Tales of the unexpected -
handling unusual DNS client behaviour

Netnod, spring 2015 – Cathy Almond, ISC

What is this talk about?

- Random DNS query attacks against specific domains
- Impact on Recursive Server operators (“collateral damage”)
- Mitigation approaches

The attack

- first seen in 2009
- reappeared during 2014
- attack is directed at DDOSing DNS authoritative provider, but incidentally degrades ISP resolvers in the path

The parties involved

- Sometimes this is an extortion attack
- Frequently seems to originate and terminate in China

- Target domain may be hosted with many non-targeted domains
- Targets hop from provider to provider

Initiator of DDoS traffic

Target of the DDOS Authoritative provider

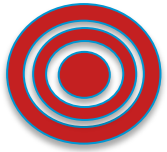
Identifying the attack

high volume of queries for non-existant sub-domains

<randomstring>.www.example.com
<anotherstring>.www.example.com

does not exist

exists



Attack begins

Insecure Home gateways

Home users are unaware

1. Requests for randomstring.www.example.com

nothing about this in the cache

ISP resolvers

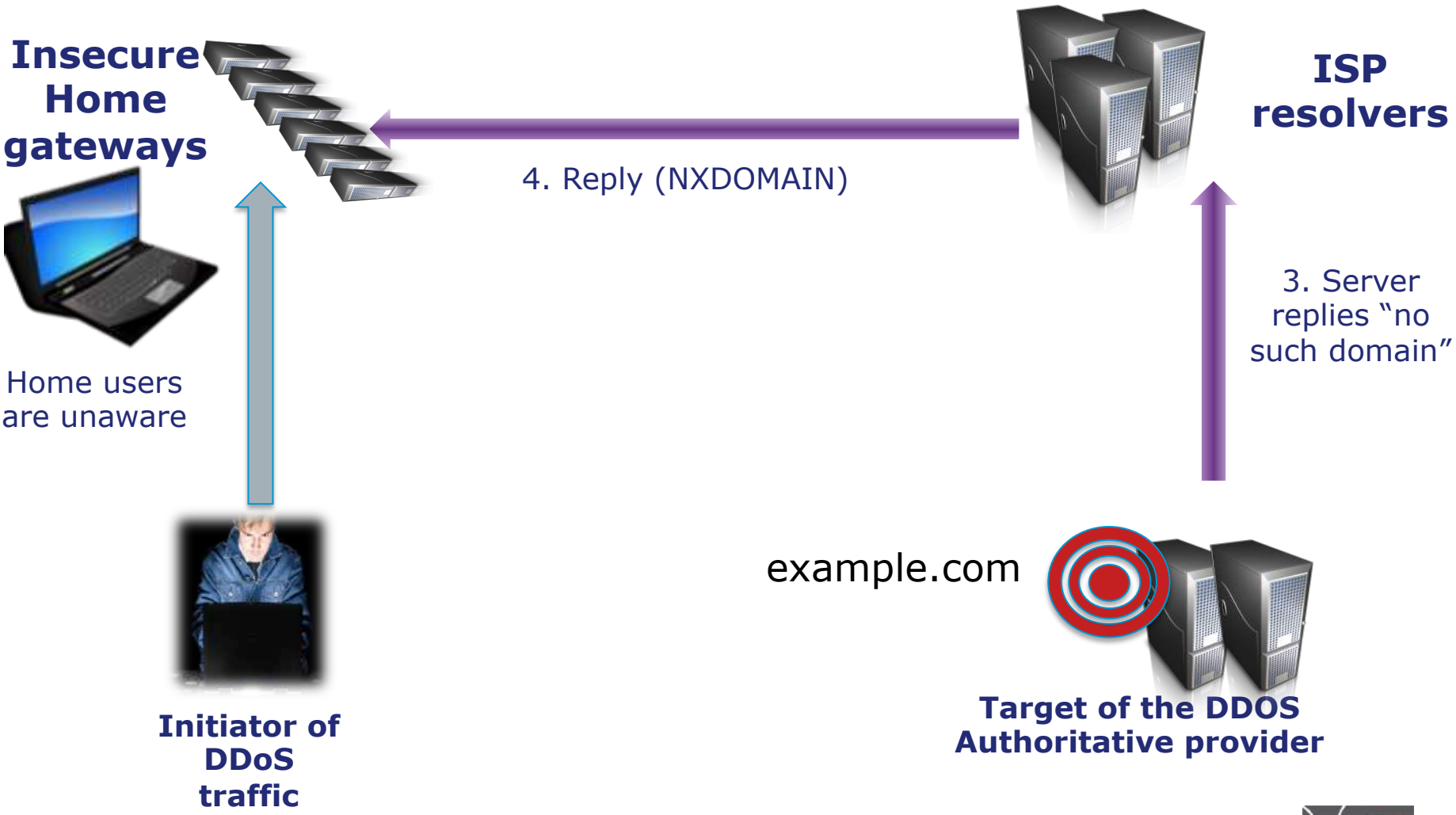
2. Attempt to resolve

Initiator of DDoS traffic

example.com

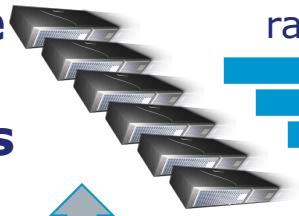
Target of the DDOS Authoritative provider

Initially, the target responds



More requests flood in

Insecure Home gateways



1. Requests for randomstring2.www.example.com

ISP resolvers

Home users are unaware

Initiator of DDoS traffic

example.com

Target of the DDOS Authoritative provider

Target is overwhelmed

Insecure Home gateways

Home users are unaware

