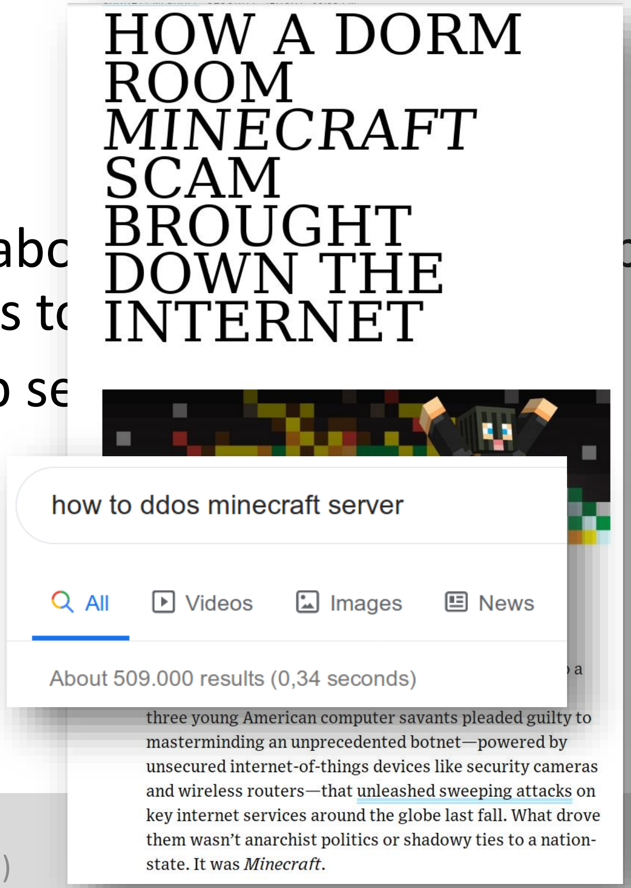
The Threat...

(source: Julie Sigwart - "Geeks")

Introduction

- What is Denial of Service?
  - Denial of Service (DoS) attacks aim at ***denying*** or ***degrading*** legitimate users' ***access to a service*** or network resource, or at bringing down the servers offering such services

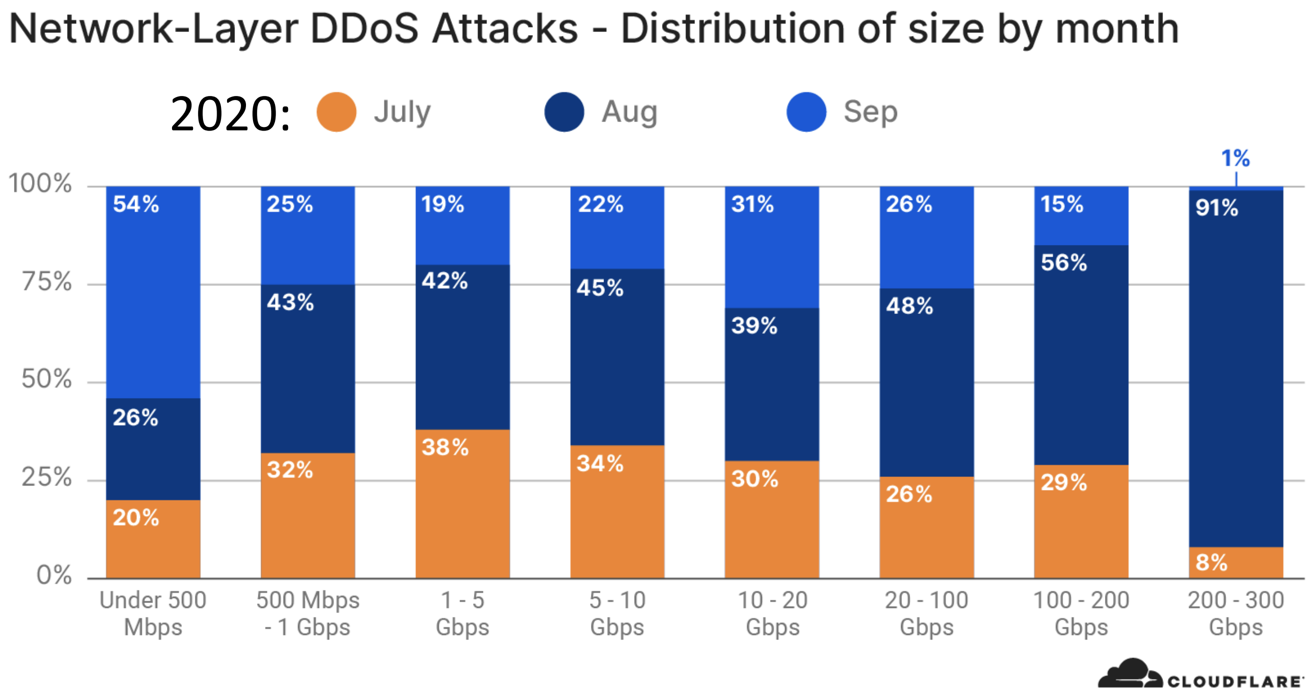
- Motivations for launching DoS attacks:
  - Hacking (just for fun, by "script kiddies", ...)
  - Gaining information leap (→ 1997 attack on bureau of labo … possibly launched as unemployment information has implications to …
  - Discrediting an organization operating a system (i.e. web se…
  - Revenge (personal, against a company, ...)
  - Political reasons ("information warfare")
  - Financial advantage (mirai and minecraft, 2016)
  - ...



How serious is the DoS problem? (1)

Karlsruhe Institute of Technology

- Qualitative answer:
  - Very, as our modern information society depends increasingly on availability of information and communications services
  - Even worse, as attacking tools are available for download



- Largest seen DoS attack so far: 2.3 Tbps (on Amazon AWS in 2020)

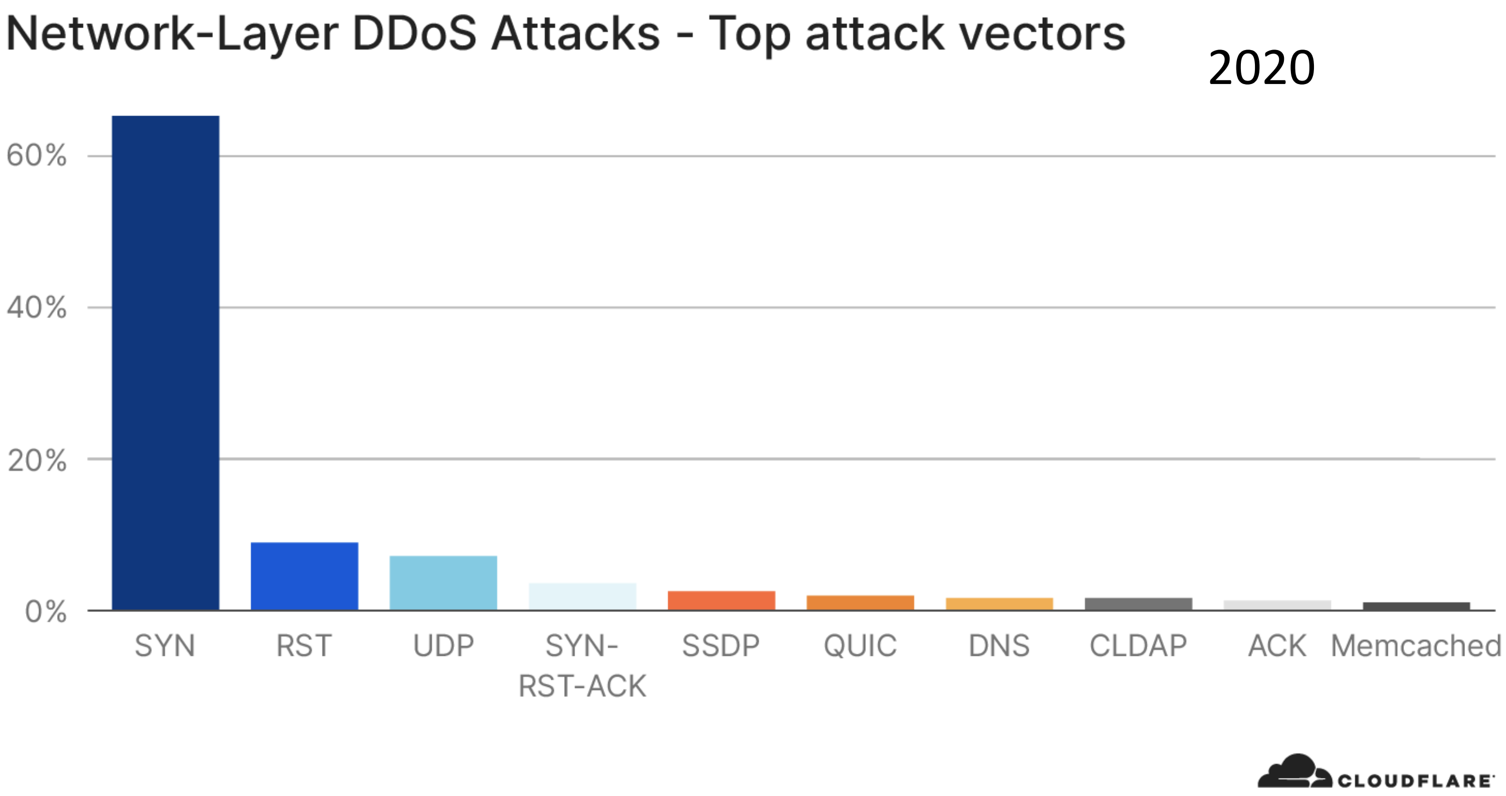
https://blog.cloudflare.com/network-layer-ddos-attack-trends-for-q3-2020/

