Routing Security and DNS Services

Geoff Huston

APNIC

The Internet's routing system works by rumour

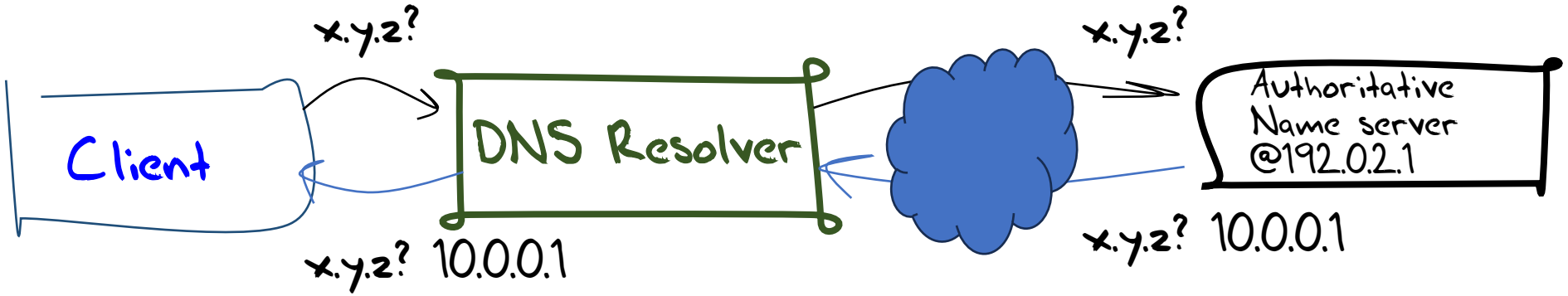
"I tell you what I know, and you tell your mates, and your mates tell their mates, and …"

But what if I tell you a lie?

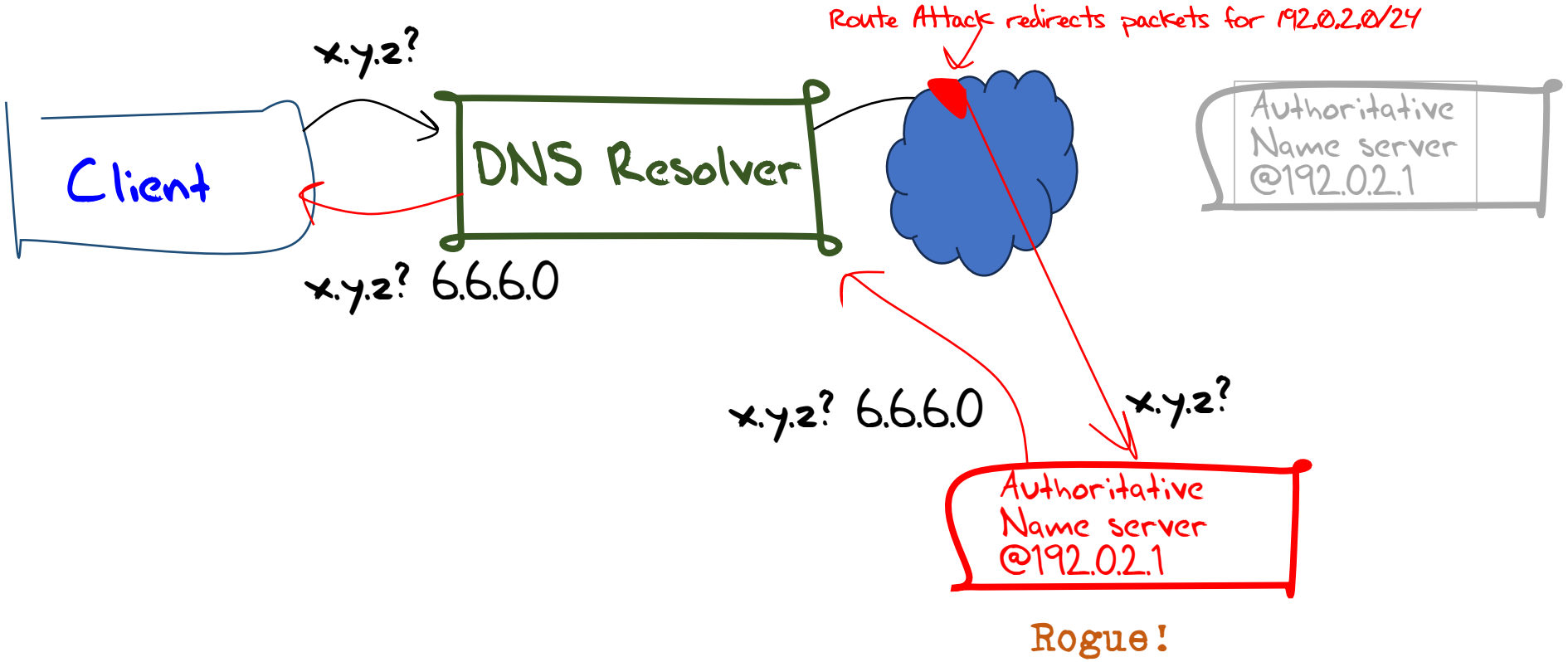
Or someone maliciously alters what I'm trying to tell you?

