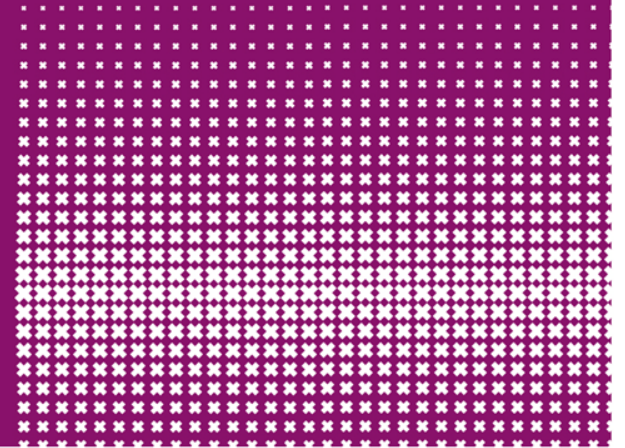




Ralph Dolmans



# A solution for the DNS amplification attack problem

July 4<sup>th</sup>, 2013

# Context

- Spamhaus was attacked with 300Gbps
- Every day attacks are getting bigger
- Sites can be held hostage, banks cannot talk
- Global problem, the end of the Internet?

# Annoy people by sending bricks

- Send an unsolicited brick by mail
- Annoying for the receiver, but only obstructive when done by many people at once



# Easy way to send lots of bricks



# 1: Sender verification

- Factory contacts customers to verify order
  - Dramatic change in order process
  - More work for bricks factory employees
  - More time needed to handle requests

= Three way handshake, DNS over TCP

## 2: Prevent sender address spoofing

- Validation at postal sorting center, only process orders when the delivery address is in the area in which the mail is posted
  - Only works when all postal sorting centers can be trusted

= BCP38

## 3: Rate limiting

- Limit the number of orders the factory handles per customer address
    - Factory can falsely drop orders, thereby losing money
    - Factory can falsely allow orders, thereby still sending unsolicited bricks
- = DNS Response Rate Limiting (DNS RRL)

# Shipping to intended users only





# DNS parallel

- Bricks factory = Authoritative name server (ANS)
- Local reseller = Recursive resolver (RRNS)
- Local customer = User of a specific resolver

# DNS amplification attacks

- Same solution:
  - ANS handles orders coming from RRNS
  - RRNS only handles orders coming from local users
- Instead of dropping unwanted orders, the ANS could apply a rate limit to enable debugging

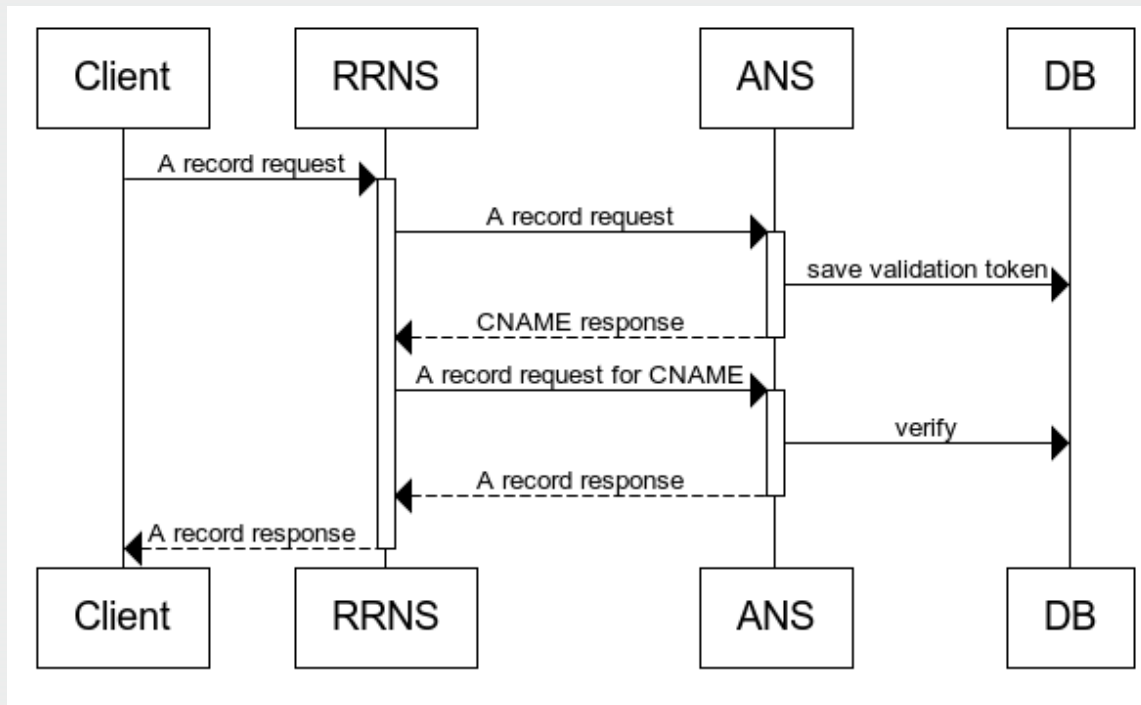
# Whitelists

- RRNS needs whitelist of customers
  - RRNS providers know the IPs of their network
- ANS needs global whitelist of RRNS servers
  - There are no list containing all resolvers, so we need a method to create this list