How serious is the DoS problem? (2)

Karlsruhe Institute of Technology

- Various attack vectors used

DDoS blackmailing is a lucrative business model!



https://blog.cloudflare.com/network-layer-ddos-attack-trends-for-q3-2020/

Denial of Service Attack Classes

Classification depending on different aspects:

- *Attack effect*
  - Resource destruction
  - Resource depletion

- *Origin of malicious traffic*
  - Single source with single / multiple (forged) source addresses
  - Multiple sources (Distributed DoS)

- *Attack target*
  - Victim
  - Infrastructure

Attack Effect in Denial of Service

- *Affected resource*
  - Network connectivity (uplink, transit link)
  - Computation
  - Memory

- *Resource **destruction***:
  - Hacking into systems
  - Making use of implementation weaknesses like buffer overflows
  - Deviation from proper protocol execution
  - Your common TU Dresden Excavator

- *Resource **depletion*** by causing:
  - Storage of (useless) state information
  - High traffic load (requires high overall bandwidth from attacker)
  - Expensive computations ("expensive cryptography"!)
  - Resource reservations that are never used (e.g. bandwidth)

So how is it done?

Attacking Techniques

Karlsruhe Institute of Technology

- ***Reflector*** attacks: Generate traffic indirection
  - Request service in the name of the victim (e.g. spoofed IP – *which protocols*?)
  - Hides attack source, allows for external amplification

- ***Amplification*** attacks: Leverage asymmetry in protocols
  - Send lightweight requests (low cost) that generate heavyweight responses or heavy load on the service (crypto)
  - Increases damage

Resource Destruction

Resource Destruction – Examples (1)

- *Resource Destruction:*
- Physically/Logically destroy a resource that is vital for targeted service

- *Hacking:*
  - Exploiting weaknesses that are caused by careless operation of a system
  - Examples: default accounts and passwords not disabled, badly chosen passwords, social engineering (incl. malware attachments), etc.

- *Making use of implementation weaknesses*
  - Buffer Overflows, Format-String-Attacks, ...

- *Deviation from proper protocol execution:*
  - Example: exploit IP's fragmentation & reassembly

Resource Depletion

Background: Internet Control Message Protocol

- Internet Control Message Protocol (ICMP) has been specified for communication of error conditions in the Internet
- ICMP PDUs are transported as IP packet payload and identified by value “1” in the protocol field of the IP header

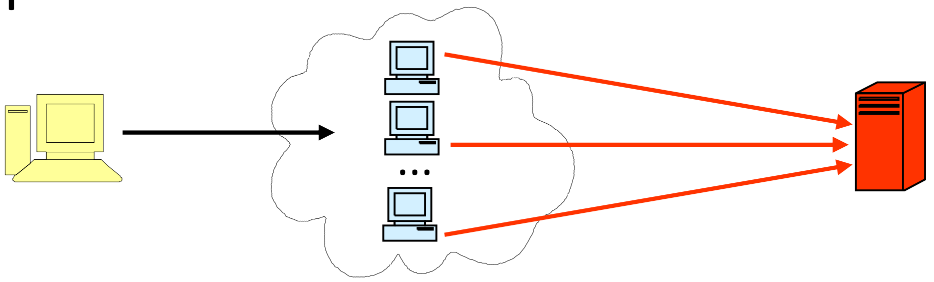
- Two main reasons make ICMP particular interesting for attackers:
  - It may be addressed to broadcast addresses
  - Routers respond to it

The mother of DoS: Smurf – ICMP Bandwidth Depletion

Karlsruhe Institute of Technology

- Two reasons make ICMP particular interesting for attackers:
  - It may be addressed to broadcast addresses
  - Routers respond to it

- The ***Smurf attack*** - ICMP echo request to broadcast:
  - Routers (sometimes) allow ICMP echo requests to broadcast addresses…
  - An attacker sends an ICMP echo request to a *broadcast address* with the *source address* forged to refer to the victim
  - All devices in the addressed network respond to the packet
  - The victim is flooded with replies to the echo request
  - With this technique, the network being abused as an (unaware) attack amplifier is also called a *reflector network*:



More recent examples…

Global Distributed Denial-Of-Service (DDoS) Protection Market 2019 –
…nie Networks, ARBOR NETWORKS, Imperva
…uard

The global "**Distributed Denial-Of-Service (DDoS) Protection**" market report
…ted Denial-Of-Service (DDoS)
…also assesses the Distributed Denial-
… of topography, technology, and
consumers. The report also covers the volume of the market during the projected
…ted Denial-Of-Service (DDoS)
…sentation of the Distributed Denial-
…the global and regional level. **The key**
…R NETWORKS, Imperva Incapsula

Jonker, Mattijs, et al. "Millions of targets under attack: a macroscopic characterization of the DoS ecosystem." *Proceedings of the 2017 Internet Measurement Conference*. ACM, 2017.

Rossow, Christian. "Amplification Hell: Revisiting Network Protocols for DDoS Abuse." *NDSS*. 2014.