Initiator of DDoS traffic

ISP resolvers

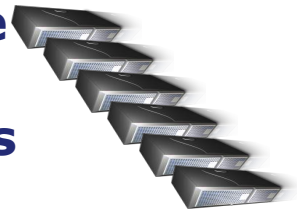
2. Attempt to resolve

3. Server is unresponsive

example.com

Target of the DDOS Authoritative provider

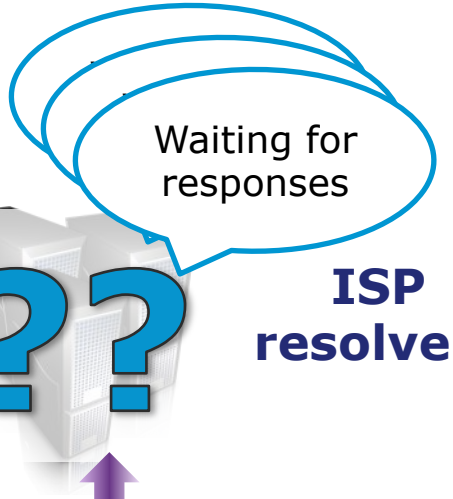
Resolver is degraded



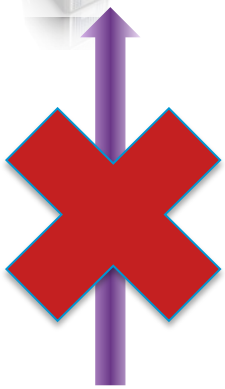
Insecure Home gateways

Home users are unaware

Initiator of DDoS traffic



ISP resolvers



3. Server is unresponsive

example.com

Target of the DDOS Authoritative provider

Legitimate queries fail

Stage 1: “hair on fire”

Accurate diagnosis - symptoms

- Increased inbound client query traffic
- Increased outbound NXDOMAIN and SERVFAIL responses
- Resolution delays to clients
- Dropped responses
- Increased memory consumption
- Firewall connection table overflows

Accurate diagnosis - evidence

- Backlog of recursive client queries
  - which queries are in the backlog?
  - is there a pattern?
  - originating from few or many clients?
- Open outbound sockets
  - to which servers; is there a pattern?
- Query logging / query-errors logging
- Network packet traces

“Do”s…

- Eliminate open resolvers
  - is yours an open resolver?
  - open client CPE devices
  - small business users forwarding local open caches to your servers
- Investigate compromised/infected clients
  - ‘hearsay’ evidence that these exist now
  - but it’s only a matter of time…

And “don’t”s…

- Panic!!
- Assume that increasing server resources (e.g. recursive client contexts, sockets, network buffers etc..) is going to help
- Block your clients

MITIGATION TECHNIQUES

What can we do?

What has been tried in production?

