Combating DNS amplification attacks using Cookies

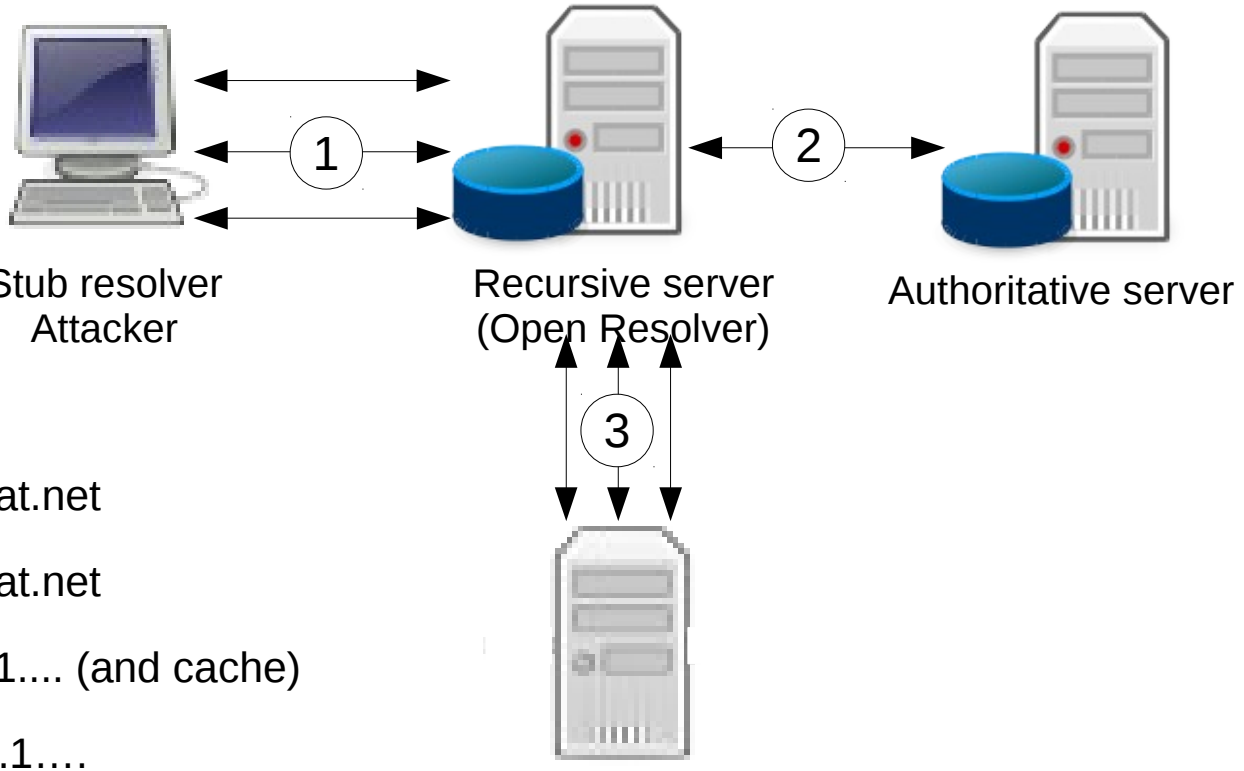
By:
Sean Rijs

Supervisor:
Roland van Rijswijk
SURFnet

Agenda

- I am going to do my presentation

DNS amplification attacks



1)query ANY delaat.net

2)query ANY delaat.net

response 1.1.1.1.... (and cache)

3)Response 1.1.1.1….