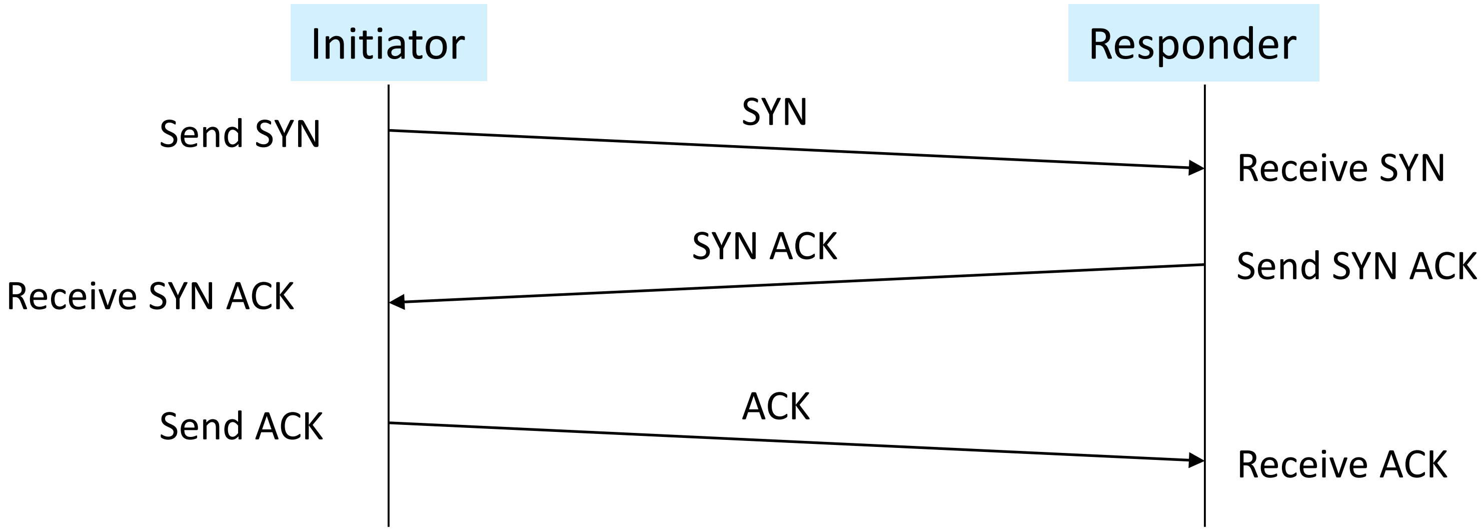
"Identifying the scan and attack infrastructures behind amplification DDoS attacks." *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016.

Schuchard, Max, et al. "Losing control of the internet: using the data plane to attack the control plane." *Proceedings of the 17th ACM conference on Computer and communications security*. ACM, 2010.

Smith, Jared M., and Max Schuchard. "Routing around congestion: Defeating DDoS attacks and adverse network conditions via reactive BGP routing." *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018.

Depleting Memory: TCP’s Three-Way-Handshake

- The *Transmission Control Protocol (TCP):*
  - provides a connection-oriented, reliable transport service
  - uses IP for transport of its PDUs
- TCP connection establishment is realized with handshake:



- After handshake, data can be exchanged in both directions
- Both peers may initiate termination of the connection (two-way-handshake)

More recent CPU Exhaustion Attacks…

Nexus Intelligence Insights: CVE-2018-1109-Braces Regular expression Denial of Service (ReDoS) attack

by Elisa Velarde on June 28, 2019

Summ
once a
Expre

What r
and filt
slighte

ReDoS
the po
consur
occur c

Name

Cisco Security

Cisco Security Advisories and Alerts

Advanced Search

	ADVISORY/ALERT	IMPACT	CVE	LAST UPDATED	VERSION
	Search Advisory/Alert Name	All	Search CVE	Most Recent	
▶	Cisco ASA and FTD Software Cryptographic TLS and SSL Driver Denial of Service Vulnerability	High	CVE-2019-1873	2019 Jul 11	1.1
▶	Cisco IOS XR Software BGP MPLS-Based EVPN Denial of Service Vulnerability	High	CVE-2019-1849	2019 Jul 10	1.1
▶	Multiple Issues in Cisco Small Business 250/350/350X/550X Series Switches Firmware and Cisco FindIT Network Probe	Informational		2019 Jul 09	1.1
▶	Cisco Unified Communications Manager Session Initiation Protocol Denial of Service Vulnerability	High	CVE-2019-1887	2019 Jul 08	1.1
▶	Cisco IP Phone 7800 and 8800 Series Session Initiation Protocol Denial of Service Vulnerability	Medium	CVE-2019-1922	2019 Jul 08	1.1
▶	Cisco Web Security Appliance HTTPS Certificate Denial of Service Vulnerability	High	CVE-2019-1886	2019 Jul 03	1.0
▶	Cisco Small Business Series Switches Memory Corruption Vulnerability	High	CVE-2019-1892	2019 Jul 03	1.0
▶	Cisco Small Business Series Switches HTTP Denial of Service Vulnerability	High	CVE-2019-1891	2019 Jul 03	1.0

So what can we do?

Defending Against Resource Depletion DoS

Karlsruhe Institute of Technology

- Defenses against resource depletion:
- Generally:
  - ***Rate Control*** (ensures availability of other functions on same system)
  - Distribution of load
  - Authentication & Accounting
- Expensive computations: careful protocol design, verifying the initiator's "willingness" to spend resources himself (e.g. "client puzzles")
- Memory exhaustion: stateless protocol operation

Attack Sources and Spoofed Addresses

- Concerning origin of malicious traffic:
- Defenses against single source attacks:
  - Disabling of address ranges (helps if addresses are valid)

- Defenses against forged source addresses:
  - Ingress Filtering at ISPs (if the world was an ideal one...)
  - “Verify” source of traffic (e.g. with exchange of “cookies”)
  - Tracing back the true source of packets with spoofed addresses

- Widely distributed DoS:
  - Offloading to Site Delivery Services/CDN

Memory Exhaustion: Stateless Protocols

Karlsruhe Institute of Technology

- Basic idea:
  - Avoid storing information at server, before DoS attack can be ruled out
  - So, as long as no assurance regarding the client has been reached all state is "stored" in the network (transferred back and forth)

Stateful Operation		Stateless Operation
1. $C \rightarrow S$: Msg_1		1. $C \rightarrow S$: Msg_1
2. $S \rightarrow C$: Msg_2	S stores $State_{S1}$	2. $S \rightarrow C$: Msg_2, $State_{S1}$
3. $C \rightarrow S$: Msg_3		3. $C \rightarrow S$: Msg_3, $State_{S1}$
4. $S \rightarrow C$: Msg_4	S stores $State_{S2}$	4. $S \rightarrow C$: Msg_4, $State_{S2}$
...		...

- Drawback: requires higher bandwidth and more message processing

CPU Exhaustion: Client Puzzles/Proof of Work

Observations and assumptions:

- DoS (also: spam) works because there's no postage paid (cost) when message is sent
- Amplification attacks require few resources at client and cause large load at victim

- *Proof of Work*: level the playing fields by making the clients prove that they invested resources
- One-way functions are cheap to evaluate, but "impossible" to invert

- Good (as any) approach to inversion is guessing, partial guessing may be possible:
  - Chances to guess x such that

    `P[H(x) = yyyyyyy0] = .5`

    what about `P[H(x) = yyyy000]?` ;-)

Simple Client Puzzles:

- Let server draw a pre-image at random
- Provide client with image and request it to provide the pre-image

Conclusion

Karlsruhe Institute of Technology

- Increasing dependence of modern information society on availability of communication services

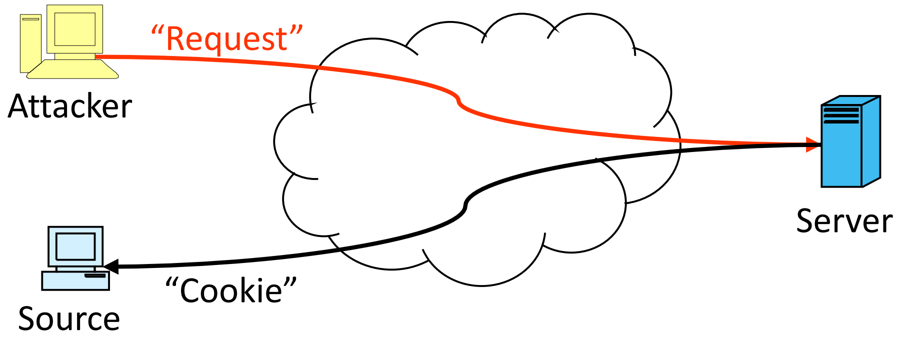
- While some DoS attacking techniques can be encountered with “standard” methods, some can not:
  - Hacking, exploiting implementation weaknesses, etc. may be encountered with firewalls, testing, monitoring etc.
  - Malicious protocol deviation & resource depletion is harder to defend against

- Designing DoS-resistant protocols emerges as a crucial task for network engineering:
  - Network protocol functions and architecture will have to be (re-)designed with the general risk of DoS in mind
  - Base techniques: stateless protocol design, cryptographic measures like authentication, cookies, client puzzles, etc.