- Then the altered routing information will propagate throughout the routing system

So what?


x.y.z?
Client
DNS Resolver
x.y.z?
Authoritative
Name server
@192.0.2.1
x.y.z? 10.0.0.1
x.y.z? 10.0.0.1

So what?


Route Attack redirects packets for 192.0.2.0/24
x.y.z?
Client
DNS Resolver
Authoritative
Name server
@192.0.2.1
x.y.z? 6.6.6.0
x.y.z? 6.6.6.0
x.y.z?
Authoritative
Name server
@192.0.2.1
Rogue!

That will never happen…

- Yeah, except when it does!
  - MyEtherwallet attack used BGP hijacking to inject more specific routes for Amazon's Route53 DNS service, redirecting queries to the attackers DNS server.
  - The rogue server answered SERVFAIL for all queries EXCEPT MyEtherWallet
  - The entire attack was over in under 2 hours!

Don’t believe everything the DNS tells you

- The DNS operates under the assumption that if you direct your DNS query to the “right” IP address then you can take any response you get as authentic.
- This is an extremely foolish assumption!
  - The problem is that DNS clients have no knowledge of where the DNS response came from, let alone validate the authenticity of either the server that generated the response or the authenticity of the contents of the DNS response
  - So a routing attack can substitute one DNS server in place of another with implicit redirection of queries with no external indication

So, let’s use DNSSEC everywhere!

- If we can’t stop these substitution attacks, then why not harden the DNS to detect rogue responses?
- If every DNS zone was DNSSEC-signed, and every client validated the DNSSEC-signed responses that they receive, then injecting false information into the DNS would be more challenging

- But we all don’t do that, because <reasons>

If not DNSSEC, then…?

- Points of service in the DNS (recursive resolvers and authoritative servers need to take steps to protect their integrity
  - We could follow the lead from the Web environment and push the DNS to use DNS over TLS/QUIC/HTTPS everywhere
    - Which pushes us into a different world of dependent trust on the integrity of the Web PKI in authenticating the server's identity by name
    - And of course this also has the large scale load impost of shifting DNS query/response traffic from DNS over UDP to DNS over secure reliable transport
    - So its an expensive option!
    - And that makes it highly unlikely as a general "solution"
  - Or we could try and make routing attacks on DNS infrastructure a little harder to pull off!