# Generating a global list of resolvers

- We cannot simply scan IP space as is done by <http://openresolverproject.org/>
- Log source address in requests at ANS
- Introducing integrity using a simple CNAME handshaking dialogue

# Simple CNAME handshake



# Custom ANS software

- Implemented using python + twisted
- ping val.stopddosattacks.org

```
ralph@debian:~$ ping val.stopddosattacks.org
PING B34tQ8sYzTjTHHda1372851505.val.responsible-resolvers.org (145.100.104.46) 56
(84) bytes of data.
64 bytes from 145.100.104.46: icmp_req=1 ttl=57 time=4.43 ms
```

- 1200+ resolvers in the MySQL database so far

# ANS whitelist check

- Using standard firewall instead of changing DNS software (BIND, NSD, PowerDNS)
- Firewall rules for ANS:
  - Accept packet when source on whitelist
  - Rate limit packer otherwise
- Does this perform?

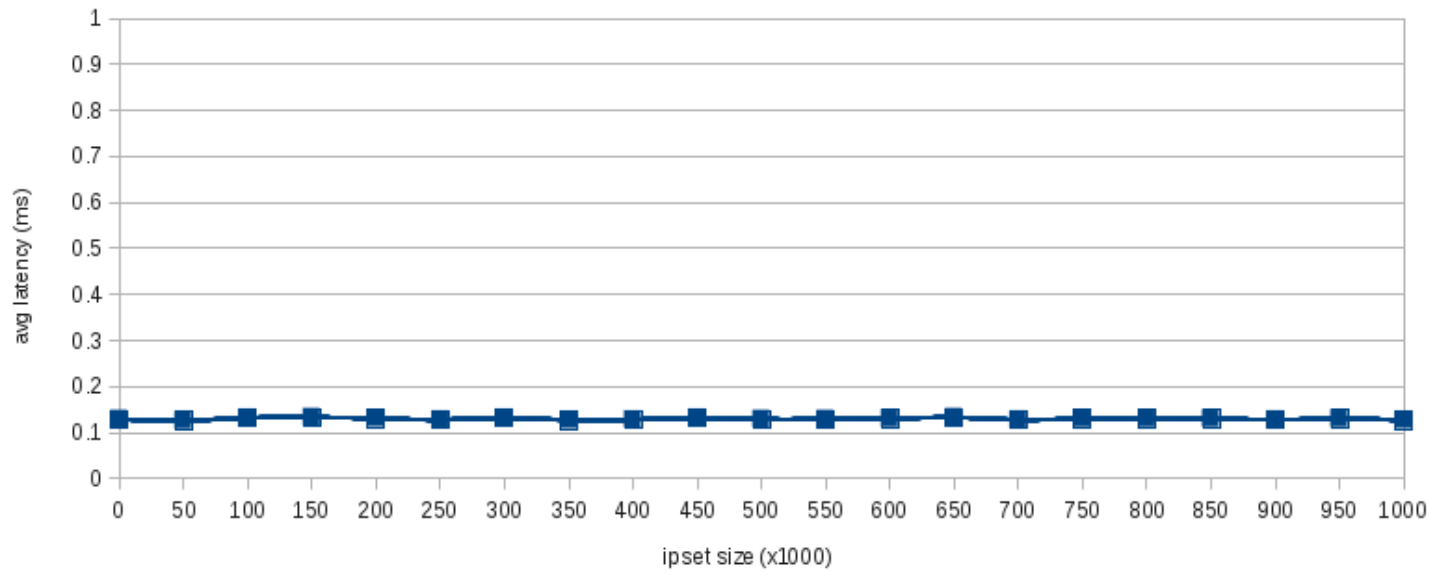
# Iptables + ipset whitelist

- Ipset for the whitelisted IPs
- Benchmarks:
  - Average latency, handling 10 million requests, 200K per second
  - CPU load for 1 million whitelisted IPs

