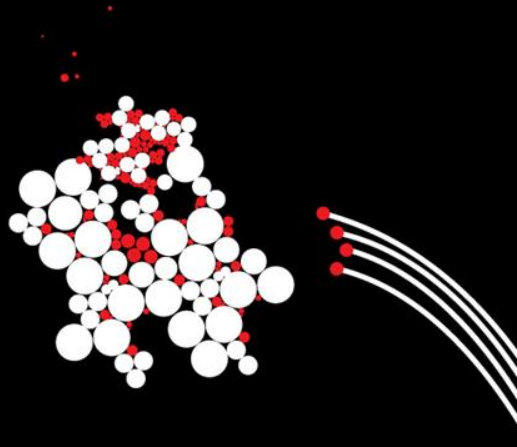
UNIVERSITY OF TWENTE.



Network Security

Attack and Defense Techniques

Anna Sperotto, Ramin Sadre

Design and Analysis of Communication Networks (DACS)

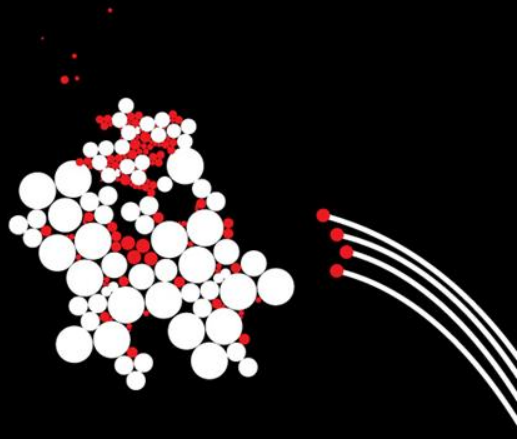
University of Twente

The Netherlands

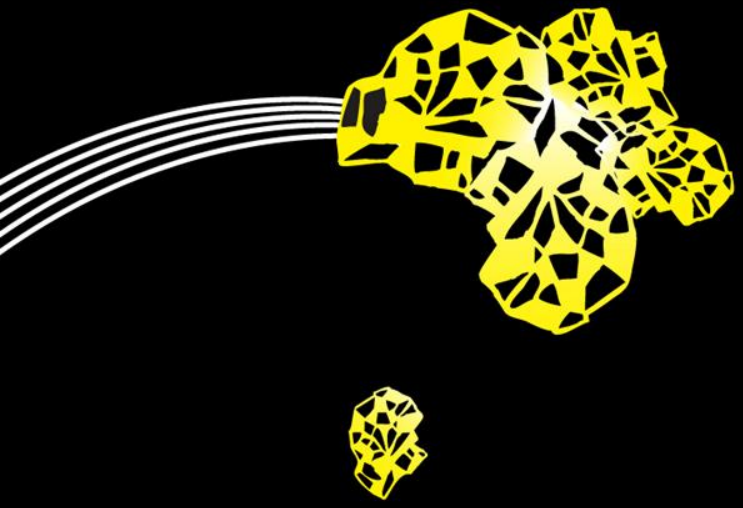
Attack Taxonomy

- Many different kind of attacks
- Possible classifications:
  - Attack type (scan, denial of service,...)
  - Attack target (a service, a network, a user,...)
  - Attack goal (crash the target, steal information, modify information,...)
  - ...

UNIVERSITY OF TWENTE.



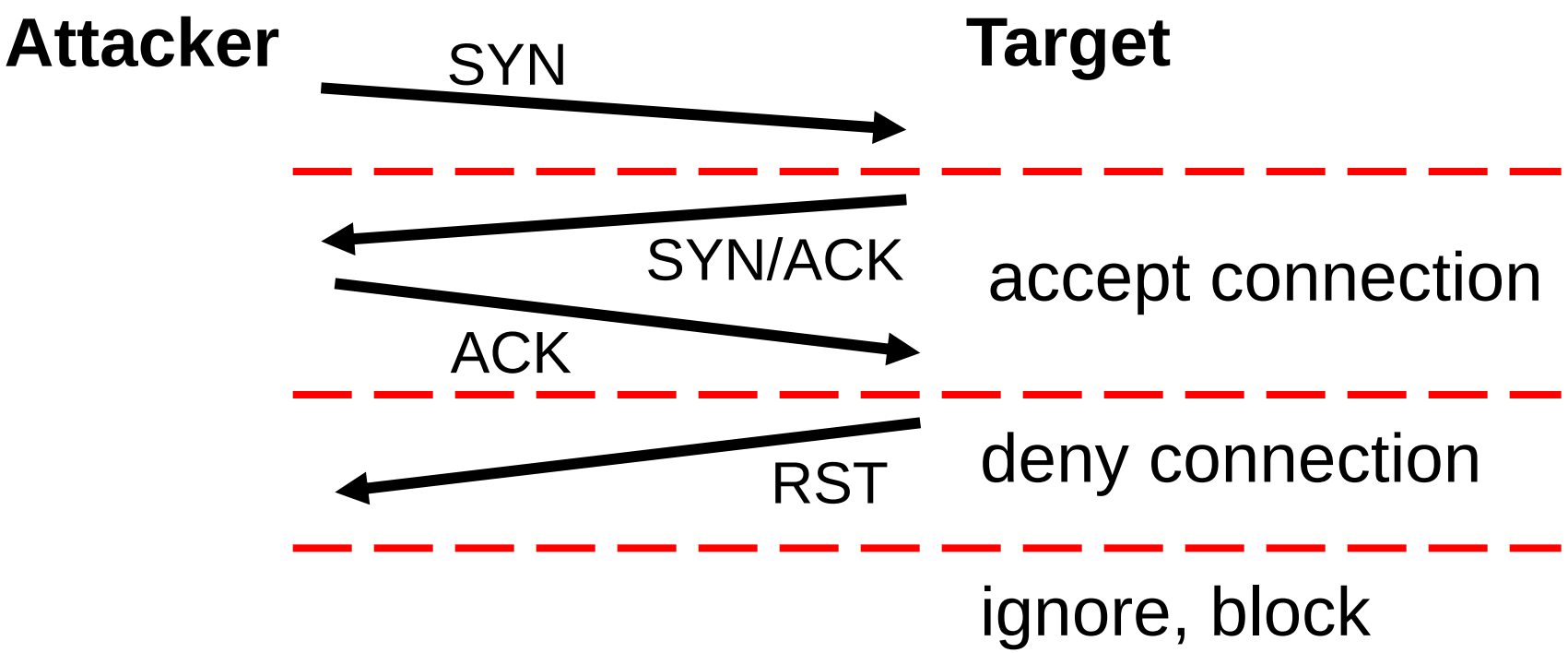
Scans



Port Scans

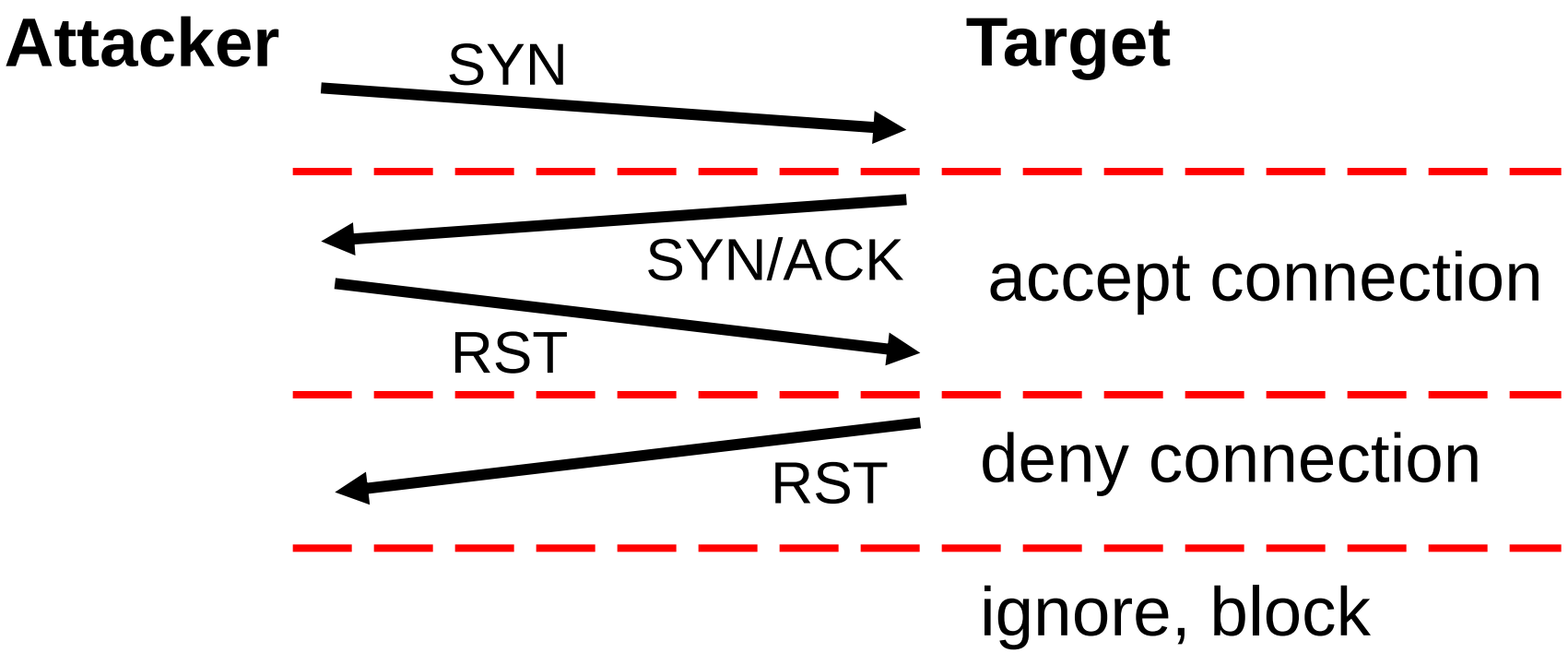
- Scans are *information gathering attacks*:
  - Find vulnerable services/hosts
  - Discover network topology (used IP addresses,…)
  - System fingerprinting
  - …
- Can be combined with a “real” attack, e.g., a buffer overflow (Ping Of Death, 1997)
- Tool for scanning: `nmap`