LIE
if necessary

Create a local answer

- Make recursive server temporarily authoritative for the target domain
  - Local zone
  - DNS-RPZ (*qname-wait-recurse yes;)

- *Manual configuration change*
- *Need to undo the mitigation afterwards*

Stage 2: Automate filtering

(Near) Real Time Block Lists

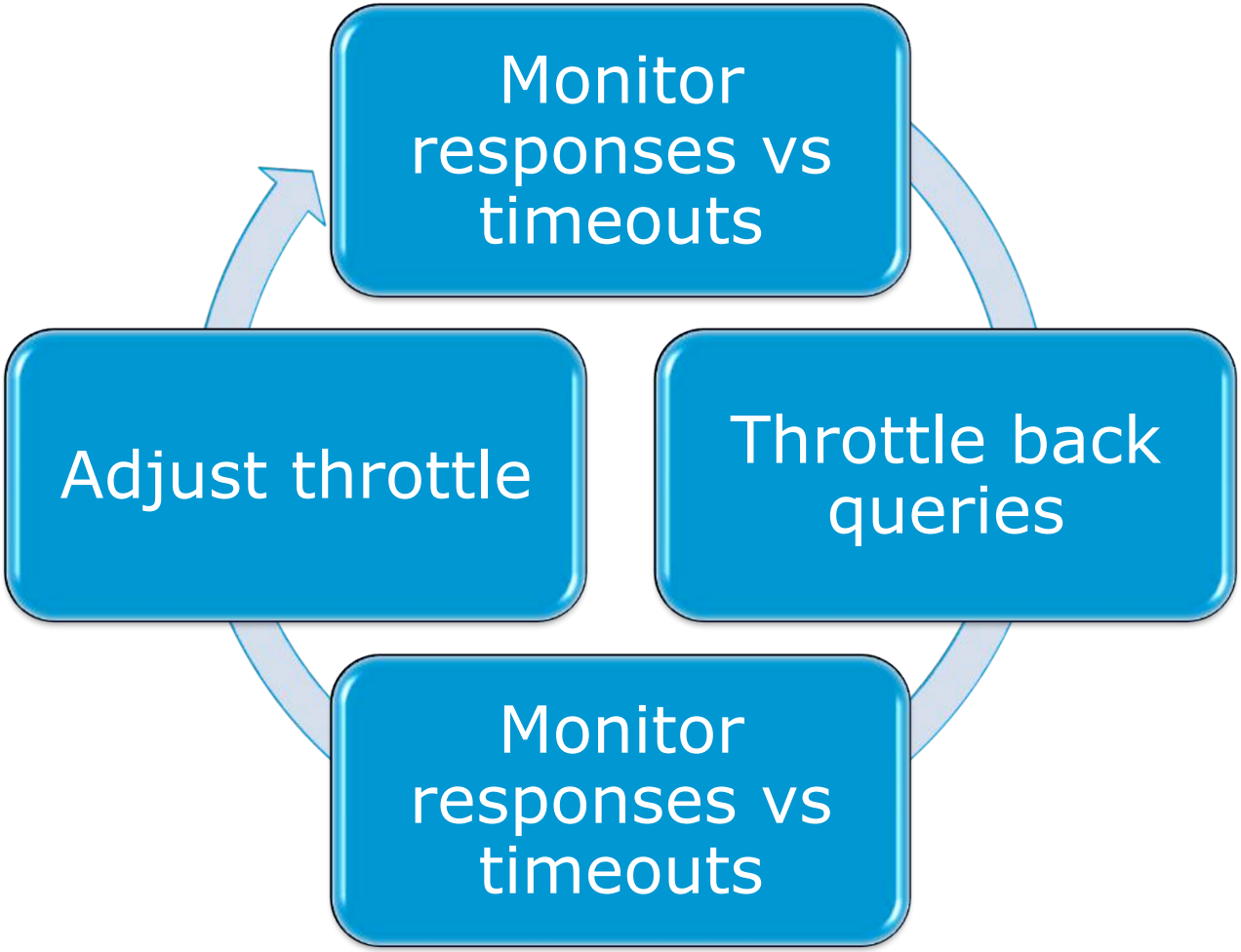
- Detect ‘bad’ domain names or just the problematic queries & filter them at ingress to the resolver
- Local auto-detection scripts
- Nominum Vantio
- BIND DNS-RPZ
- Costs associated with feeds
- Potential false-positives