$$\frac{response\ size}{query\ size}$$

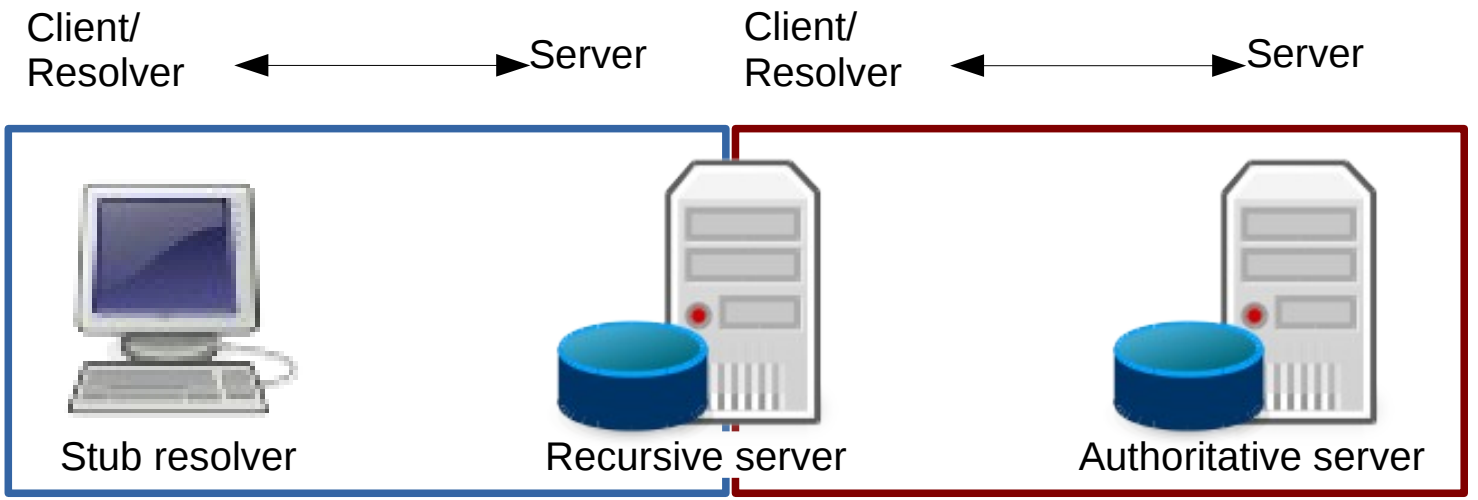
	Q-Size	R-Size	Amplification factor	Attacker	Victim
	40	512	12.8	100M	1.28G
EDNS0	40	1472	36.8	100M	3.68G
EDNS0	40	4096	102.4	100M	10.24G

Table by Rijswijk-Deij et al. [*DNSSEC and its potential for DDoS attacks*]

DNS Cookies
IETF Internet Draft

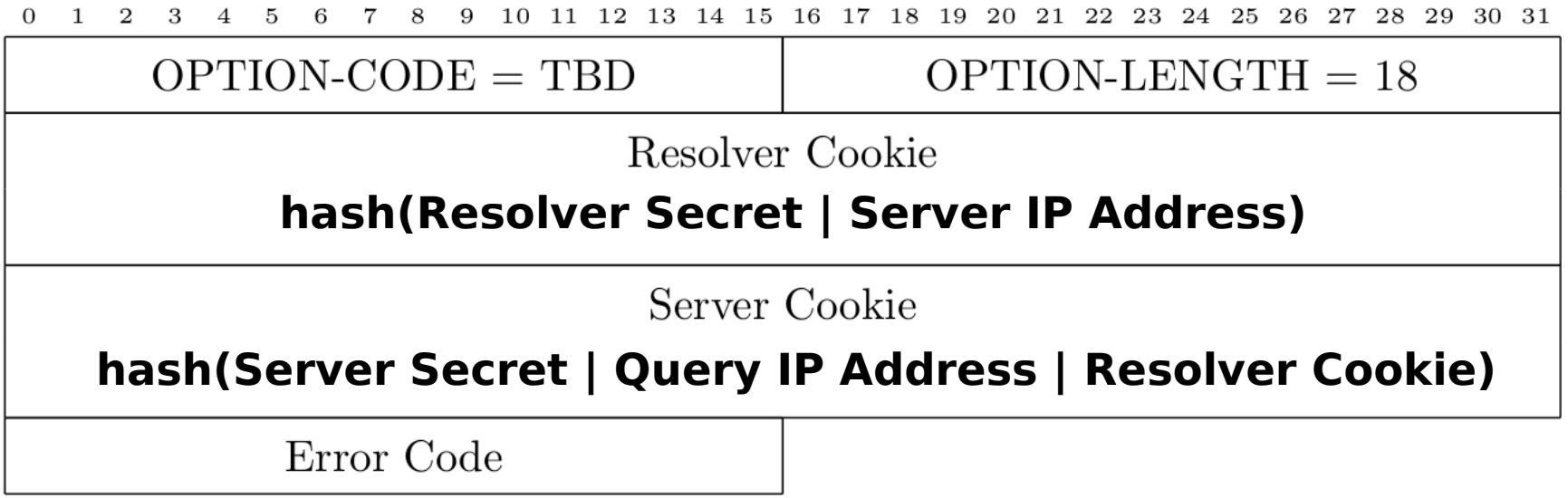
- By Donald Eastlake 2006-2014
  - Authentication of source IP
  - Off-path
- No pre-configuration required

- Research question:
  - Is the draft effective against DNS amp. attacks?



Terminology confusing?