TCP port scan: regular connection



+ Easy to implement
– Slow

TCP port scan: SYN scan

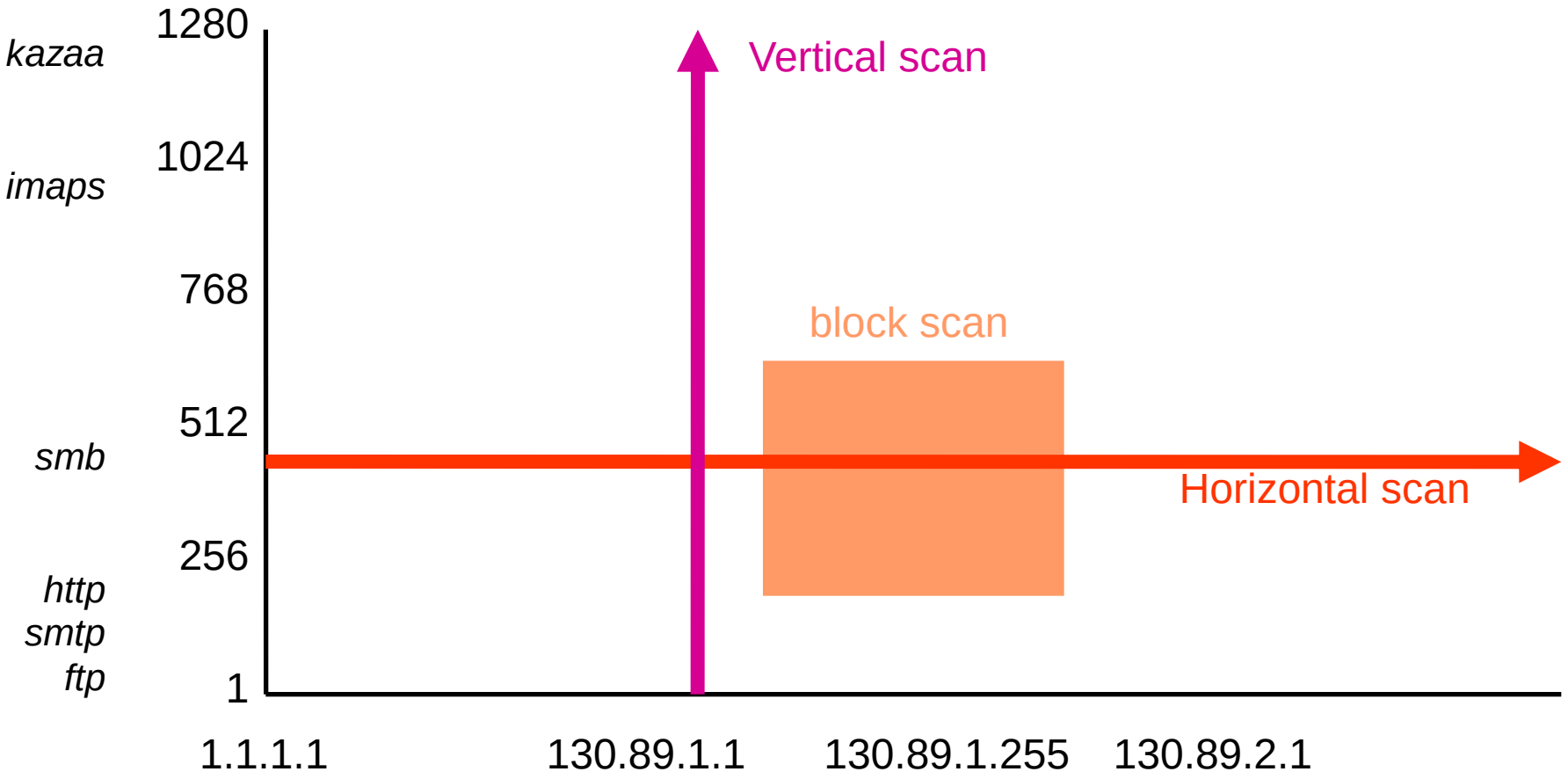


+ Fast
– Do-it-your-self

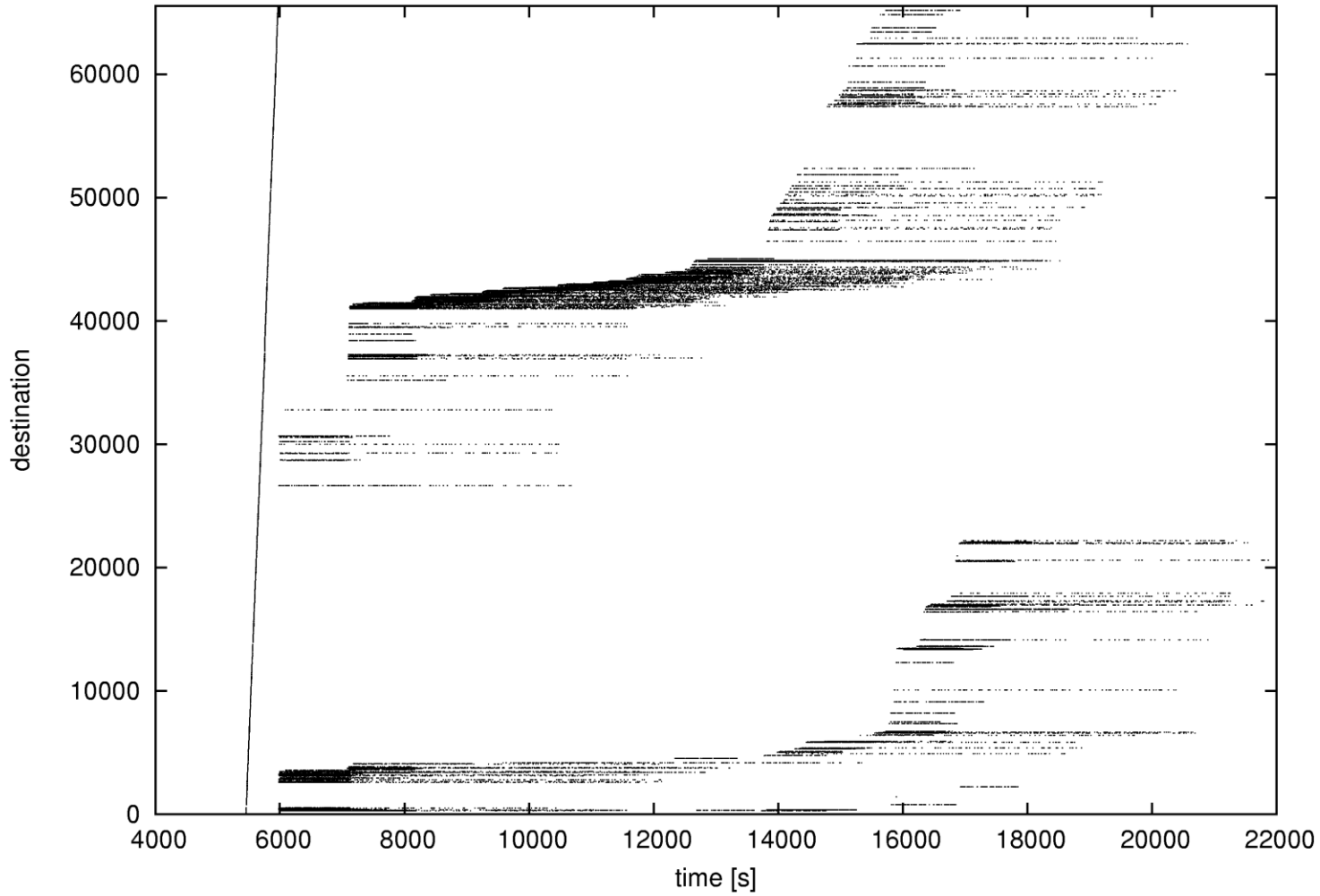
UDP port scan

- UDP is connectionless
- Two approaches:
  1. Wait for negative answer
     (ICMP message “port unreachable”)
  2. Wait for positive answer
     Example: send DNS query to port 53 and wait for DNS response