Stage 3: Tune your resolver

PER ZONE

PER SERVER

Fetches-per-server



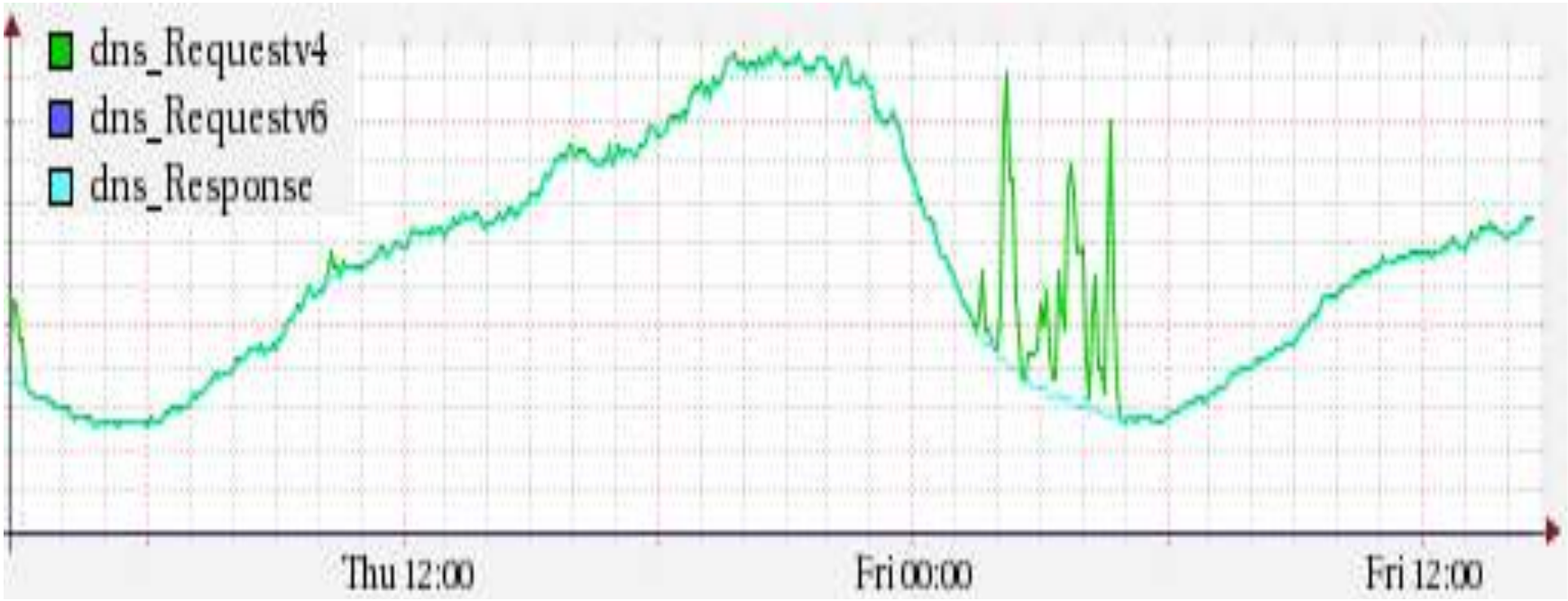
fetches-per-server

- Per-server quota dynamically re-sizes itself based on the **ratio of timeouts to successful responses**
- Completely non-responsive server eventually scales down to fetches quota of 2% of configured limit.
- Similar (loosely) in principle to what NLnet Labs is doing in Unbound