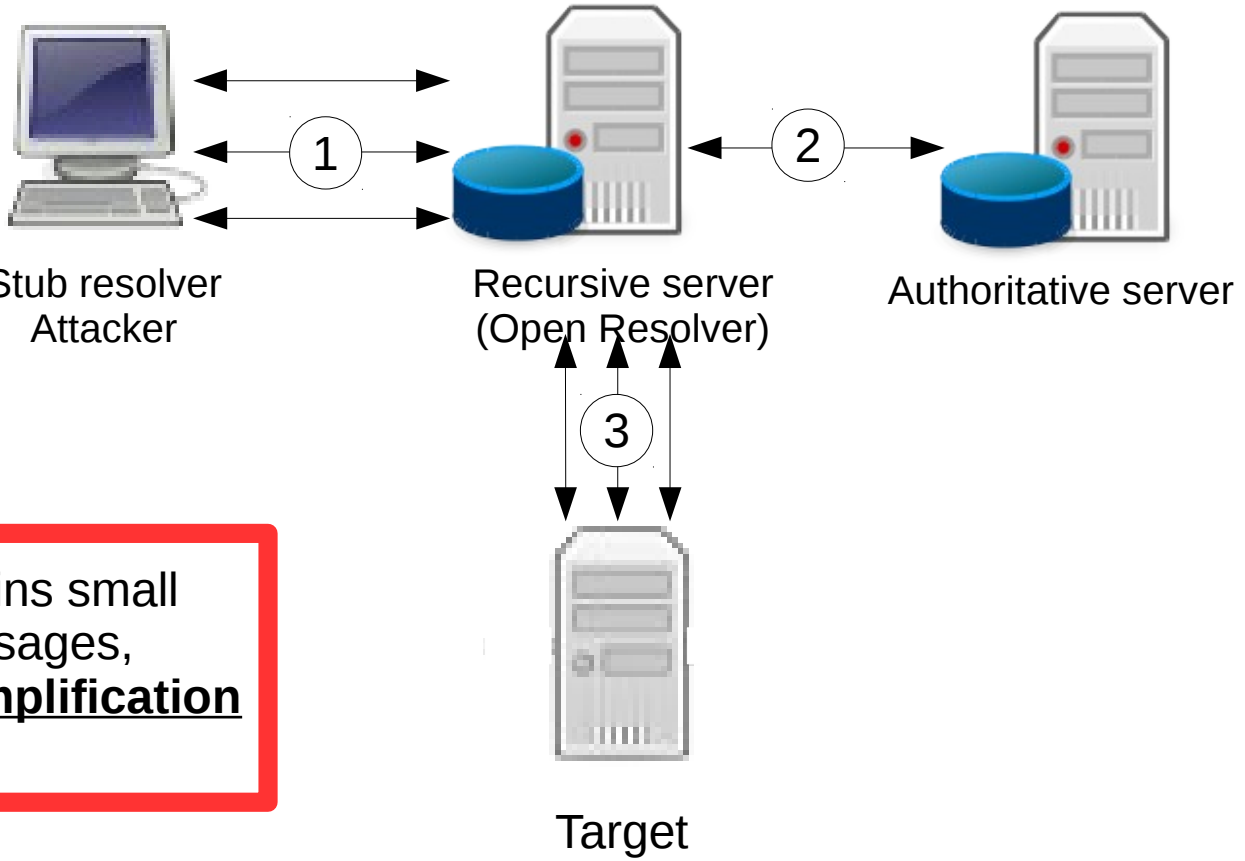
Cookies
OPT RR
(EDNS0)



- May occur once
- Max. 22 bytes
- Proposed hash = FNV-64

- Costs?
  - Initially 2x RTT
  - Hashing
  - Caching

What if?