Types


1280
kazaa
1024
imaps
768
512
smb
256
http
smtp
ftp
1
Vertical scan
block scan
Horizontal scan
1.1.1.1
130.89.1.1
130.89.1.255
130.89.2.1

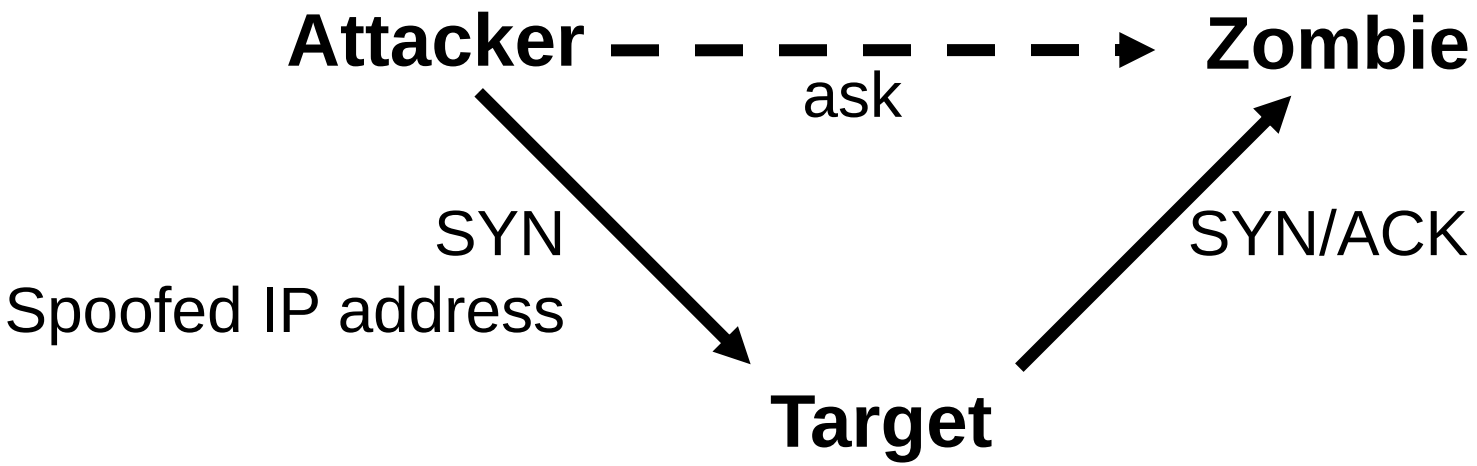
SSH attacker



How to hide

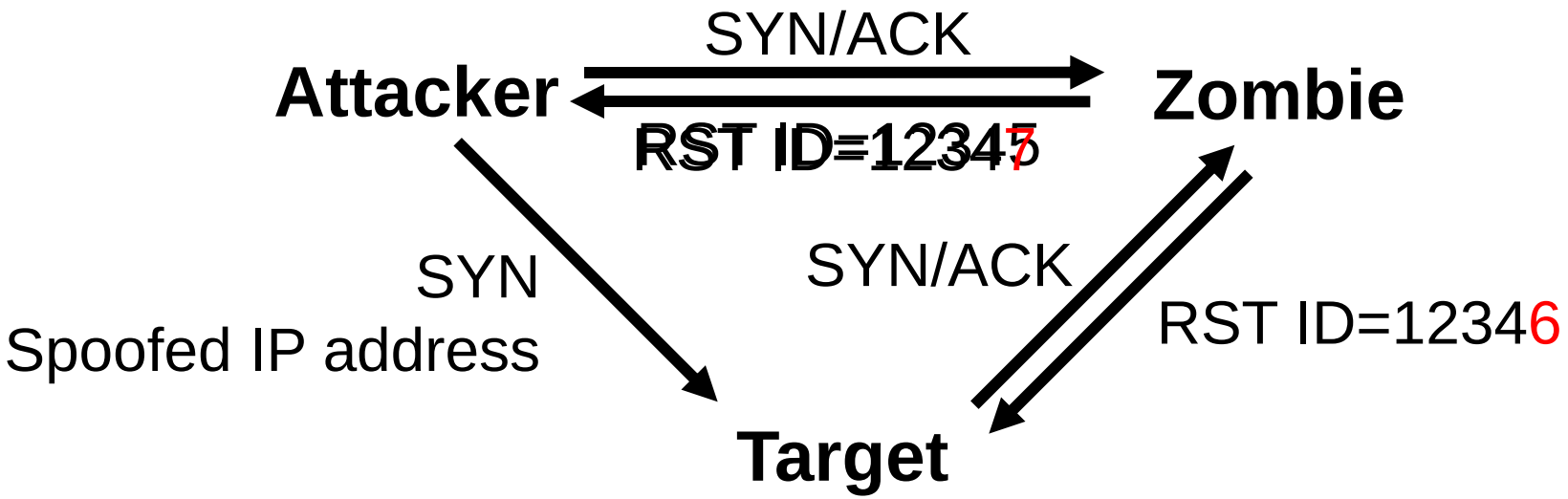
- The target system knows your IP address
  - *Slow scan*
  - *Distributed scan*: multiple, coordinated scanners
  - *Indirect scan*: idle scan (1998),...
  - ...

Idle scan

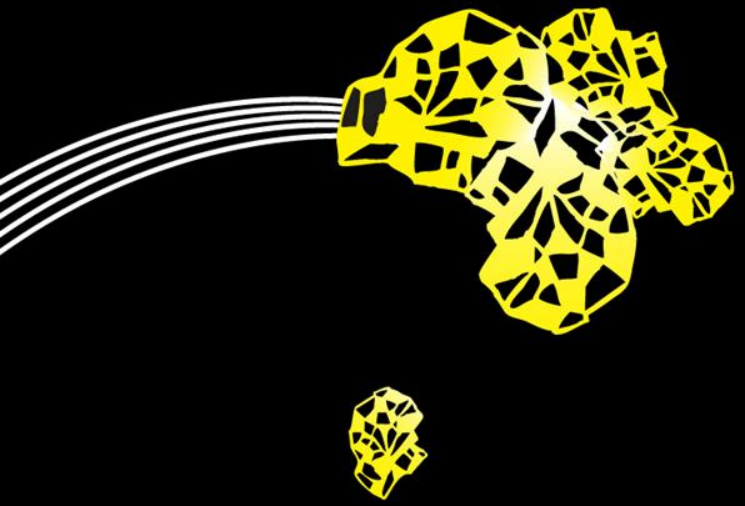


- How to ask the zombie?
- Fragment ID field in IP header

Idle scan


SYN/ACK
Attacker
Zombie
RST ID=12347
SYN
Spoofed IP address
SYN/ACK
RST ID=12346
Target

UNIVERSITY OF TWENTE.



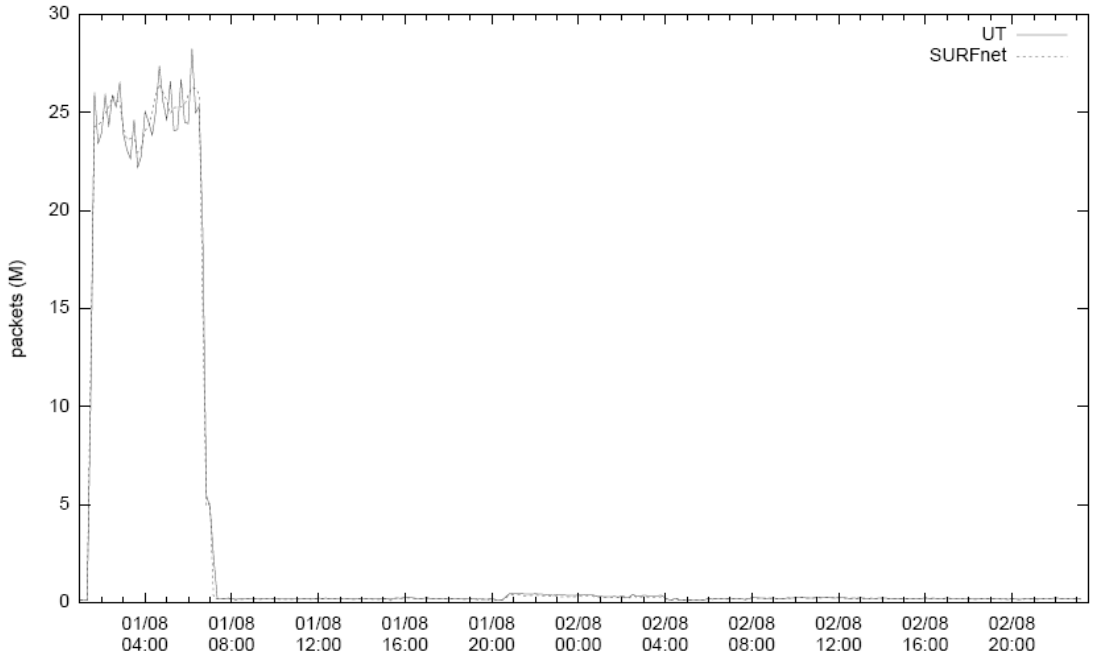
Denial of Service Attacks

Denial-Of-Service (DoS)

