Some thoughts about

A Speculation on DNS DDOS

for

Geoff Huston
APNIC

What we know…

Well – guess - from the snippets that have been released…

It was a Mirai attack
It used a compromised device collection
It used a range of attack vectors
    TCP SYN, TCP ACK, GRE, …
    One of these was DNS

DDOS Attacks

- Are nothing new – unfortunately
- And our response is often responding like for like
  - Build bunkers of bandwidth and processing capacity that can absorb the attacks
  - And leave the undefended open space as toxic wasteland!
- But using the DNS for attacks opens some new possibilities…

What we **guess** about the DNS attack

The attack queries looked like queries that are often seen at authoritative name servers

- So front end pattern matching and filtering may not work

It loaded the authoritative servers with legitimate queries

- So its likely it was <random>.target queries to defeat caches and simple filters
- Just like Chrome!

The victim set

- Authoritative DNS name servers for the attacked domain names
- Because the <random> name form will defeat the recursive resolver caching function and the query is passed to the auth name server to resolve
- The result is that the name will fade out once recursive resolvers' cache entries expire, and they cannot refresh

Possible Mitigations – 1

”A Bigger Gun”

- More *Foo*
  - Add more authoritative name servers
  - Add more bandwidth to authoritative name servers
  - Add more CPU and memory to authoritative name servers

- i.e. more ”foo” and try and absorb the attack at the authoritative name server infrastructure

Possible Mitigations - 2

Longer TTLs:

- Low TTL's make you more vulnerable because recursives need to refer to authoritatives more frequently
- With a longer TTL, the attack will still happen, but the legit recursives may not get a cache expiry so quickly
- The recursive resolvers will still serve cached names from their cache even when the authoritative name server is offline
- Attackers will need to attack for longer intervals to cause widespread visible damage

- But..
  - Nobody likes to cement their DNS with long TTLs
  - And recursive resolvers won't honor longer TTLs anyway!

Possible Mitigations – 3

Filter queries:

- Try to get a fix on the <random> name component in the queries
- Set of a front end query filter and block these

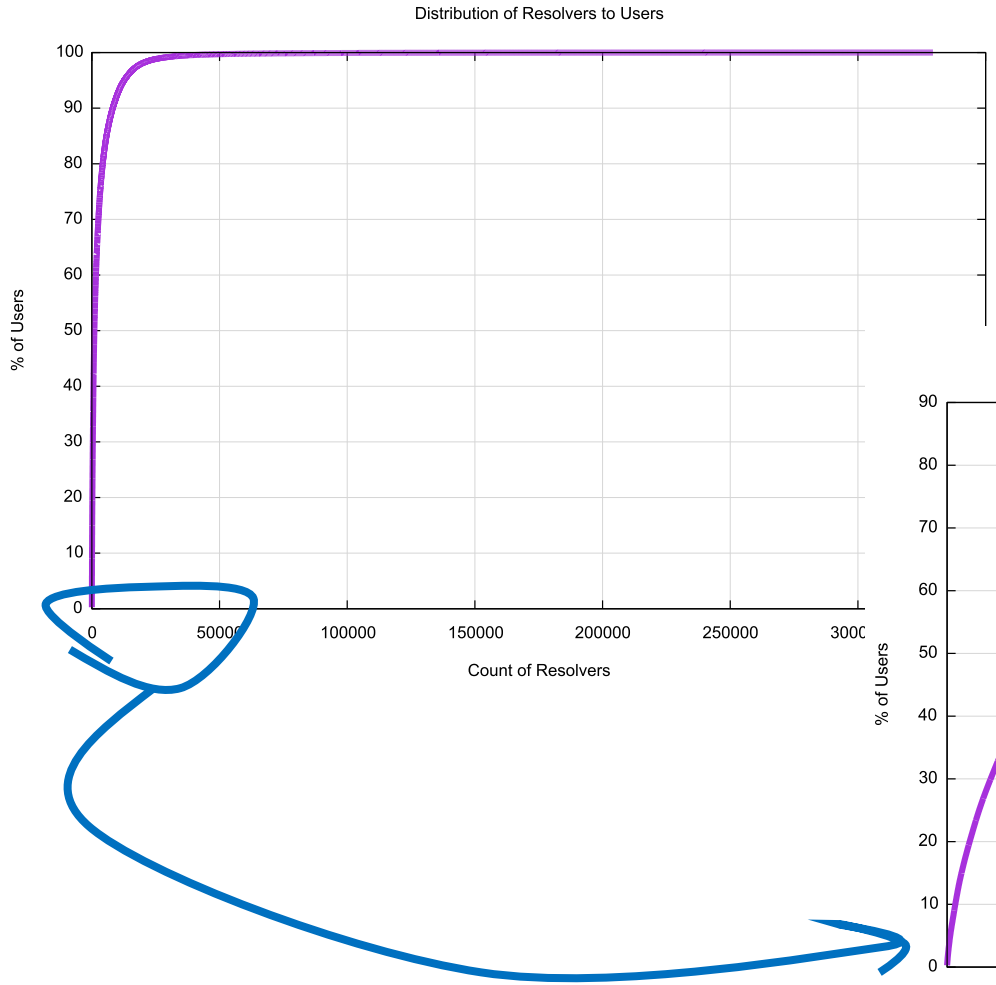
- But
  - This is just tail chasing!