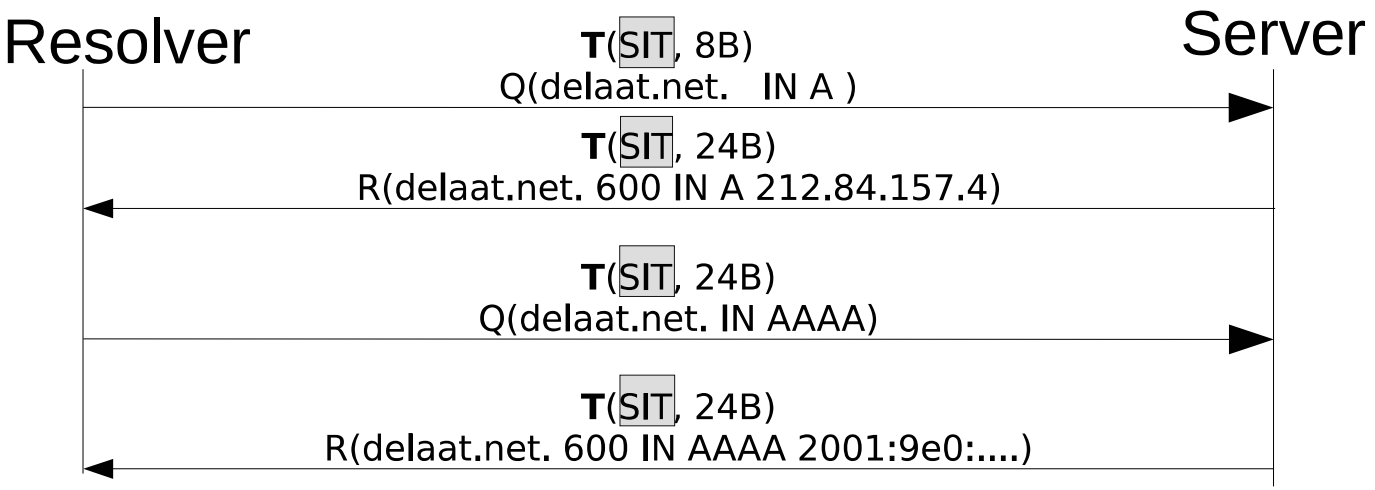
3 just contains small error messages, **no big amplification**

Policy

- ~~Disabled: do nothing with cookies~~
- Enabled: opportunistic (recommends RRL on server side)
  - Not a solution for recursive servers
- Enforced: Ignore everything without Cookies
  - Not gonna happen (in the near future)
- Policy is important, as it determines the incremental implementation

Source Identity Token (SIT)

- BIND 9.10-P1 (two months ago)

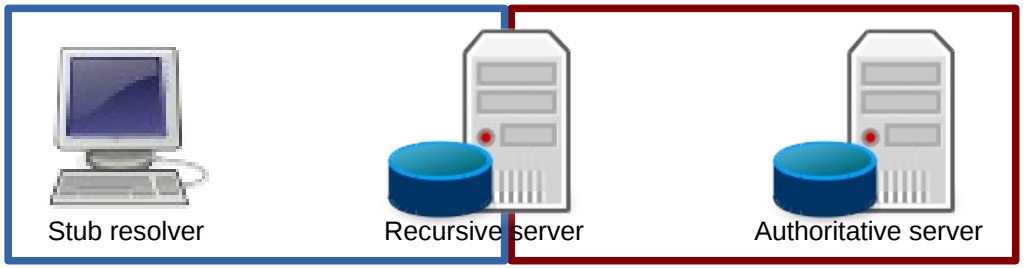


2x RTT has disappeared?

Differences SIT / Internet Draft

- Similar except:
  - Hashing: ~~FNV-64~~, AES-MAC, SHA1, SHA256
  - RRL: whitelists valid clients
  - Policy: no one is going to use it

Analysis of impact



- Stub resolvers are stateless
- A lot of end devices: bound by release cycles

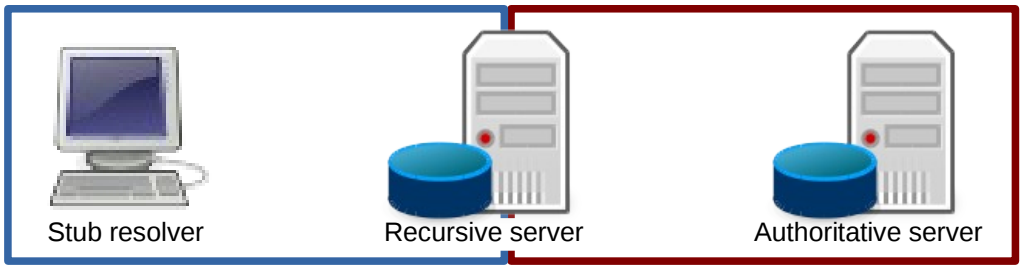
- Recursive server and authoritative are stateful
- Already use RRL

Measurements

- What do want to find out?
  - Do we need EDNS0 for normal use?
  - Do we need large response sizes for normal use?
- How?
  - PCAPs and EEMO

Measurement sources

- Stub resolver: www.nu.nl (with its adds) using:
  - Windows - Internet Explorer
  - OS X - Safari
  - Ubuntu Linux – Firefox
- Stub resolver: Alexa top 10 using:
  - Ubuntu Linux - Firefox
- Recursive server: SURFnet
  - 1500 – 2000 queries per second
  - 10m during a workday on noon