- Goal: overload or crash the server to make the service unavailable
- Types
  - Brute-force:
    - Send a lot of data (overload network), a lot of queries (overload server CPU),…
  - Semantic:
    - Exploit vulnerability (buffer overflow,…)
    - Send heavy requests (triggering complex operations)

DoS against DNS server

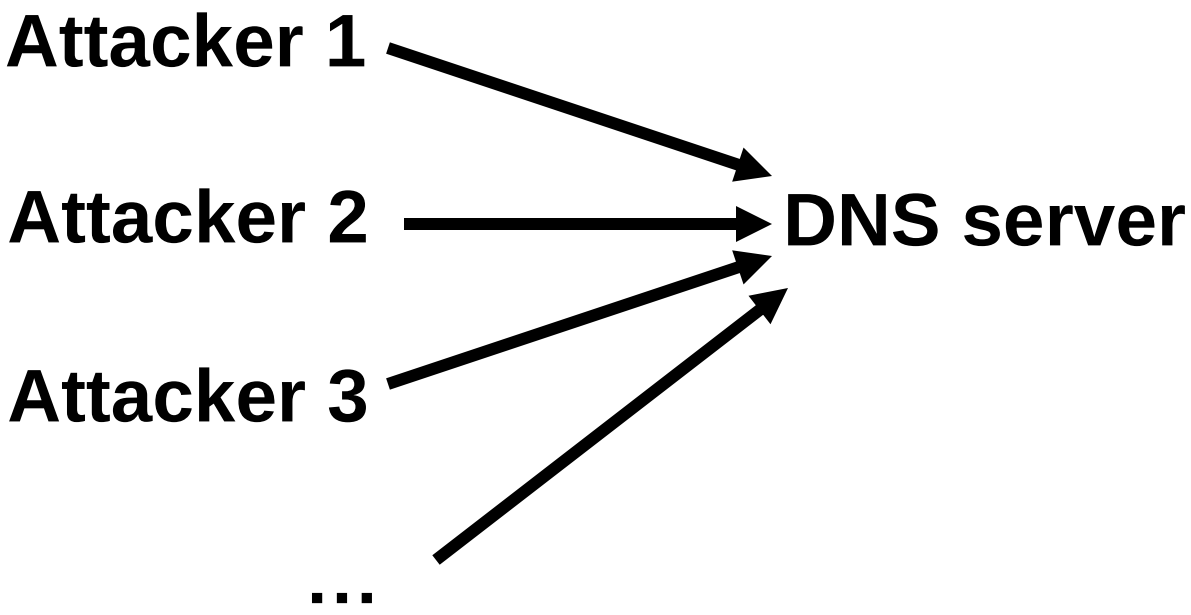
- Overload DNS server with queries



- Problems:
  - Attacker may be too slow (CPU, network bandwidth,...)
  - Defense: blocking the attacker's IP address is easy

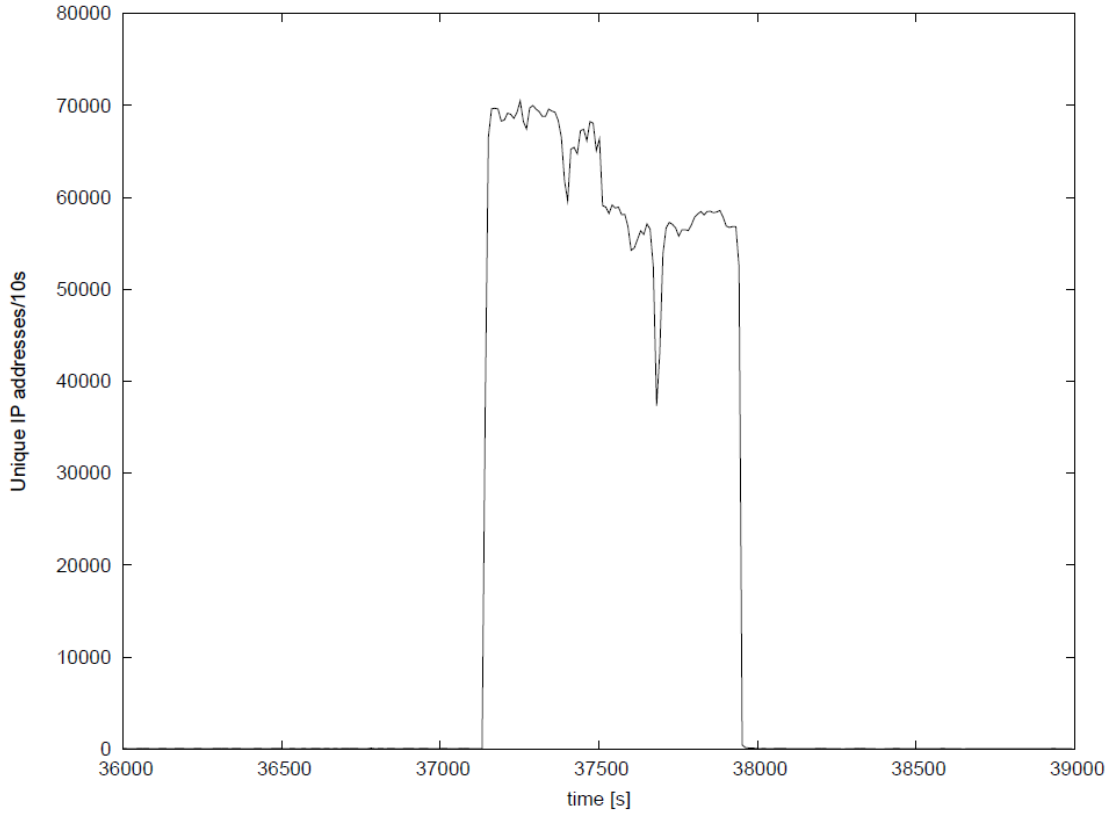
Distributed DoS (DDoS)