Possible Mitigations - 4

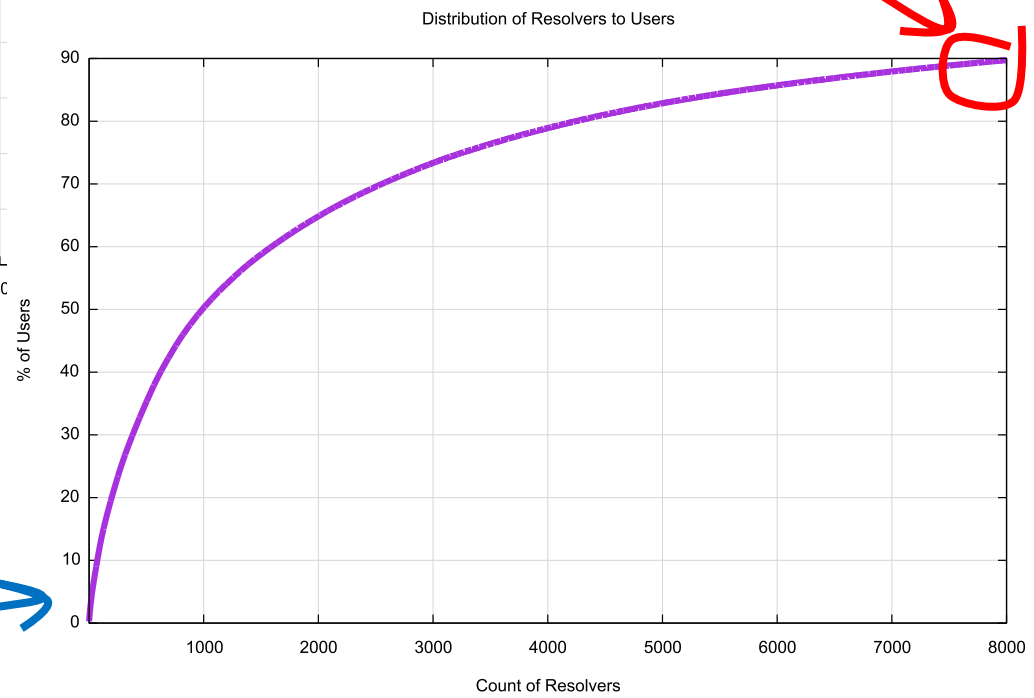
What if the devices are passing the queries directly to the authoritative name servers?

Filter IP addresses!

All resolvers might be equal, but some resolvers are more equal than others!



8,000 distinct iP addresses (2.3÷ of all seen iP addrs) for resolvers serve 90÷ of all experiments



Possible Mitigations - 4

“Filter Filter Filter” (IP sources)

Only 8,000 discrete IP addresses account for more than 90% of the users’ DNS queries

These are the main recursive resolvers used by most of the internet – answer them!

Put all other source IP address queries on a lower priority resolution path within the authoritative name server

Divide queriers into “Friends” and “Strangers”: Just like SMTP!

Possible Mitigations - 5

What if the devices are passing the queries via recursive resolvers?

Possible Mitigations - 5

Get assistance!

(Yes, I'm dreaming, but it's a Good Dream!)

Use DNSSEC and apply *NSEC Aggressive caching*

- The attack will generate NXDOMAIN answers
- So why not get the recursive resolvers closer to the individual devices to answer the NXDOMAIN query directly
- This can be done with the combination of DNSSEC and NSEC signing, using the NSEC span response to then respond to further queries within the span without reference to the authoritative servers
- This means that the recursive system absorbs the attack and does not refer the queries back to the authoritatives

If only…

- Piecemeal solutions deployed in a piecemeal fashion will see attackers pick off the vulnerable again and again
- And the long term answer is not bigger walls, as the IoT volumes will always be higher
- We need to think again how to leverage the existing DNS resolution infrastructure to be more resilient
- And for that we probably need to talk about this openly and constructively and see if we can be smarter and make a more resilient DNS infrastructure
- And for that we probably need to talk about the DNS and DNSSEC and how it works, and how it can work for us