Verifying the Source of a Request

- Problem: Spoofed addresses allow adversaries to hide

- Basic solution:
  - Before working on a new request, verify if the "initiator" can ***receive messages,*** sent to the claimed source of the request



  - Only a legitimate client or an attacker which can receive the "cookie", can send the cookie back to the server
  - Of course, an attacker must not be able to guess the content of a cookie

- Discussion:
  - Advantage: allows to counter simple spoofing attacks
  - Drawback: requires one additional message roundtrip

But…

- Verifying the source of a request with a cookie exchange can ***avoid spending significant computation*** or ***memory*** resources on a bogus request

- What if the attacker is only interested in ***exhausting*** the access or packet processing ***bandwidth*** of a victim?
  - Obviously, sending cookies to all incoming packets even aggravates the situation!
  - Such an attack situation, however, is quite easy to detect: there are simply too many packets coming in

- Problems in such a case:
  - Which packets come from ***genuine sources*** and which are ***bogus ones***?
  - Even worse: source addresses given in the packets may be spoofed
  - Where do the spoofed packets come from?

Possible Solutions to DDoS-Attacks (1)

- Solutions to ***Reflector Attacks***: secure available services
  - Prevent amplification: Balance effort of request and reply
    e.g.: Prohibit ICMP-Echo-Request to broadcast addresses
    - => Reflectors don't amplify attack magnitude
      (however: does this work with all protocols? DNS?)
  - Access-controlled services: provide service to authorized parties only
    e.g.: Prohibit recursive DNS queries for external users

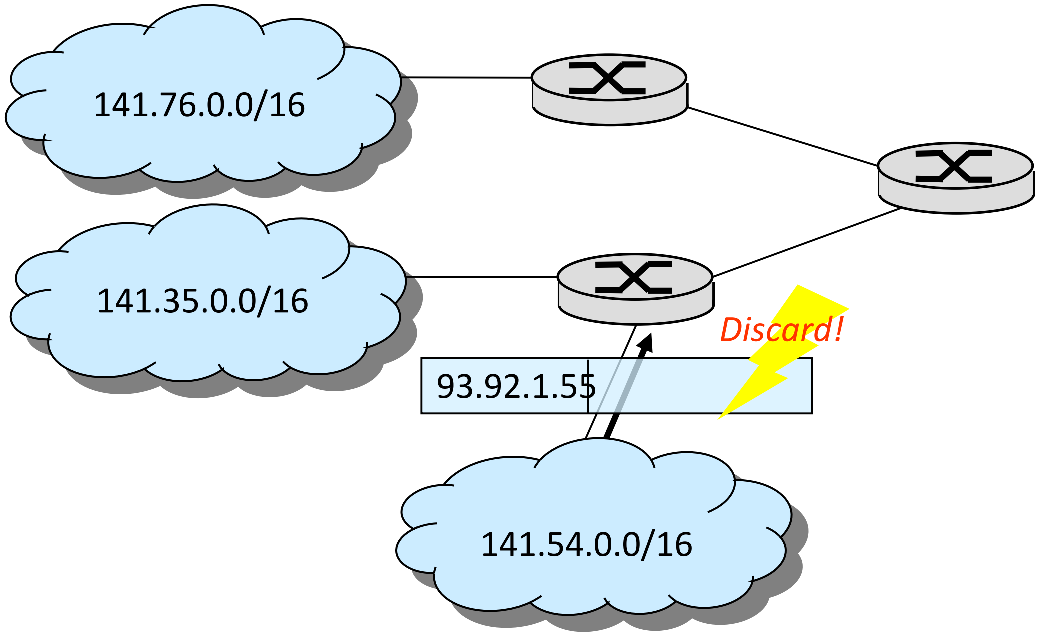
Possible Solutions to DDoS-Attacks (2)

- Possible Solutions to ***Direct Attacks***:
  - Avoid IP-Address spoofing
  - Live with spoofed addresses and restrain effect of attacks
    - Locate source of attack-packets
    - Filter traffic from attacking nodes
    - Inform admin/root of attacking networks/node

- But: IP is connectionless! Necessary to find means to trace back the traffic to the original source / attacking node!
- Identify: zombie, spoofed address, ingress router, routers on path…

Inhibiting Spoofed Addresses: Ingress Filtering (RFC 2267)

- Routers block arriving packets with illegitimate source addresses.



- IETF BCP 38 (May 2000)

Ingress Filtering (2)

Karlsruhe Institute of Technology

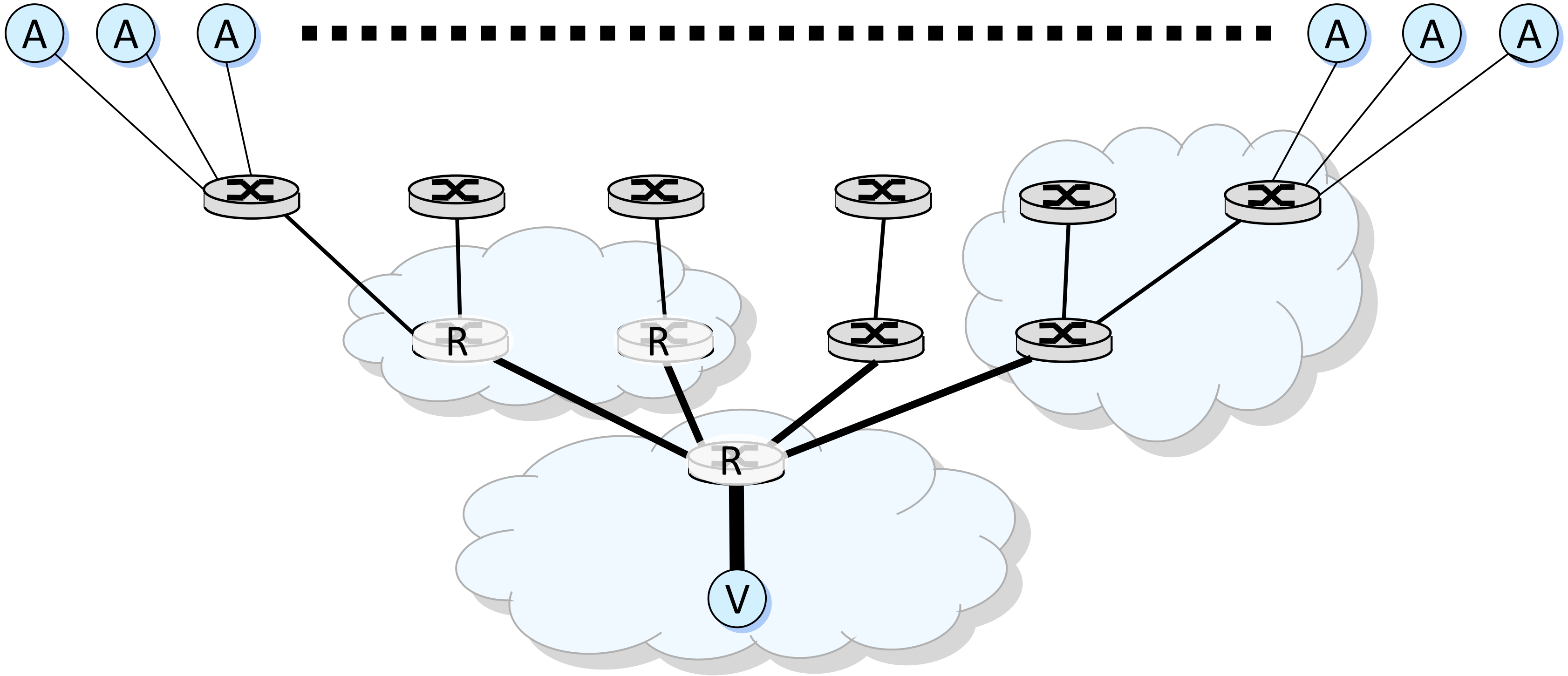
- Difficult in the backbone *(how to check if route is valid?)*
- Easily possible at access links → ISPs

- Problems occur:
  - Issues with Mobile-IP (theoretic) and load testing (local)
  - Large management overhead at router-level
  - Processing overhead at access routers
    - (e.g., big ISP running a large AS with numerous IP-Ranges and DHCP)
  - Universal deployment needed (cf. the situation today…)

- *ISPs don't really have an incentive in blocking any traffic*

KASTEL

Identify Malicious Nodes: DDoS Attack-Tree



- Rooted Tree with
  - Victim *(V)* (root of the tree)
  - Routers *(R)*
  - Attackers *(A_i)*

Questions with forged IP addresses:

- Where are malicious nodes?
- Which router (ISP) is on attack path?