Coordinated attack from multiple hosts



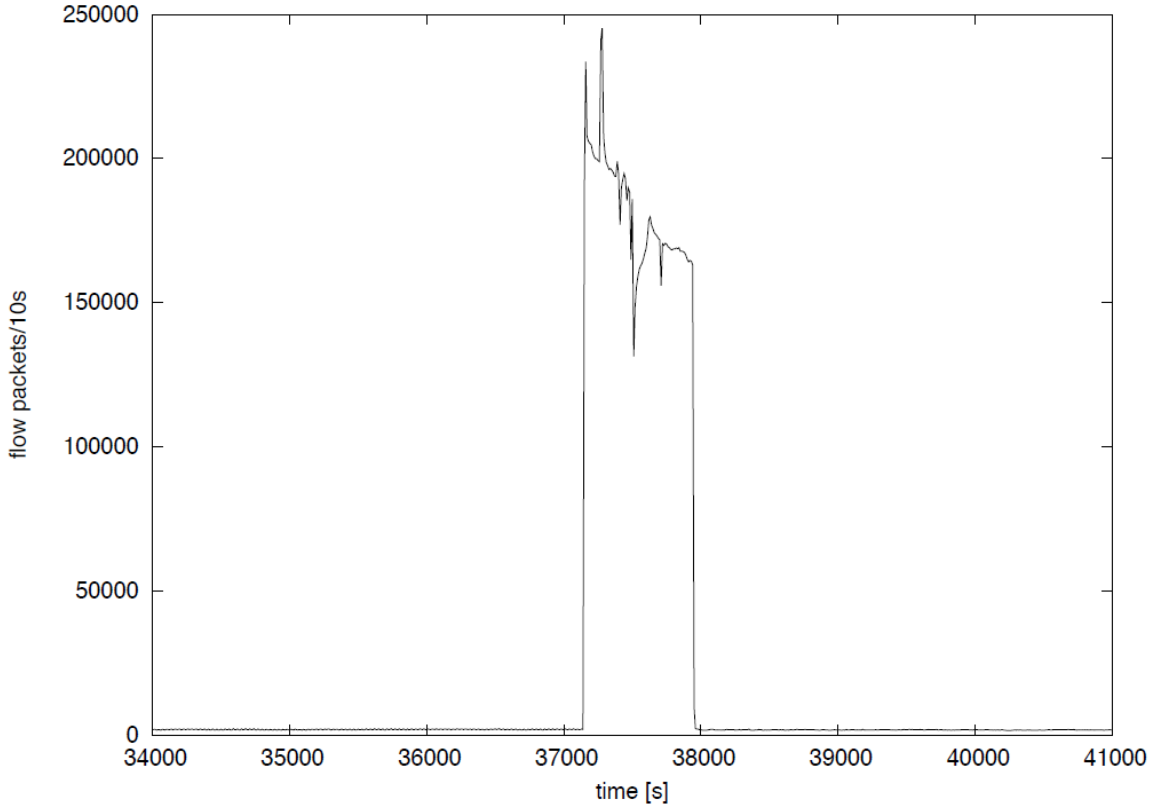
DDoS against IRC Server

- ~375 Million SYN packets in 800s

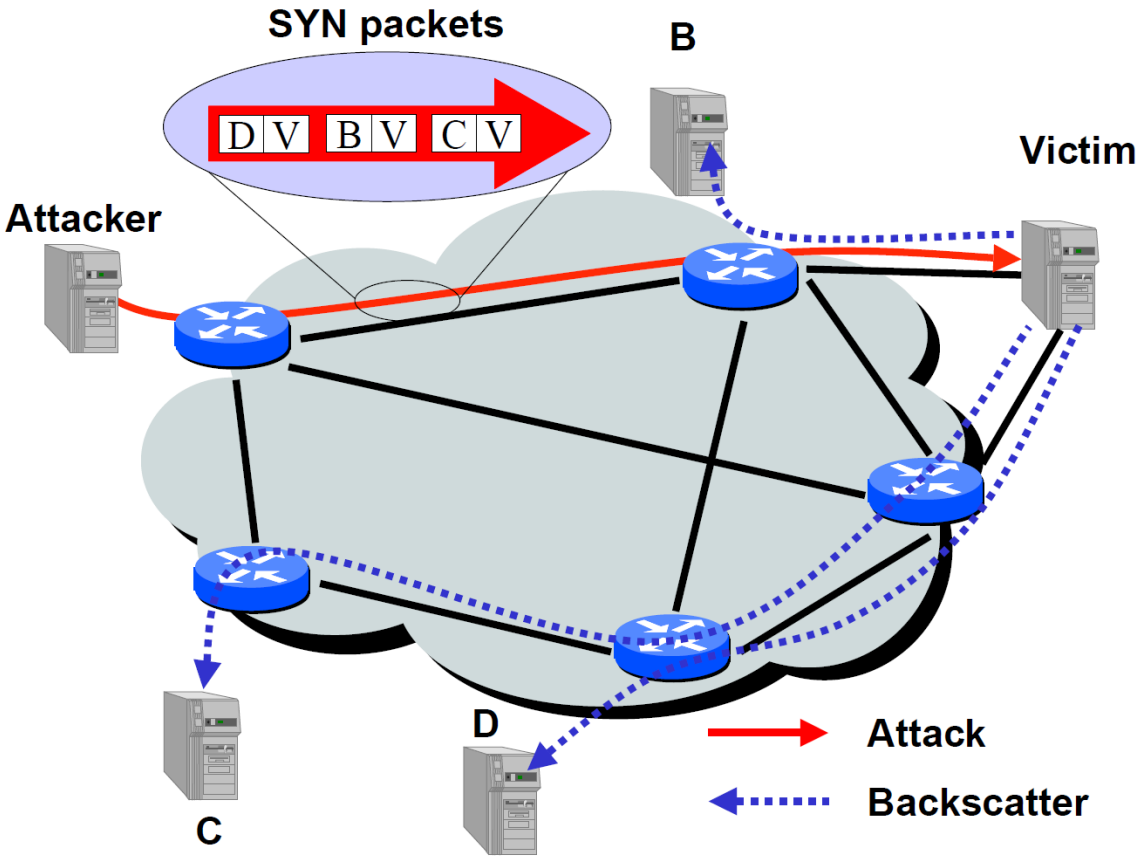


DDoS against IRC Server

- Attacks can have side effects on your monitoring/defense infrastructure
- Here: data loss at mirror port and at collector



On Large Scale:
Backscatter Analysis with a Network Telescope



(Source: Inferring Internet Denial-of-Service Activity, Moore *et al.*, 2001)

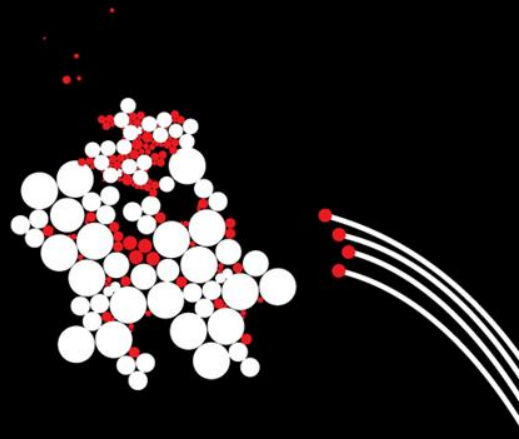
Backscatter Analysis for DoS attacks

- In Moore, 2001, a /8 network was monitored
- ~24.5 DoS attacks per hour

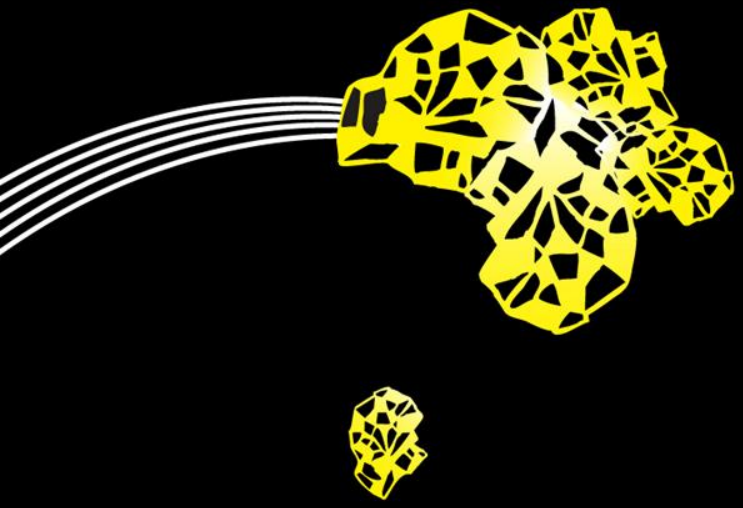
- Assuming uniformly distributed spoofed source addresses, this would correspond to

$$24.5 \cdot \frac{2^{32}}{2^{24}} = 6272 \text{ attacks/h}$$

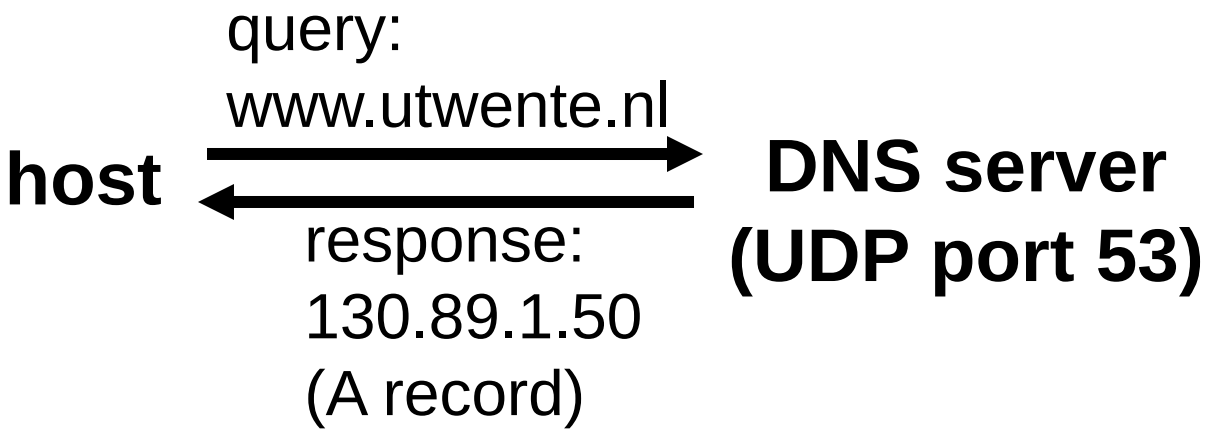
UNIVERSITY OF TWENTE.