fetches-per-zone

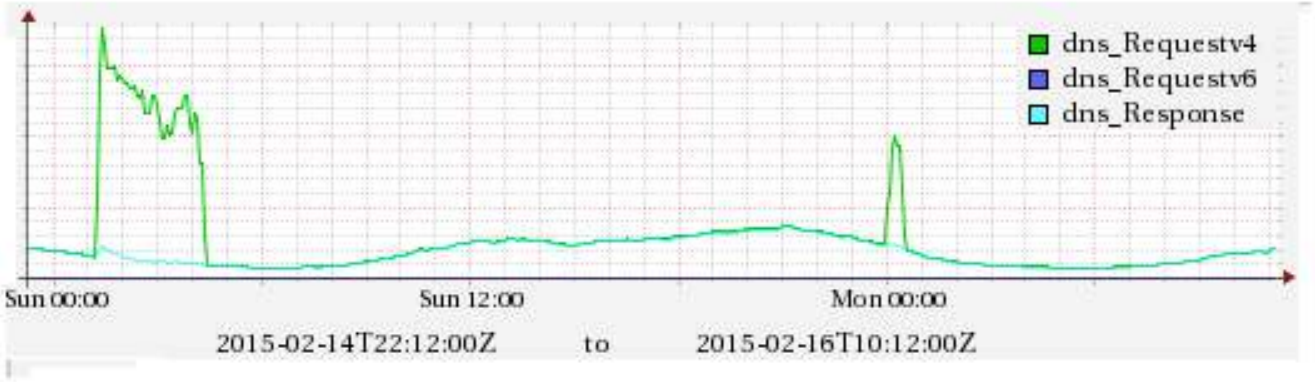
- Works with unique clients
- Default 0 (no limit enforced)
- Tune larger/smaller depending on normal QPS to avoid impact on popular domains

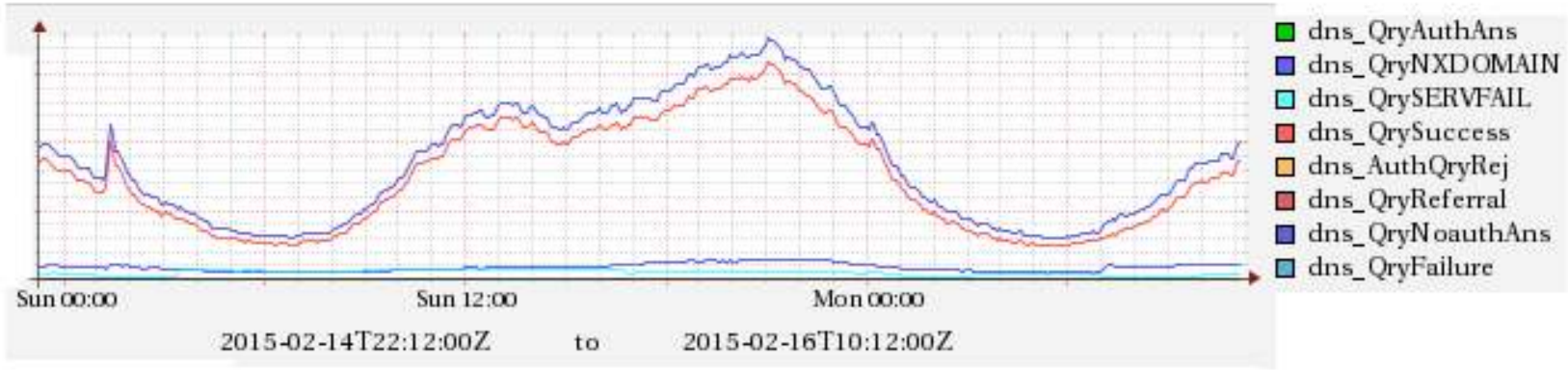
fetches-per-zone at Jazztel



Spanish triple-play ADSL carrier & ISP
Roberto Rodriguez Navio, Jazztel Networking Engineering
used with permission