Identifying Malicious Nodes: Assumptions

Karlsruhe Institute of Technology

- Packets are subject to ***reordering and loss***
- ***Resources*** at routers are ***limited***
- ***Routers*** are usually ***not compromised***
- Attackers may ***generate any packet***
- Attackers are ***aware of tracing***
- ***Multitude*** of ***attacking packets*** (usually many)
- ***Routes*** between A and V are ***stable*** (in the order of seconds)
- Multiple attackers can act in ***collusion***

Identify Malicious Nodes: Proposed Solutions

Simple classification of solutions:

- Network Logging
    - Log information on processed packets and path
- Attack Path Traceback
    - Trace attack path through network
- Other / Related
    - Attack Mitigation/Avoidance

Identifying Malicious Nodes: Proposed Solutions

- Network Logging
  - Local network logging
  - Aggregated network logging
  - Source Path Identification („Hash-based IP-Traceback“)
- Attack Path Traceback
  - Input Debugging
  - Controlled Flooding
  - ICMP Traceback
  - Probabilistic Packet Marking („IP-Traceback“)
- Other / Related
  - Hop-Count Filtering
  - Aggregate Based Congestion Control (ACC)
  - Secure Overlay Services

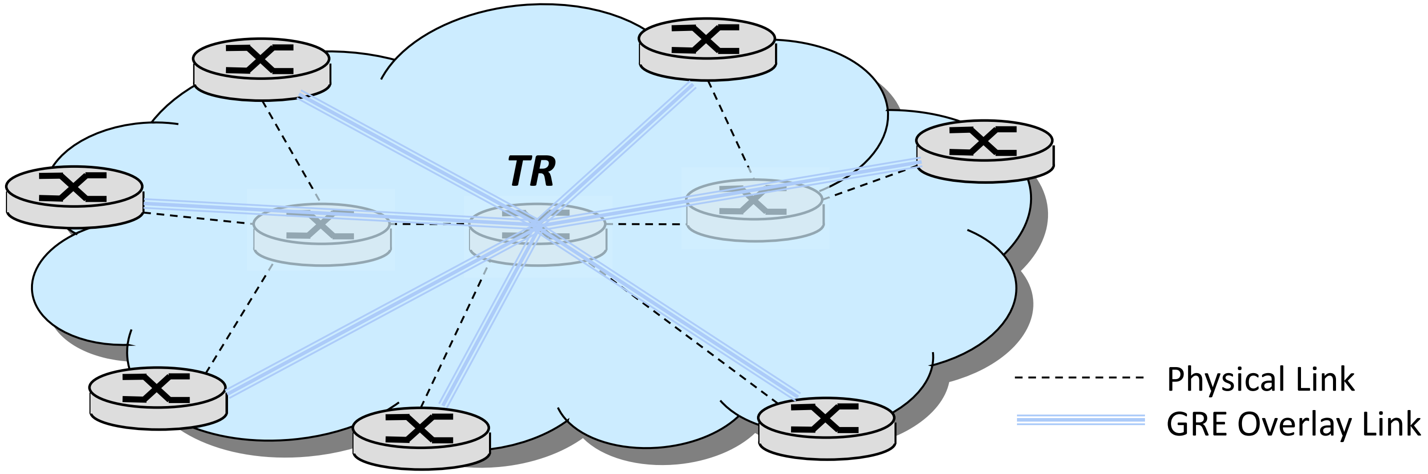
Logging Approaches

- Log information on processed packets and path

- Network logging
  - Local network logging:
    - All routers log all traffic
    - Too much overhead!
    - **Does not scale**
  - Aggregated network logging
  - Source Path Identification („Hash-based IP-Traceback“)

Aggregated Network Logging

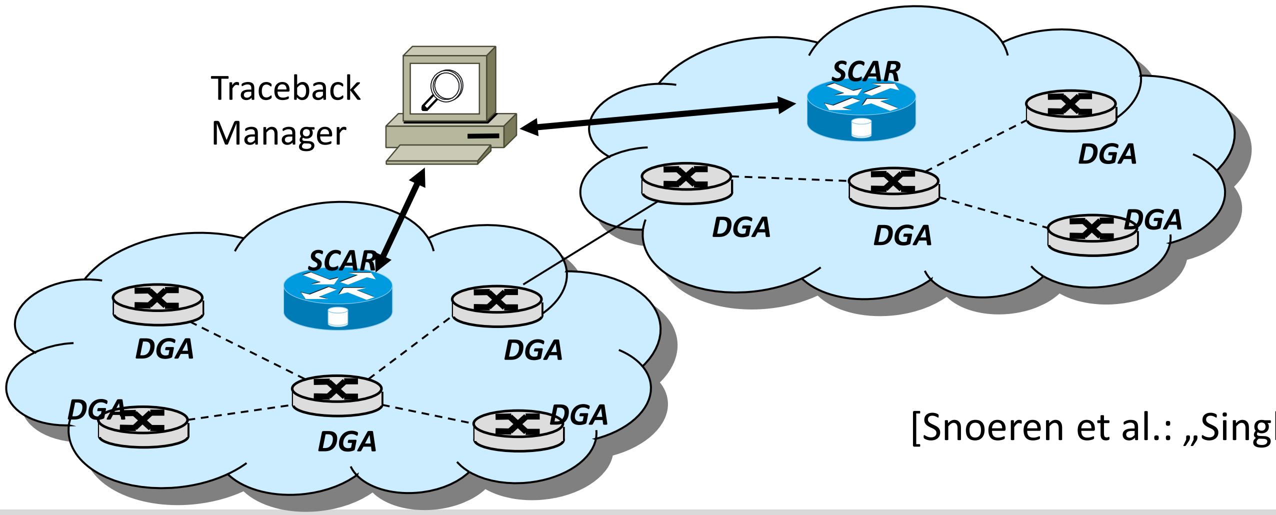
- Centralized approach:
  - Introduction of „Tracking Router“ (TR)
  - Forwarding all traffic through TR (*via GRE*)
  - TR logs all traversing traffic
  - Creates one single point of failure! Does not scale! *(Altough: SDN…)*



[Stone: „Centertrack: An IP Overlay Network for Tracking DoS Floods“]

Source Path Identification

- Source Path Identification Engine *(SPIE, aka Hash-based IP Traceback)*
- Storage of compressed data in specialized devices
  - DGA generate digests of data *(Data Generation Agent)*
  - SCAR for storage and retrieval *(SPIE Collection & Reduction Agents)*
  - STM for central management *(SPIE Traceback Manager)*