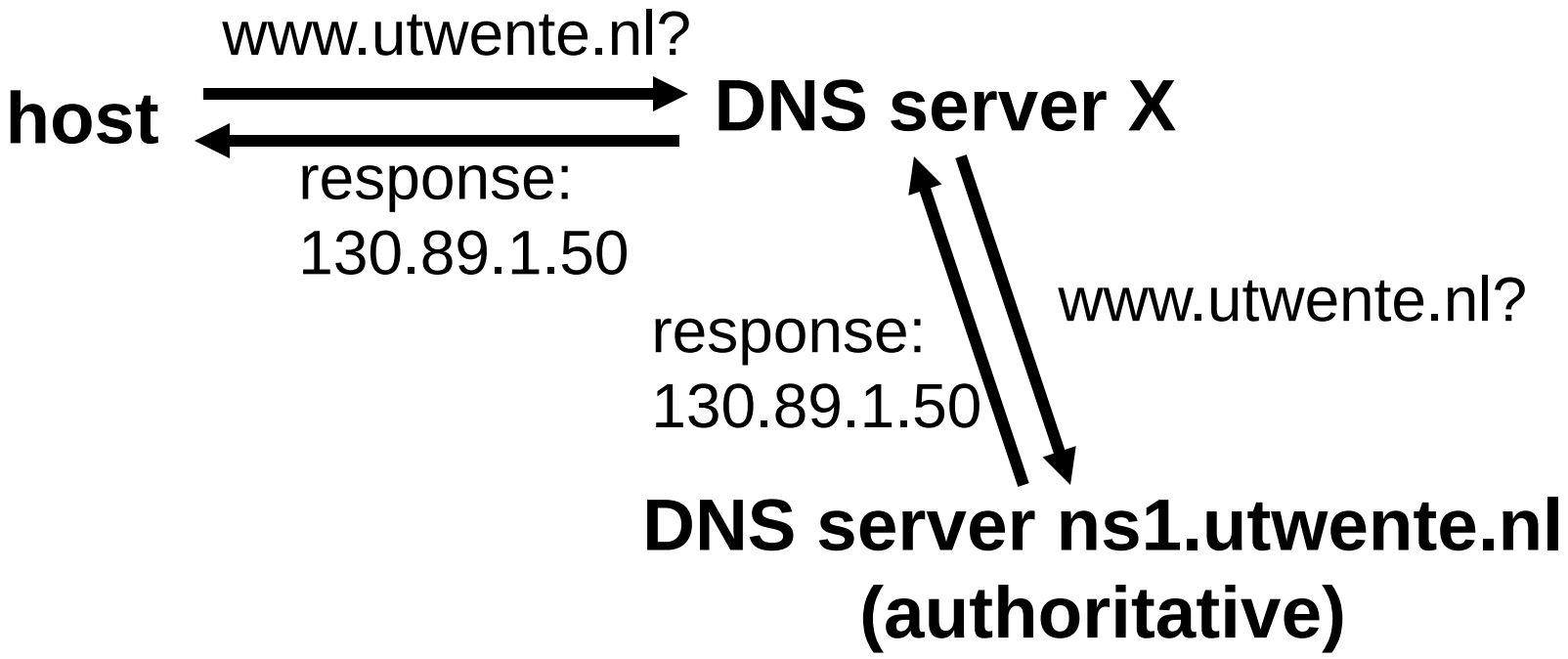
DNS



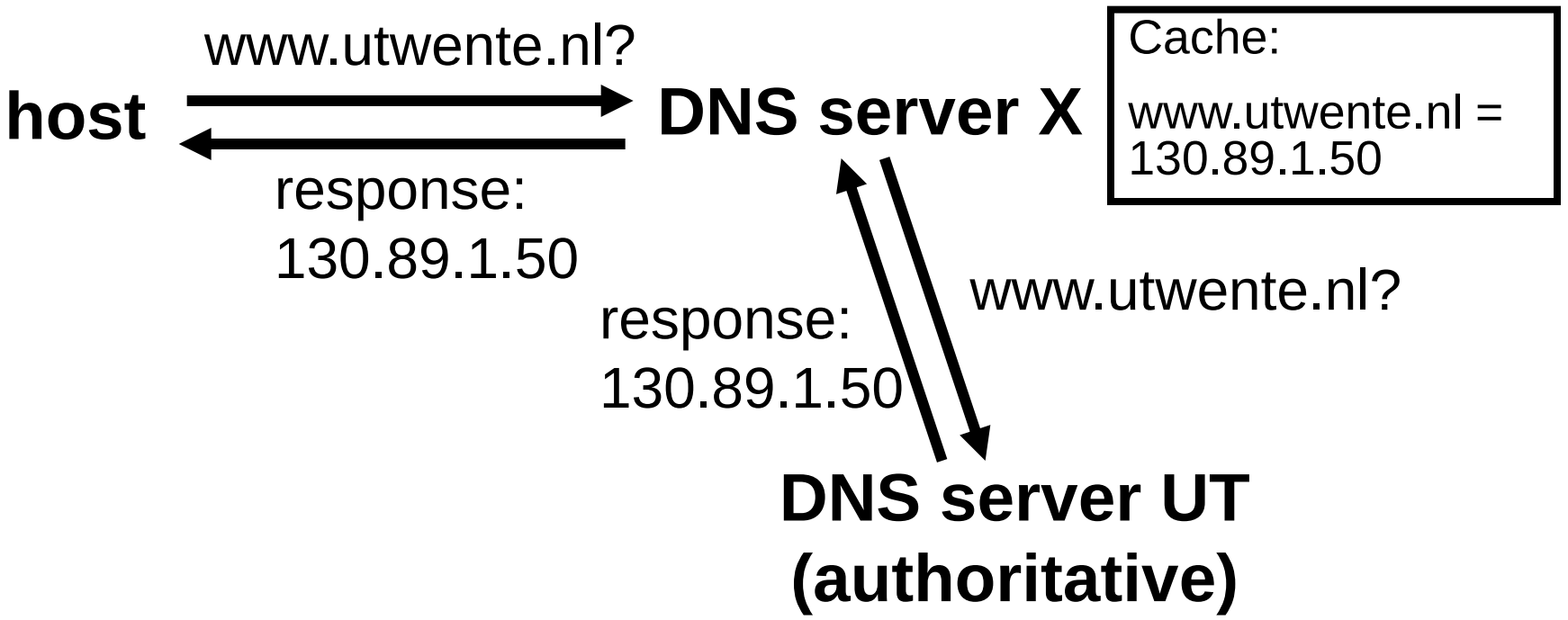
Simple DNS Query



Recursive DNS Query



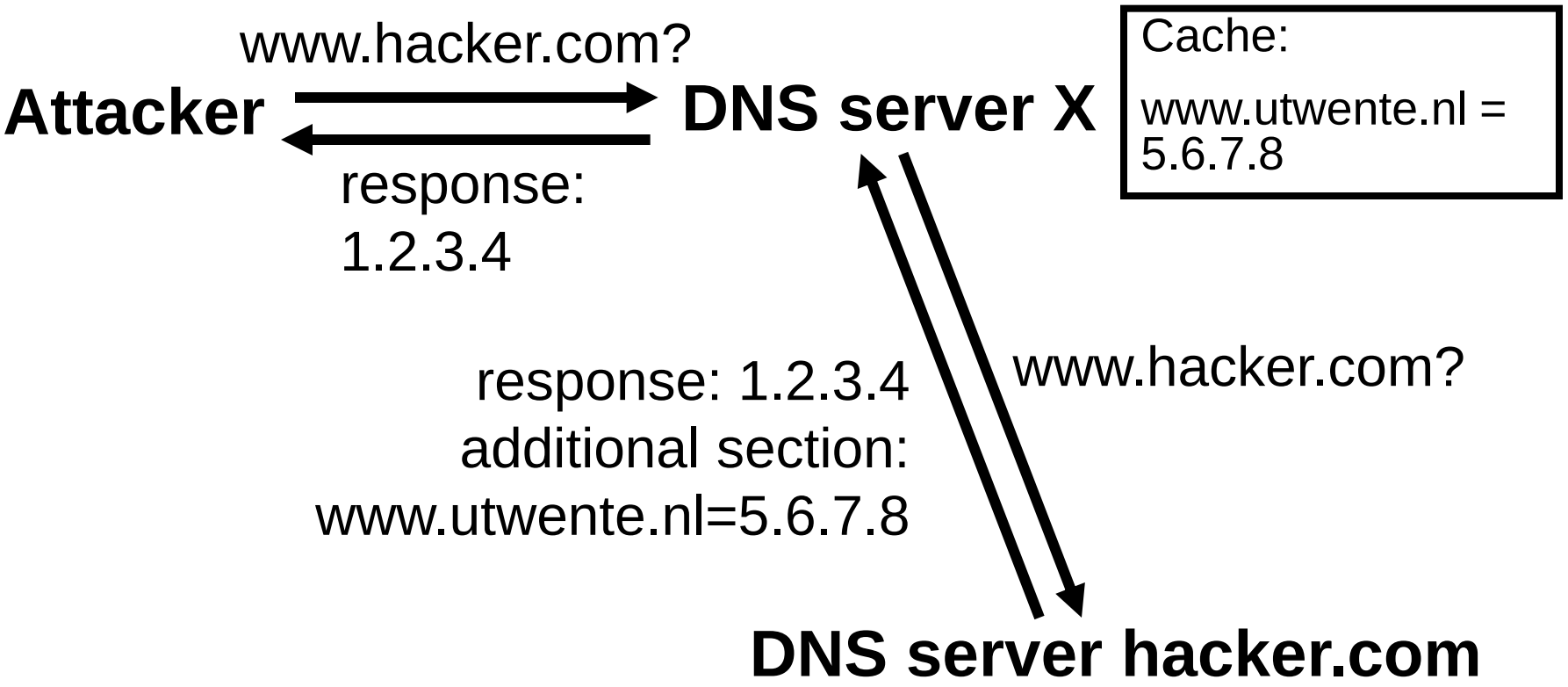
DNS Response Cache



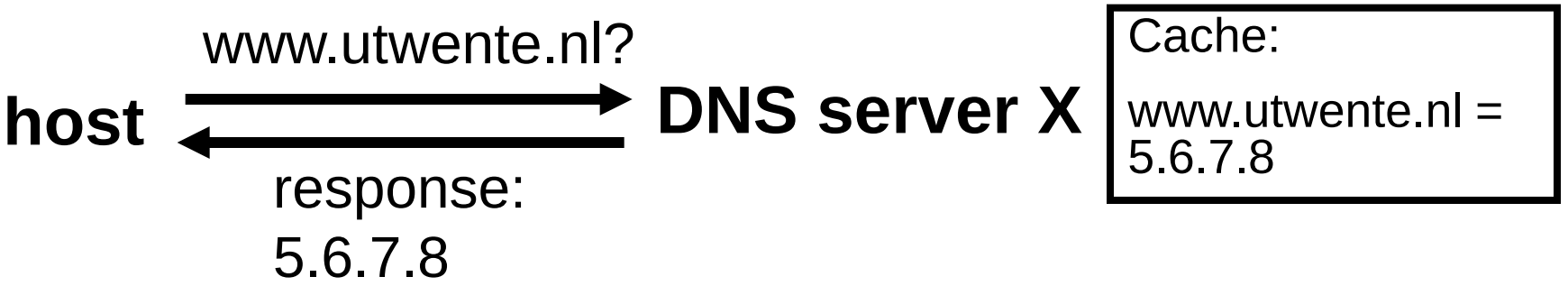
Cache Poisoning (Variant 1)