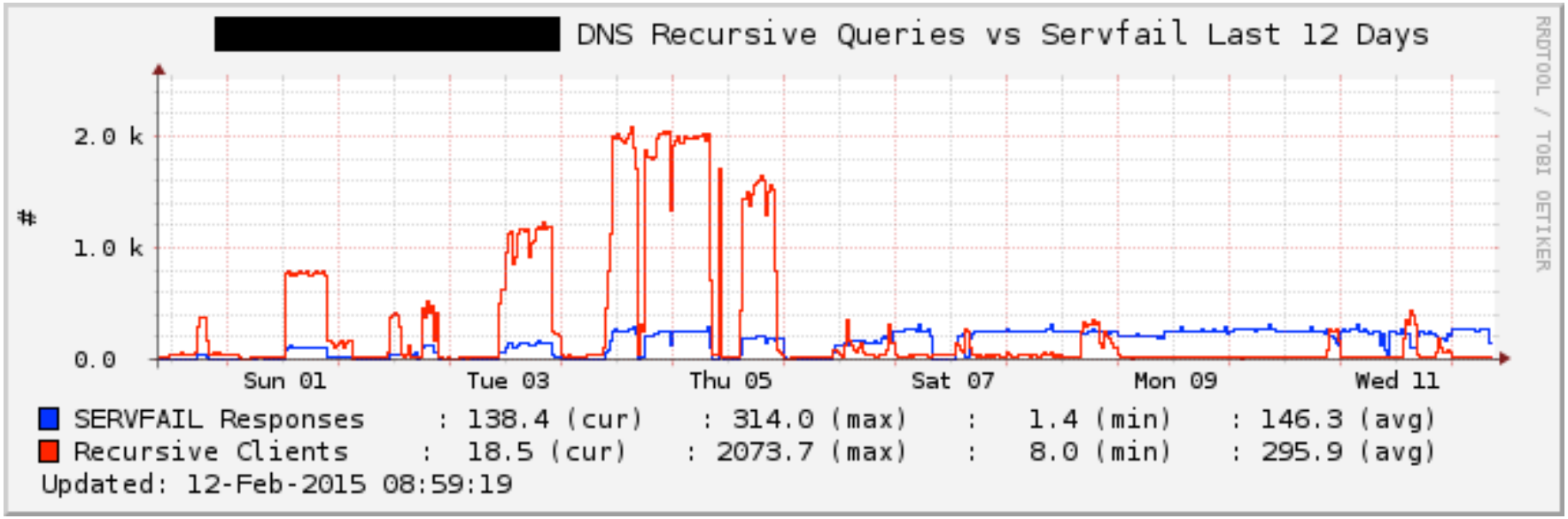
More on fetches per zone



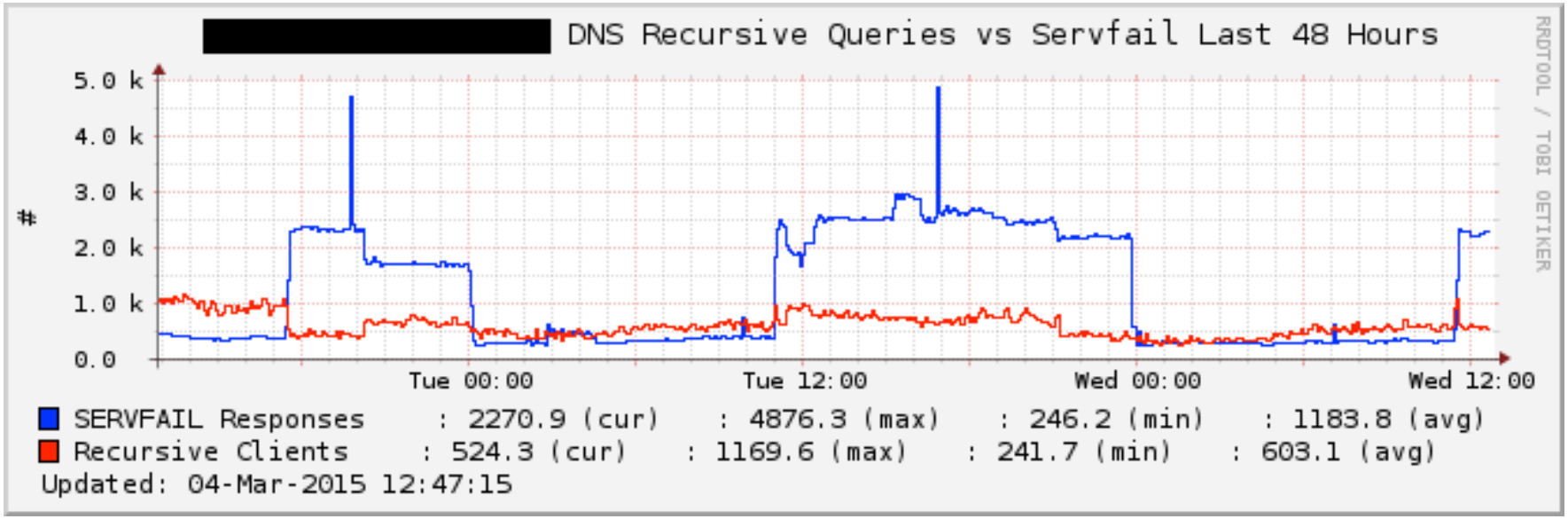


Spanish triple-play ADSL carrier & ISP
Roberto Rodriguez Navio, Jazztel Networking Engineering
used with permission