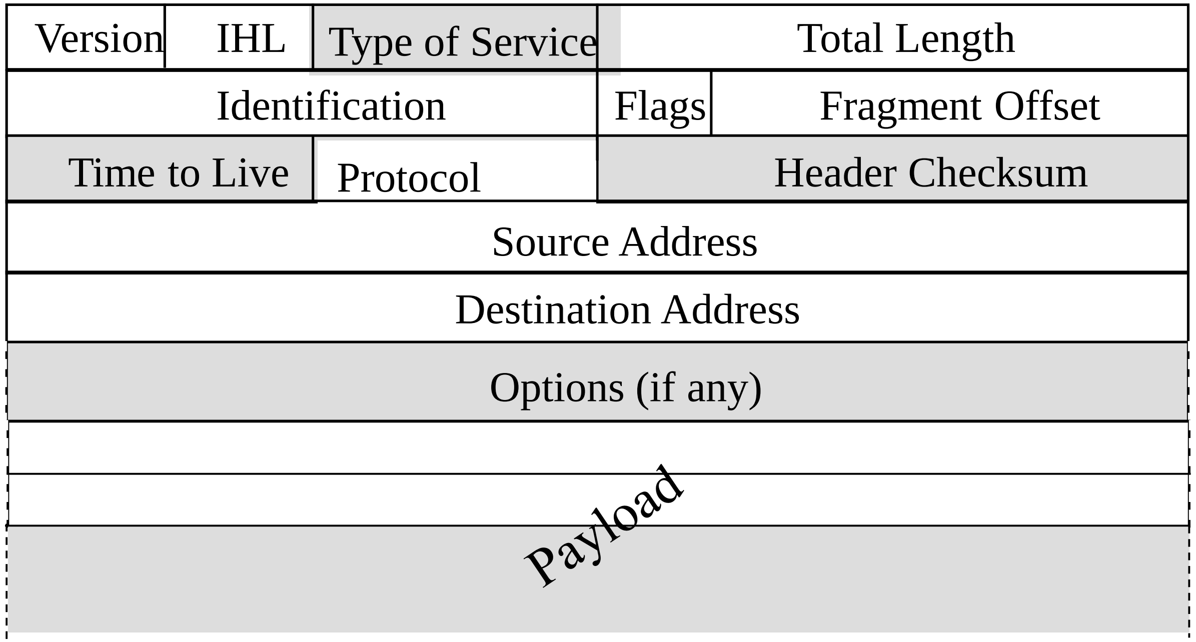
[Snoeren et al.: „Single-Packet IP-Traceback"]

Source Path Identification (2)

- *„Store all information on traversed packets?“*
- No! *What do we need to store?*
- Store digests of:
  - Constant fields in IP Header (16 bytes)
  - First 8 bytes of payload
- Still a lot, compress:

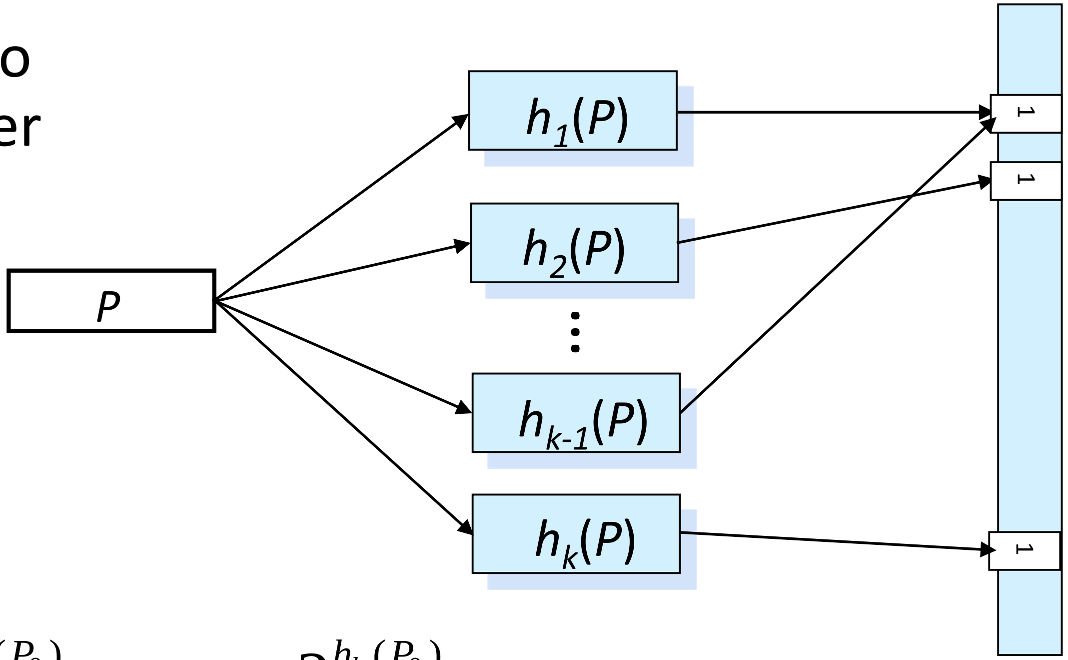
  Hashed in

  Bloom Filters



Source Path Identification: *Bloom Filters* (1)

- 24 bytes of each packet hashed with *k* hash functions h_i
- Hash values stored in filter:
  - To store $h_i(P)$, write a 1 into position $2^{h_i(P)}$ in bloom filter



$$BF(P_0) = 2^{h_1(P_0)} \text{ or } 2^{h_2(P_0)} \text{ or } \ldots \text{ or } 2^{h_k(P_0)}$$

$$BF(P_n) = BF(P_{n-1}) \text{ or } 2^{h_1(P_n)} \text{ or } 2^{h_2(P_n)} \text{ or } \ldots \text{ or } 2^{h_k(P_n)}$$

Traceback Approaches

- Trace attack path backwards through network

- Attack Path Traceback
  - Input Debugging
  - Controlled Flooding
  - ICMP Traceback
  - Probabilistic Packet Marking („IP-Traceback“)

Input Debugging

- During attack:
  - Trace attack-path „by hand“
  - Contact administrator / ISP
  - Admin matches ingress port for a given packet pattern of egress port
  - Repeat until source is found…

- Disadvantages:
  - Cumbersome (what if admin X is not available?)
  - Slow
  - Expensive (manual intervention)
  - Not scalable

…Yet the most applied method until today…

Controlled Flooding

- During Single Source DoS-Attacks, traversed backbone links on the attack path are (heavily) loaded

- Traceback attack path by testing links:
  - Measure incoming attack traffic
  - From victim to approximate source:
    - Create load on suspect links in the backbone
    - Measure difference in incoming attack traffic: if less attack packets arrive, the link is on the attack path…

- Need possibility to create load on links to test with access on end-hosts around the backbone (chargen-service on multiple foreign end-hosts)

- ☹ DoS of the backbone in itself

- Testing high speed backbone links using end-hosts difficult (how many dsl-links do you need to saturate one CISCO-12000-Link (10Gbps)?