- Goal: compromise the DNS information
- Based on:
  1. Feature: DNS clients and servers cache responses
  2. Feature: DNS responses can contain additional entries
  3. Bug: some DNS server implementations don't validate the authority of a responder

Cache Poisoning (Variant 1)

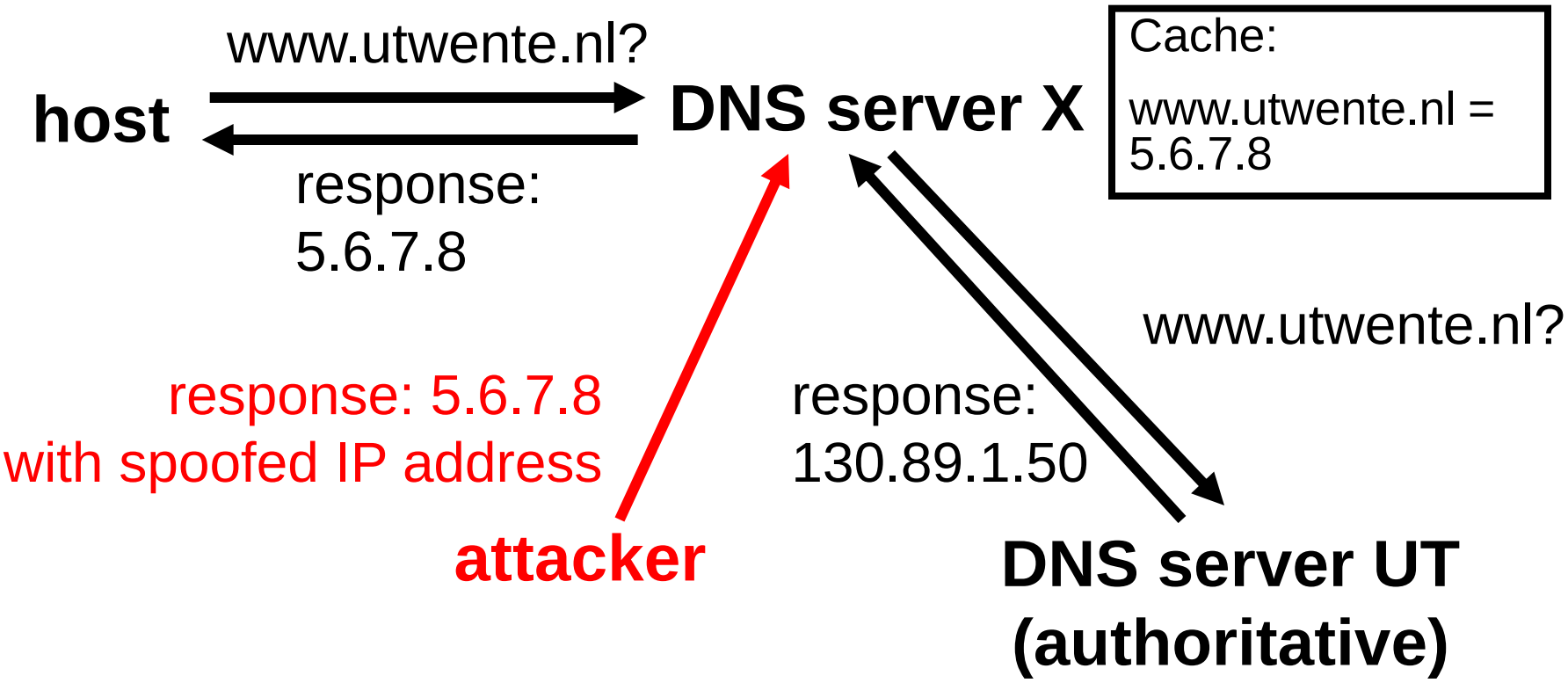


Cache Poisoning (Variant 1)



- also possible for entire domains: modify the cache entry for the nameserver of an another domain

Cache Poisoning (Variant 2)



Cache Poisoning (Variant 2)

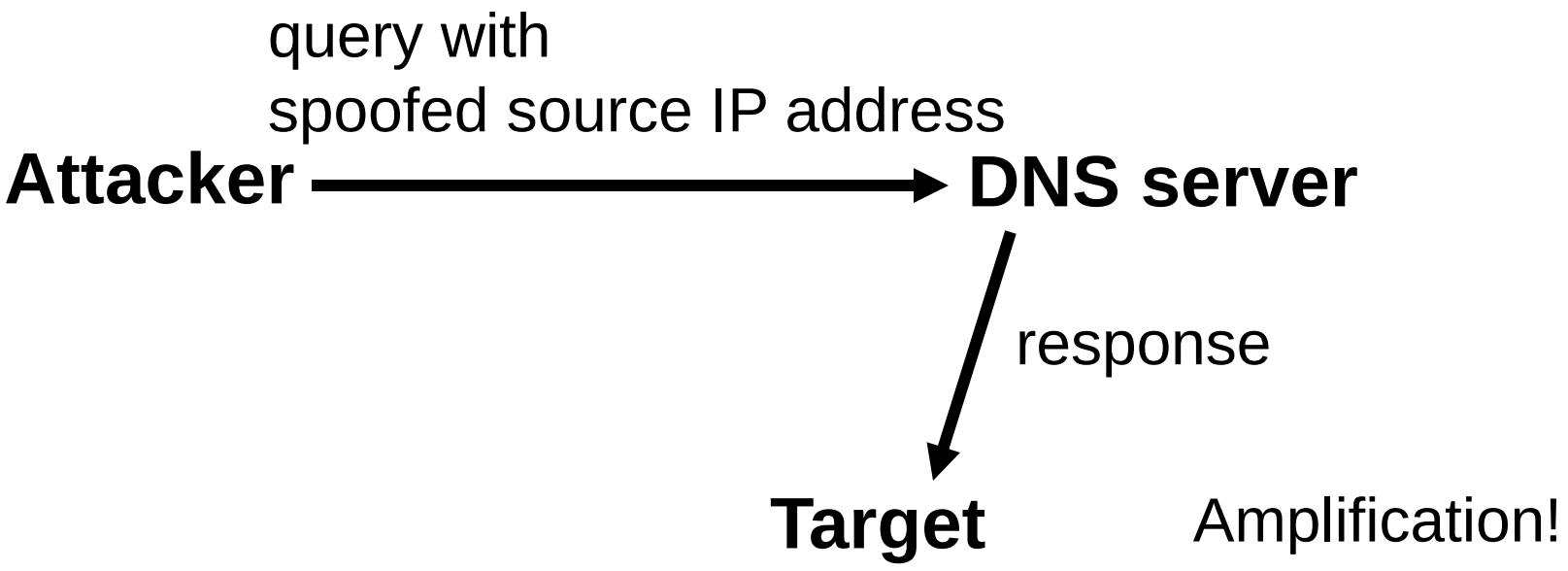
- Not so easy! DNS uses query IDs:
  - queries and responses carry a random ID
  - response ID must match query ID
- Attacker has to guess query ID
  - brute-force: send thousands of responses with different IDs
  - predict ID: some DNS servers use(d) flawed RNG to generate next ID

Cache Poisoning (Variant 2)

Brute-force attacks work!