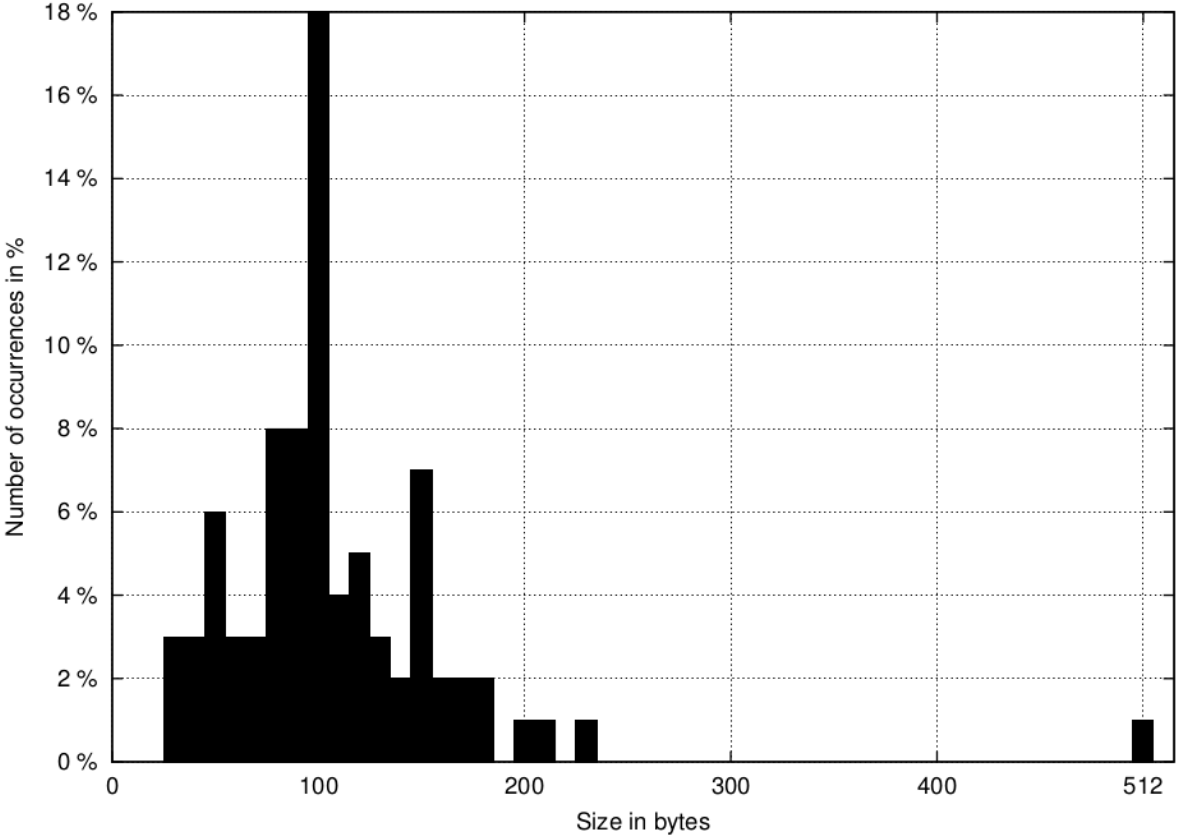


Stub resolver

- No EDNS0 found
- No large response responses:
  - Size <= 512 bytes
  - truncated/TCP communication = 0

Recursive server

- 22% EDNS0
- Average size
  - 133 B
- 99% <= **240 B**



Conclusion/Discussion

- Based on our results, we suggest unauthenticated stub resolvers should be limited to a max. response size of **240 bytes**
- Amplification reduced further:
  - **240 bytes = 6 amplification factor**
  - 100M = 600 Mbit/s

Q-Size	R-Size	Amplification factor	Attacker	Victim
40	512	12.8	100M	1.28G
40	1472	36.8	100M	3.68G
40	4096	102.4	100M	10.24G

Table by Rijswijk-Deij et al. [*DNSSEC and its potential for DDoS attacks*]

Conclusion

- RRL should not be used
  - Especially on recursive server
  - But authoritative can also be effected
- Policy for incremental implementation must be changed
- Terminology:
  - stub/recursive/authoritative
  - The cookie is actually a Message Authentication Code (MAC) and not just a hash

Discussion

- Do we need to authenticate the server?
- Yes, it provides off-path defense against:
  - Last mile problem in DNSSEC
  - Cache poisoning (by Kaminsky)

Future research

- Need more measurements
  - to confirm suggested DNS maximum response size
- FNV-64
  - The non-standard and untested hashing algorithm, which could provide performance gain. Is a performance gain required?

Questions