[Burch & Cheswick: „Tracing Anonymous Packets to Their Approximate Source“]

Probabilistic Packet Marking (aka „IP Traceback“, PPM)

Approach by marking packets:

- Mark forwarded packets with a very low probability
- In-band signaling to avoid additional bandwidth needs (mark packets directly)

- Different marking methods possible
- Different signaling (encoding) methods possible

[Savage et al.: „Network Support for IP Traceback“]

Related Techniques for Mitigation / Avoidance

- Hop-Count Filtering
- Aggregate Based Congestion Control (ACC)
- Secure Overlay Services

Aggregate Based Congestion Control

Karlsruhe Institute of Technology

- Is it possible, to restrain attack traffic in the backbone?
  - Traffic is very diverse in the backbone, in general
  - However, attack traffic forms an aggregate of similar traffic

  (Identified by analyzing the dropped traffic:

  select the destination addresses with more than twice the mean number of drops and cluster these destination addresses to 24bit prefixes)

- ACC/pushback is a reactive approach:
  - If router/link is congested, can an aggregate be identified?
  - If there is an aggregate, limit the rate of aggregate traffic
  - If the aggregate persists, perform „pushback“: inform upstream routers to limit rate of the aggregate

[Mahajan, Bellovin & Floyd: „Controlling High Bandwidth Aggregates in the Network “]

Remote-Triggered Black Hole Filtering (1)

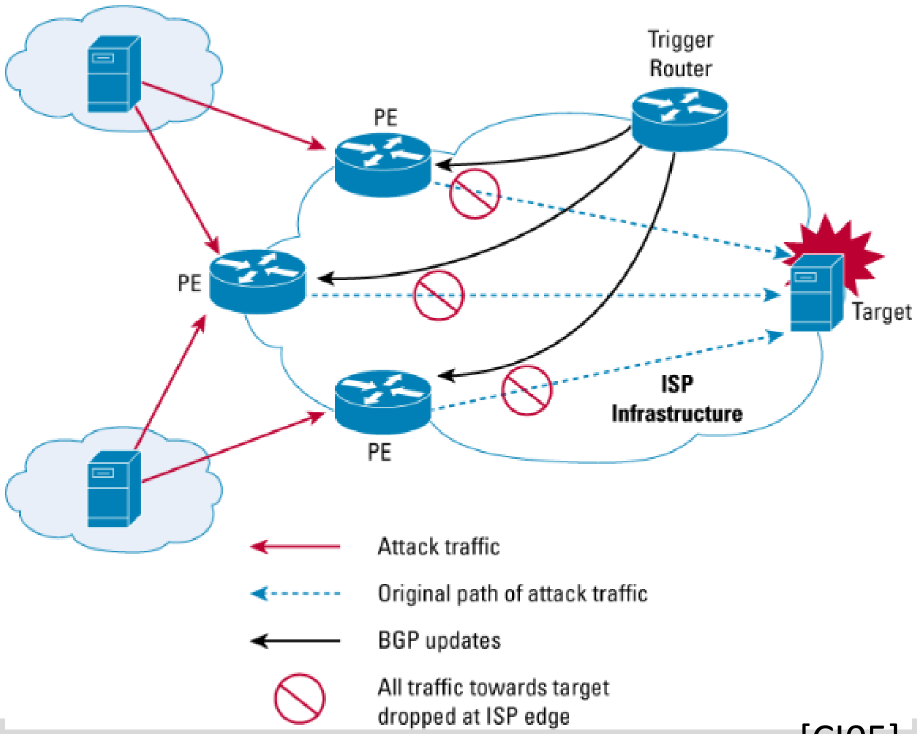
[CI05]

Karlsruhe Institute of Technology

Destination-Based Remotely Triggered Black Hole Filtering (D/RTBH)

- Goal: block all incoming traffic towards a particular address (space)
  - Before traffic enters the target network / at BGP router level
  - Update BGP table at routers to forward respective traffic to interface */dev/null*

- Leveraging BGP communities (RFC 3882)
  - To easily enable mechanism on only a subset of BGP routers
  - To control BGP-speaking routers in the attacked network to
    - either discard traffic or
    - forward it for inspection



[CI05]

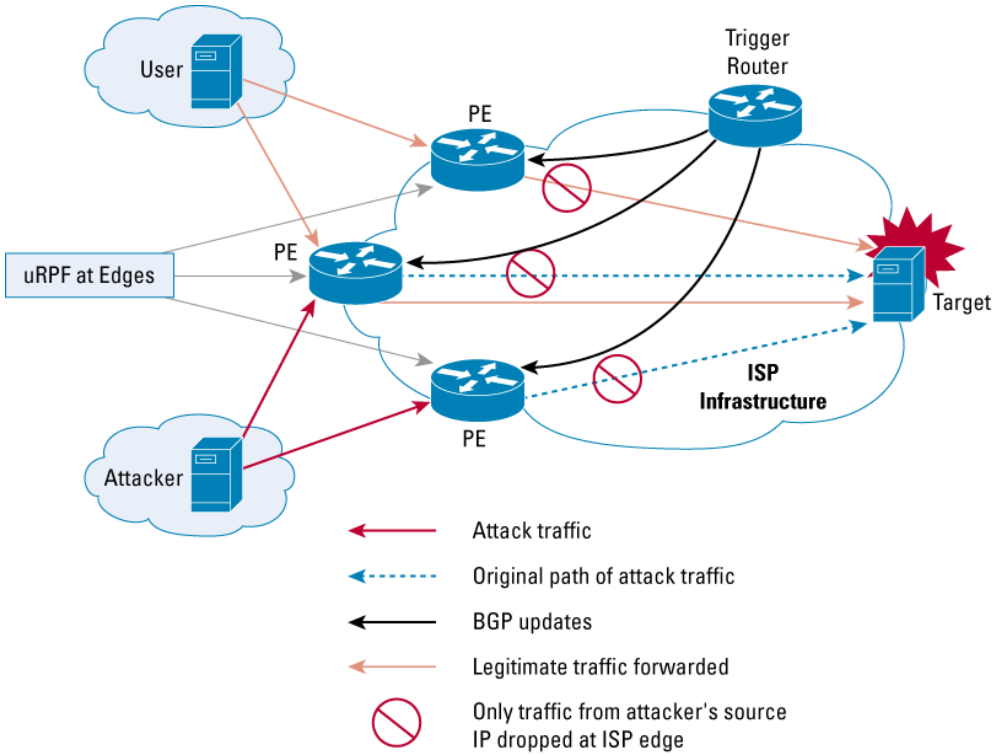
Source: https://www.cisco.com/c/dam/en_us/about/security/intelligence/blackhole.pdf

Remote-Triggered Black Hole Filtering (2) - S/RTBH

Karlsruhe Institute of Technology

Source-Based Remotely Triggered Black Hole Filtering (S/RTBH)

- Goal: Block all incoming traffic from a particular address (space)
  - Before traffic enters the target network, at BGP router level
  - Configure BGP-speaking routers to discard respective traffic that is not coming from the "expected" interface
  - Trigger router speaks iBGP (interior BGP) with border routers
  - Routers use Unicast Reverse Path Forwarding (uRPF)



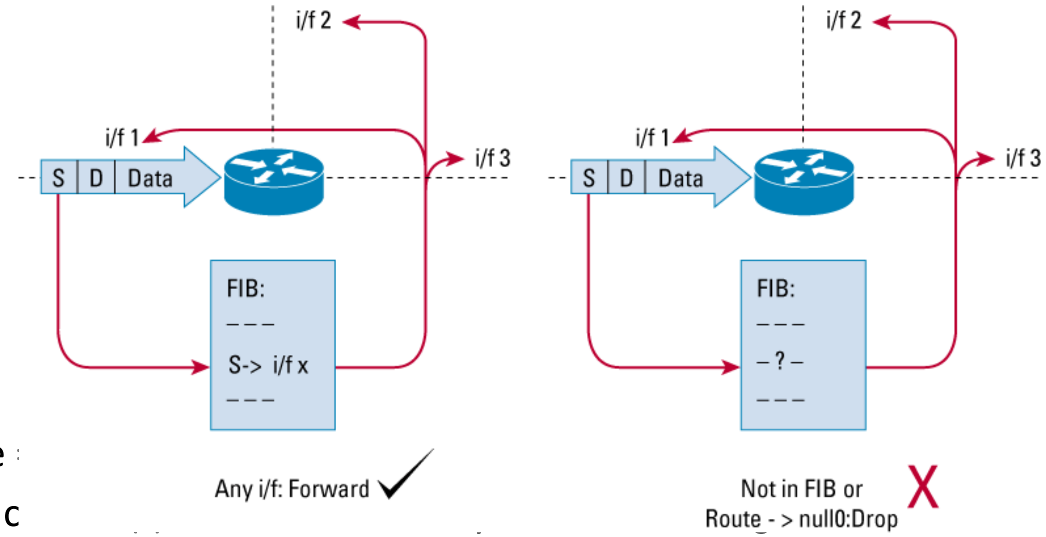
[CI05]