- Some DNS servers always use the same source port to query other servers
- Solution: randomize the source port, too (July 2008)
- Attacker has to guess ID ***and*** source port

Reflected DoS Attack



- usually as distributed attack: multiple attackers, multiple DNS servers (DDos)

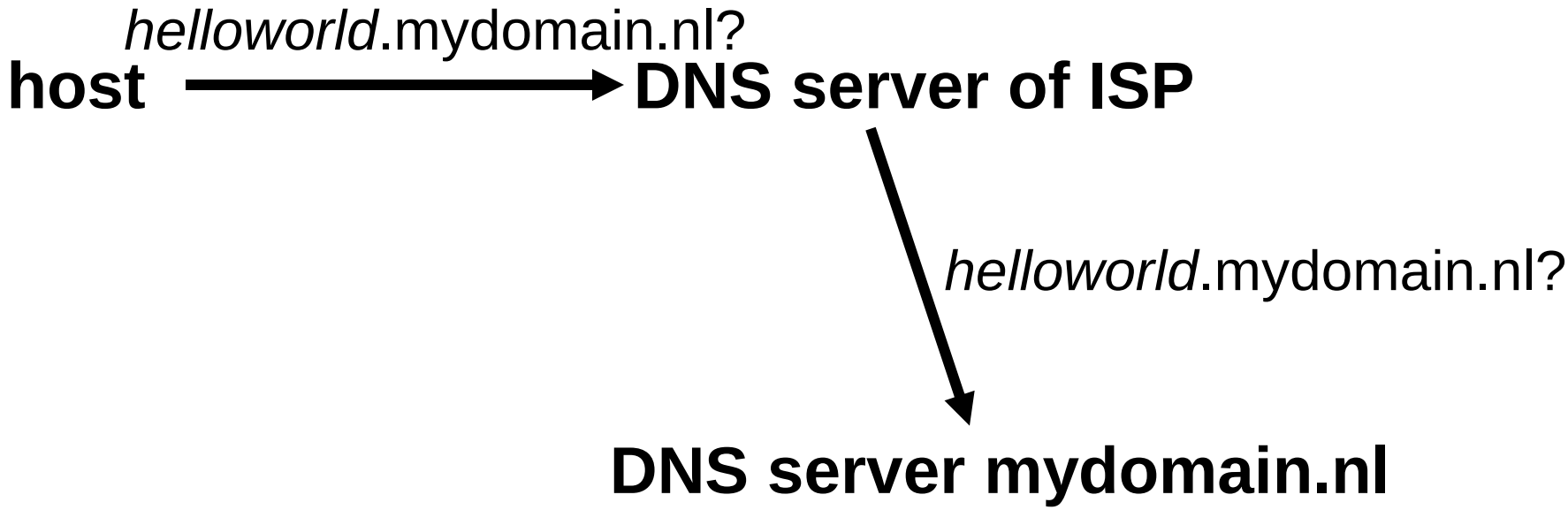
Amplification

- Initial DNS definition:
  60 bytes query → 512 bytes answer (8.5x)
- EDNS (RFC 2671) allows larger answers
- Combining different response types:
  answers larger than 4000 bytes possible (>60x)
- In 2006, Vaughn&Garon studied DDoS attacks with up to 140,000 DNS servers, resulting in 10Gbps

DNS tunneling

- You sit at the airport
- WLAN provided, but any access to a Web server, FTP, P2P,… is chargeable
- Is there a way to avoid the fee?
- Would it be an *attack*?
  - You are bypassing the billing/security policy of your ISP
  - Data exfiltration for cyber-espionage

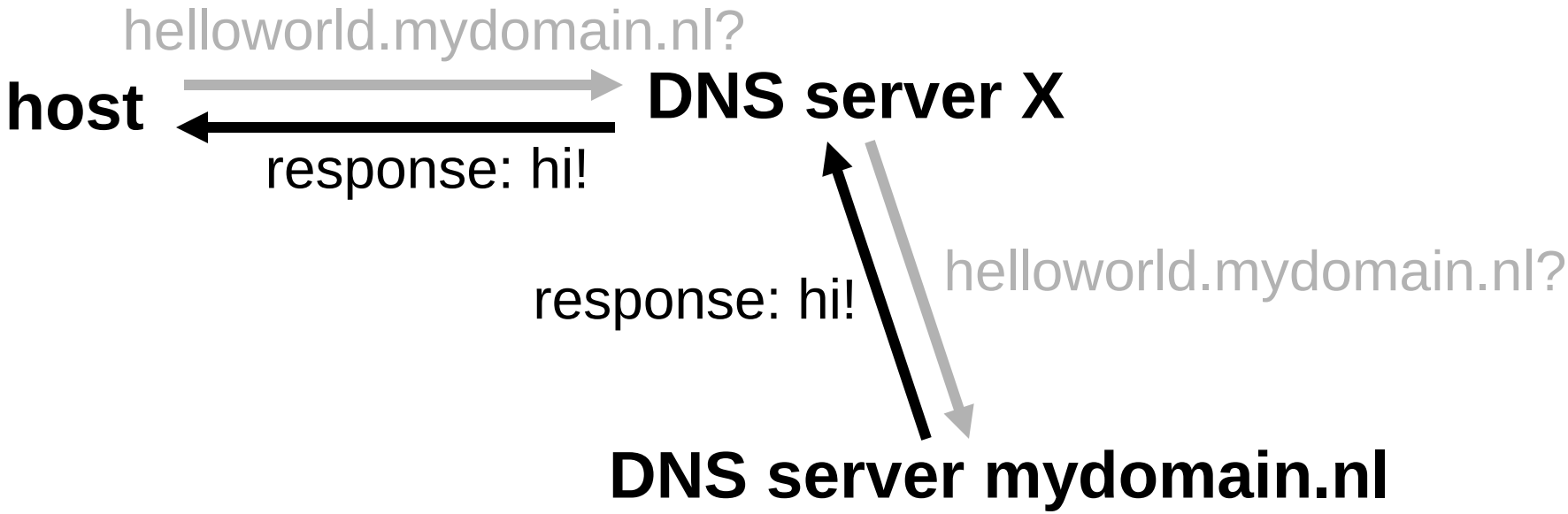
DNS tunneling: upstream



DNS tunneling: upstream

- Query can contain up to 252 characters
- Character set restricted: not case-sensitive,...
- ~5 bit/character, ~110 bytes

DNS tunneling: downstream



DNS tunneling: downstream

Main limitation:

- Response < 512 bytes to prevent fragmentation

Server responds with TXT-record:

- Character set restricted: 7 bit ASCII
- ~6 bit/character, ~220 bytes

As for amplification, EDNS (RFC 2671) allows larger answers

Using MX-records and A-records is possible, too, but more complicated (data may be reordered)

Example of DNS Tunneling (Iodine)

1329812676.512747 IP 192.168.1.2.60531 > 192.168.1.1.3128: Flags [S]

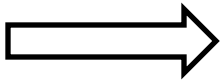
1329812676.515310 IP 1.1.1.1.51823 > 2.2.2.2.53: 22911+ [1au] NULL? 0eaba82M-J2hbM->M-nYM-VwjM-GM-MRbM-^M-^PM-\M-UM-HcvM-DtimM-eM-`M-KyM-aM-VM-IM-yM-yM-BM-jdilmnuM-iM-bM-ktaM-^XyUwtf.M-BM-^M-o8M-]M-=M-xM-=M-FouZzM-JwaeM-NaM-u

1329812676.524541 IP 192.168.1.1.3128 > 192.168.1.2.60531: Flags [S.]

1329812676.525743 IP 2.2.2.2.53 > 1.1.1.1.51823: 15184 1/0/1 NULL (140)M-N.test.domain.nl. (130)

1329812676.524573 IP 192.168.1.2.60531 > 192.168.1.1.3128: Flags [.]

1329812676.526742 IP 1.1.1.1.51823 > 2.2.2.2.53: 30638+ [1au] NULL? 0ibbb82M-J2hbM->M-nYM-VgjM-GM-MBbM-^M-^PM-\M-TM-XcvM-DtimM-eM-`M-KyM-aM-VM-IM-yM-yM-CDYM-eM-X3qWgM-JM-SM-qSM-?M->M-bYyCU.xpM-_M-VM-`M-HEM-LJM->M-nf6upM-{M->.test.domain.nl. (126)



1329812676.557242 IP 2.2.2.2.53 > 1.1.1.1.51823: 22911 1/0/1 NULL (144)

1329812676.525189 IP 192.168.1.2.60531 > 192.168.1.1.3128: Flags [P.], (request web page)

1329812676.558096 IP 1.1.1.1.51823 > 2.2.2.2.53: 38365+ [1au] NULL? 0mbbc82M-J2hbM->M-nYM-VhdM-yEM-rdM-?M->M-q5MM-tcvM-DtimM-eM-`M-KyM-aM-VM-IM-yM-yM-CDYM-eM-X3qWMM-JM-SM-CM-CM-DdbM->M-bM-p4.CM-=wM-icOM-x4oM-YM-kM-gM-SiHM-OM-guM-JcPM-<M-=rM-K0M-rf8M-cM-=M-XPgM-@M-HM-RM-\5FM-SM-uM-yM-CM-PM->GM-]M-hiM-?M-wQM-KFM-HM.0M-wM-zM-_UM-ZM-MwM-RM-C6M-?M-PpWM-tRPM-RM-fWyuM-\qM-FGtM-NBM-sgM-<huuTNl6NQ1FM-KvSkWM-H9ESaIM-AX.M-OHM-OM-bYM-wM-PM-C3MM-MM-dM-HAM-\3rM-bM-LMM-QfM-^ALM-UM-g18UhM-]CQaM-K6M-lM-mlM-IM-`M-naIDM-NM-cM->.test.domain.nl. (274)