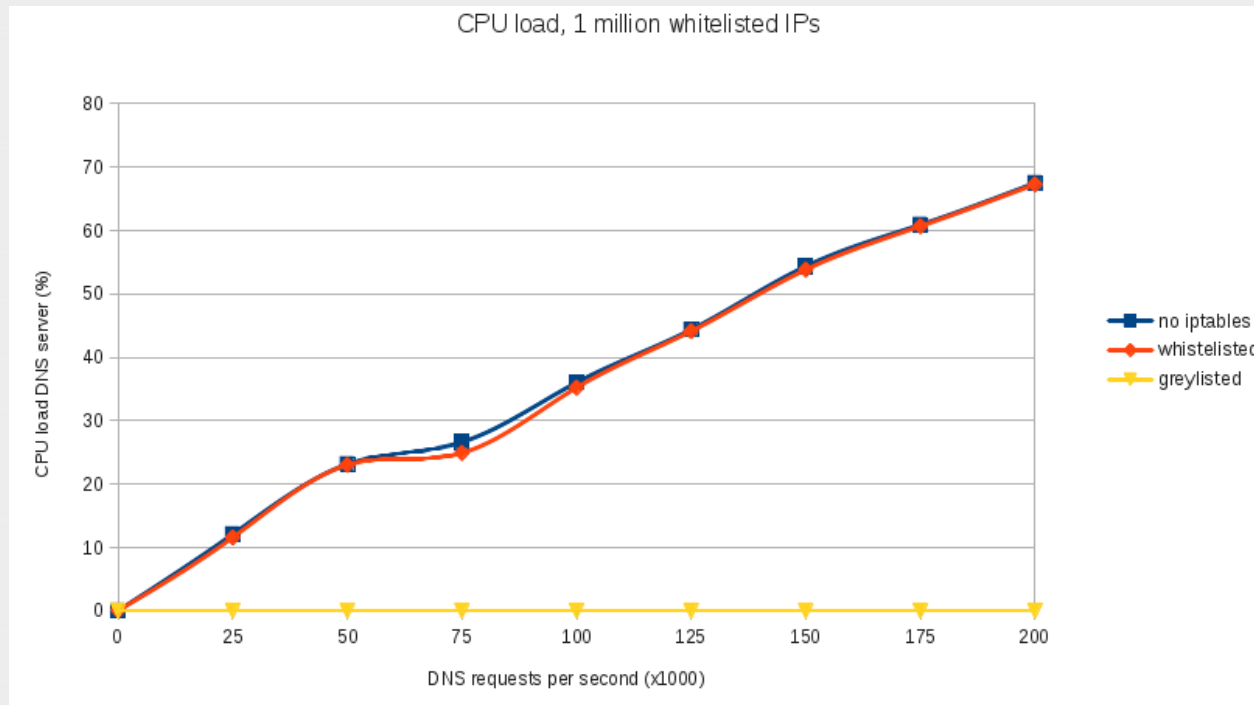


# Iptables + ipset latency

Average latency, 10 million requests, 200K requests per second



# Iptables + ipset CPU usage



# Promotion and education

- Next step:
  - Educate people about the attacks
  - Collect as many resolvers as possible
  - Encourage the use of whitelists on ANSs
- Two websites:
  - <http://stopddosattacks.org>
  - <http://reliablename servers.org>

# Stopddosattacks.org

- Check your connection (RRNS)
- Check your website (ANS)
- Encourage participation by providing badges





stop DDOS attacks.org

Save the web...

### Test your website

Check!

Test here if your (or any other) website is using a name server that can be used to execute DDoS attacks. DDoS attacks are paralyzing the

internet infrastructure that is so crucial for our modern society. Be responsible and use a reliable name server!

### Test your connection

Check it now!

Test here if the name server that you are currently using to visit this website can be used to execute DDoS attacks.

## DDoS attacks

The goal of executing DDoS attacks is making a network resource unavailable. A common way to do so is by sending unsolicited messages to the victim. One message won't hurt the receiver. Thousands or even millions of messages can introduce downtime of the victim. This can be compared to sending unsolicited bricks.



Receiving one brick is annoying, receiving thousands of bricks will make sure that the mailman is not able to access our mailbox for legitimate mails anymore. It is, however, quite hard to send so many bricks.

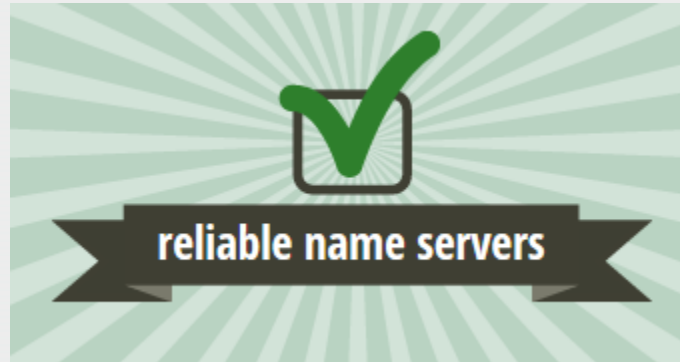
We can only carry a couple of them to the post office. When we are able to trigger a lot of people to send bricks to one address at the same time, we have a method to make the victim unreachable.

The Problem

It gets bigger

# Reliablenameservers.org

- Check you website
- Corporate and “green” feeling
- Encourage participation by providing back-links





**Problem solved, any questions?**