Remote-Triggered Black Hole Filtering (3) - S/RTBH

[CI05]

Karlsruhe Institute of Technology

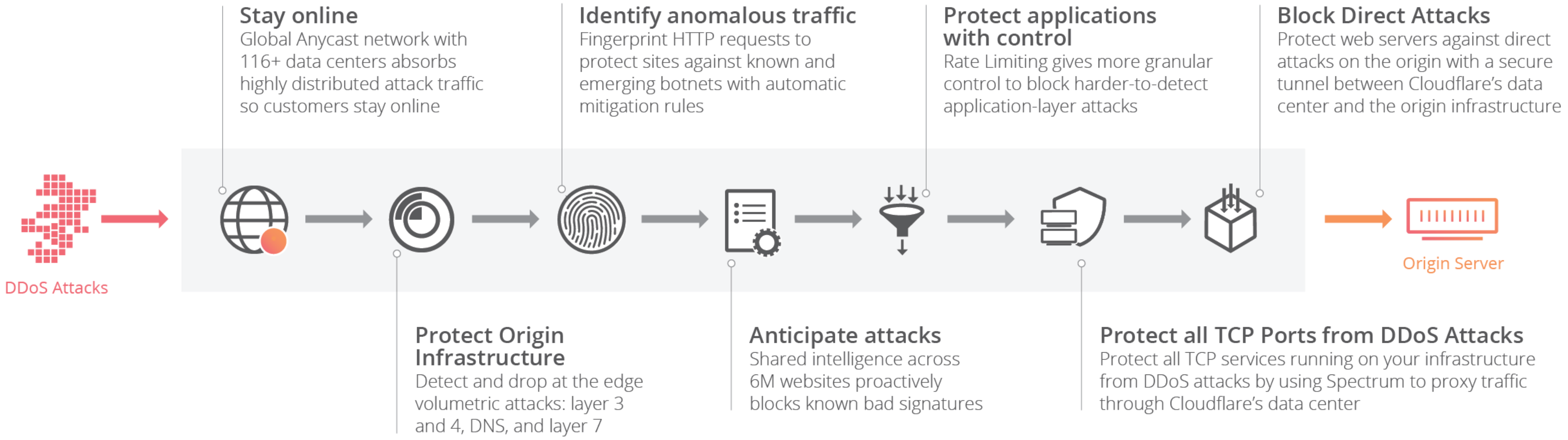
- Leveraging Unicast Reverse Path Forwarding (uRPF) (RFC 5635)
  - Routers perform a route lookup of the source address upon packet reception
  - Loose Mode:
    - Requires: egress interface for route lookup exists in Forwarding Information Base (FIB) at all [or, != */dev/null*]
    - iBGP updates to explicitly invalidate routes to suspicious source addresses by setting their next hop to /dev/null (or null0)



- Strict Mode:
  - Requires: ingress interface
  - (+) Might filter spoofed pac

DDoS Mitigation in the Wild

- Business model: being a DDoS (/security) shield.
- Companies like Cloudflare or Imperva Incapsula
  - Content Delivery Networks
  - Operation of IDSs/IPSs and Firewalls



Source: https://www.cloudflare.com/

Some Upcoming Challenges

Karlsruhe Institute of Technology

- The introduction of Internet protocols in classical and mobile telecommunication networks also introduces the Internet's DoS vulnerabilities to these networks
- Programmable end-devices (e.g., smartphones) may constitute a large base of possible slave nodes for DDoS attacks on mobile networks
- Software defined radio implementation may allow new attacking techniques:
  - Hacked smart phones answer to arbitrary paging requests
  - Unfair / malicious MAC protocol behavior
  - ...
- The ongoing integration of communications and automation may enable completely new DoS threats

Conclusion

Karlsruhe Institute of Technology

- Increasing dependence of modern information society on availability of communication services

- While some DoS attacking techniques can be encountered with “standard” methods, some can not:
  - Hacking, exploiting implementation weaknesses, etc. may be encountered with firewalls, testing, monitoring etc.
  - Malicious protocol deviation & resource depletion is harder to defend against

- Designing DoS-resistant protocols emerges as a crucial task for network engineering:
  - Network protocol functions and architecture will have to be (re-)designed with the general risk of DoS in mind
  - Base techniques: stateless protocol design, cryptographic measures like authentication, cookies, client puzzles, etc.

References (1)

[CSI00] Computer Security Institute and Federal Bureau of Investigation. *2000 CSI/FBI Computer Crime and Security Survey.* Computer Security Institute Publication, March 2000.

[Akamai16] Akamai. (2016). akamai’s [state of the internet] Q1 2016 report, 77. https://doi.org/10.1017/CBO9781107415324.004

[Dar00] T. Darmohray, R. Oliver. *Hot Spares For DoS Attacks.* ;login:, 25(7), July 2000.

[JuBr99] A. Juels und J. Brainard. *Client Puzzles: A Cryptographic Countermeasure Against Connection Depletion Attacks.* In Proceedings of the 1999 Network and Distributed System Security Symposium (NDSS’99), Internet Society, March 1999.

[Mea00] C. Meadows. *A Cost-Based Framework for the Analysis of Denial of Service in Networks.* 2000.

[MVS01] D. Moore, G. M. Voelker, S. Savage. *Inferring Internet Denial-of-Service Activity.* University of California, San Diago, USA, 2001.

[NN01] S. Northcutt, J. Novak. *Network Intrusion Detection - An Analyst’s Handbook.* second edition, New Riders, 2001.

[TL00] P. Nikander, T. Aura, J. Leiwo. *Towards Network Denial of Service Resistant Protocols.* In Proceedings of the 15th International Information Security Conference (IFIP/SEC 2000) Beijing, China, 2000.

[BA03] A. Belenky, N. Ansari:"On IP Traceback", in IEEE Communications Magazine, July 2003

[BC00] Burch & Cheswick: „Tracing Anonymous Packets to Their Approximate Source“, Proceedings of the 14th USENIX conference on System administration, 2000

[Bel01] Bellovin: „ICMP Traceback Messages“, Internet-Draft draft-ietf-itrace-01.txt, 2001

References (2)

[JWS03] Jing & Wang & Shin: „Hop-Count Filtering: An Effective Defense Against Spoofed DDoS Traffic“, Proceedings of the 10th ACM conference on Computer and communications security, 2003

[KMR02] Keromyits & Misra & Rubenstein: „SOS: Secure Overlay Services“, Proceedings of ACM SIGCOMM, 2002

[MBF01] Mahajan & Bellovin & Floyd: „Controlling High Bandwidth Aggregates in the Network“, Technical report, 2001

[RSG98] Reed, Syverson & Goldschlag: „Anonymous Connections and Onion Routing“, IEEE Journal on Selected Areas in Communications, 1998

[Sav01] Savage et al.: „Network Support for IP Traceback“, IEEE/ACM Transactions on Networking (TON), 2001

[Sto00] Stone: „Centertrack: An IP Overlay Network for Tracking DoS Floods“, Proceedings of 9th USENIX Security Symposium, 2000.

[Sno02] Snoeren et al.: „Single-Packet IP-Traceback“, IEEE/ACM Transactions on Networking (TON), 2002

[Ros14] Rossow, Christian. "Amplification Hell: Revisiting Network Protocols for DDoS Abuse." NDSS. 2014.

[JiWa+] Cheng Jing, Haining Wang, Kang G. Shin: „Hop-Count Filtering: An Effective Defense Against Spoofed DDoS Traffic“, CCS, 2003

c Cisco “Remotely triggered black hole filtering- destination based and source based” , Whitepaper, https://www.cisco.com/c/dam/en_us/about/security/intelligence/blackhole.pdf

Source: https://www.cisco.com/c/dam/en_us/about/security/intelligence/blackhole.pdf