Countering routing attacks

- If we can’t stop routing attacks from taking place, then we need to help networks to detect false routing information before they propagate it further

Route Registries can help

Route Registries are used to record a set of route advertisements with a network

- Other networks can use this as a filter set, and reject route updates that are not listed in the registry
- Route registries can also contain route policies, which can be used to detect route updates that are counter to the announced policies

Route Registries can help **a bit**

BUT

- Route registries are incomplete, contradictory, often out of date, come with poor authority models and often lack robust authenticity capabilities
- So network operators are left in a quandary whether to trust route registry data when they are unsure of the authenticity and currency of the registry data

Routing Security with the RPKI

- Let's invert the approach and start with an authority model
  - Use the number registry framework to support a robust association of addresses and ASNs with the holder's public/private key pair
    - "I control the address 203.10.60.0/24", signed by the private key of the address holder
- And then generate simplified forms of route information that are signed using keys that are authenticated in this RPKI
  - "I allow AS131074 to originate a BGP route for 203.10.60.0/24", signed by the private key of the address holder
- Collect these routing authorities (ROAs) validate them, and use them as a filter across received BGP updates
  - This is equally effective in unicast and anycast service models

This has one interesting property

- Not everyone has to perform route origination validation
- It’s the transit networks that are crucial here
  - And only 15% of networks offer transit services to other networks.
  - And if these transit networks dropped updates that did not match origination information then propagating a false route that did have the “right” originating network then the false routing information would stay within a very limited locale

Close, but just not close enough

- ROAs can be used to limit the injection of routing information
- But if you can manipulate the AS Path then you can still inject false routing information
- The IETF proposed a more complete routing framework (BGPSEC) but the barriers to universal deployment appear to be formidable

  (i.e. it's just not going to happen)
- The IETF is working on a lightweight approach that attempts to detect certain types of AS Path manipulation
  - But the effort is taking years, and its not clear yet how effective it can be

That’s great! But why is this my problem?

- DNS authoritative servers and recursive resolvers tend to operate as “promiscuous” servers
  - Their role is to answer all queries
  - Aside from local policy limitations, limiting who can ask the server does not make all thjat much sense
- So why should a DNS infrastructure operator be especially concerned about the integrity of the routing system?

It’s all about me!

- Without DNSSEC, the DNS relies on a much more primitive model of integrity:
  - If a send my query to the “right” IP address then I can trust the response that I get
- So if an attacker can misrepresent itself as the “right” server by attacking the routing system, then all clients may be mislead
- So I want to publish enough information into the routing system that will allow all others to detect when false routing information is being propagated about the location of my servers

“I want to help prevent others learning routing lies about me!”

What should I do?

- Option A: Do nothing!
  - It’s cheap!
  - And its not totally crazy!

    Applications should know by now that IP addresses are untrustable, and connections to a remote service should at a minimum use TLS server authentication to assure the client that they are connecting to the “correct” service by service name

    And if applications choose not to authenticate the remote server they are communicating with, then they are taking an insanely naive view of the Internet’s integrity!

    And who knows? Maybe DNSSEC will come into fashion in a decade or two from now! ;-)

What should I do?

- Option B: Publish the essentials
  - Obtain RPKI certificates, publish ROAs and publish route registry entries
  - Publish origination information in a route registry
  - Don’t filter your own BGP sessions
  - The problem…

    Is in keeping this published information up to date. Letting this lapse and fall out of date causes random reachability issues for others that are often challenging to resolver
  - Why do this?

    Because it’s a minimal response to help others learning routing falsehoods about your service

    It looks like you care!
  - But don’t fall into the trap of thinking that this is a complete “solution” – because its not!

What should I do?

- Option C: The full box of bananas!
  - Obtain RPKI certificates, publish ROAs and publish route registry entries
  - Publish origination information in a route registry
  - Filter your own BGP sessions and discard incoming routes that fail these authenticity checks
  - Really?

    That last step does not help your remote clients and stands the risk of increased brittleness in your DNS service

    It adds operational cost and complexity to your service without obvious benefits to your client base

There are no absolutes here

- There is no absolute “solution” to routing security or DNS integrity
- There are measures that increase the challenge to an attacker, but they come with increased cost to the service providers and their users
- Doing nothing does not seem like a satisfying option
- But trying to solve the entire problem space involves more than we know how to do
- So I’d suggest Option B seems like a prudent path at the moment!

Thanks!