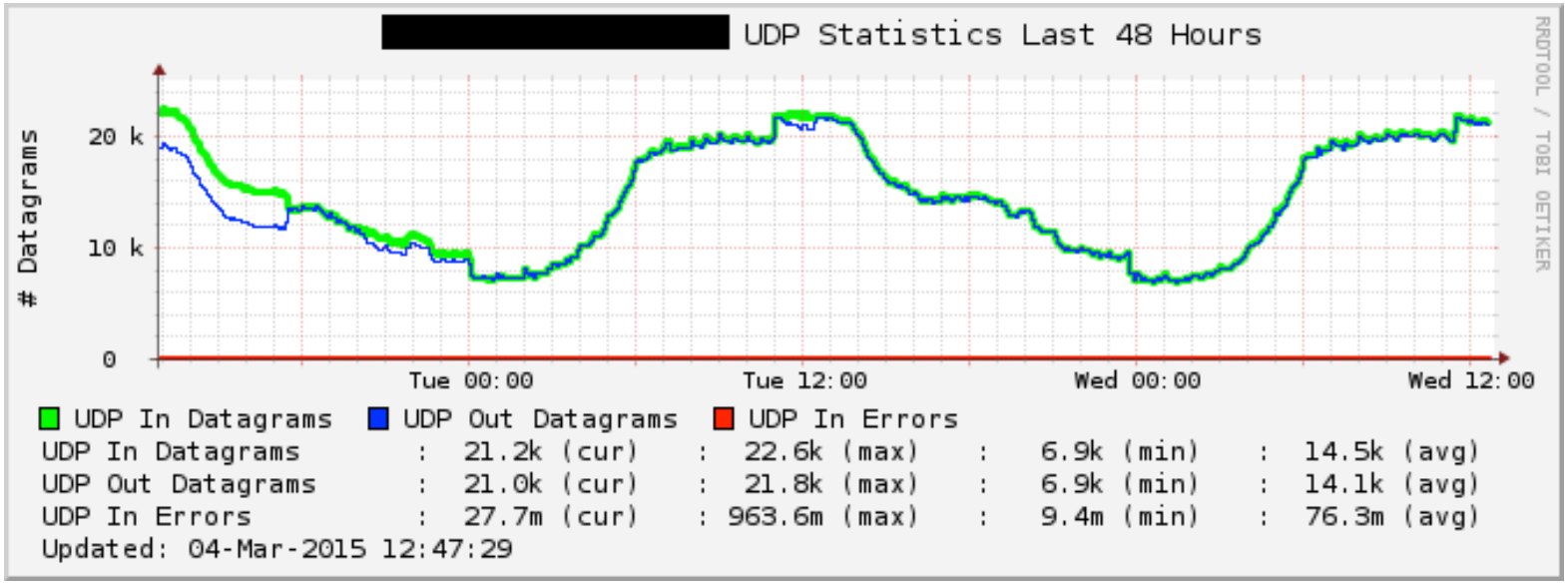
fetches-per-server



fetches-per-server



fetches-per-zone v. fetches-per-server



What will the user see?

- Situation normal – no change to their usual experience (for most)
- (Some) SERVFAIL responses to names in zones that are also served by under-attack authoritative servers (collateral damage)
- NXDOMAIN responses for names in legitimate zones for which we ‘lie’

Still experimental…

- Some controversy about adaptive approach vs blacklists
- Whitelists may be needed
- Per-server/zone settings
  - *Configurable override parameters for fetch limits on a per zone or per server basis*
- SERVFAIL cache (for client retries)
- Improved reporting & statistics

Summary

1) Configure your resolver to **LIE**
   answer authoritatively yourself
2) Configure a **BLACK LIST** of domains under attack
   possibly subscribe to a feed for this
3) Consider **ADAPTIVE QUOTAS**
   per server
   per zone

(Good feedback on these from many sources)

ISC

Ideally, close the open resolvers!!

GOOD LUCK!

bind-suggest@isc.org, cathya@isc.org