Impact of DNS Protocol Developments on Enterprise Networks

Jim Reid
RTFM llp
jim@rfc1035.com

```
#include <std_disclaimer.h>
```

The IETF

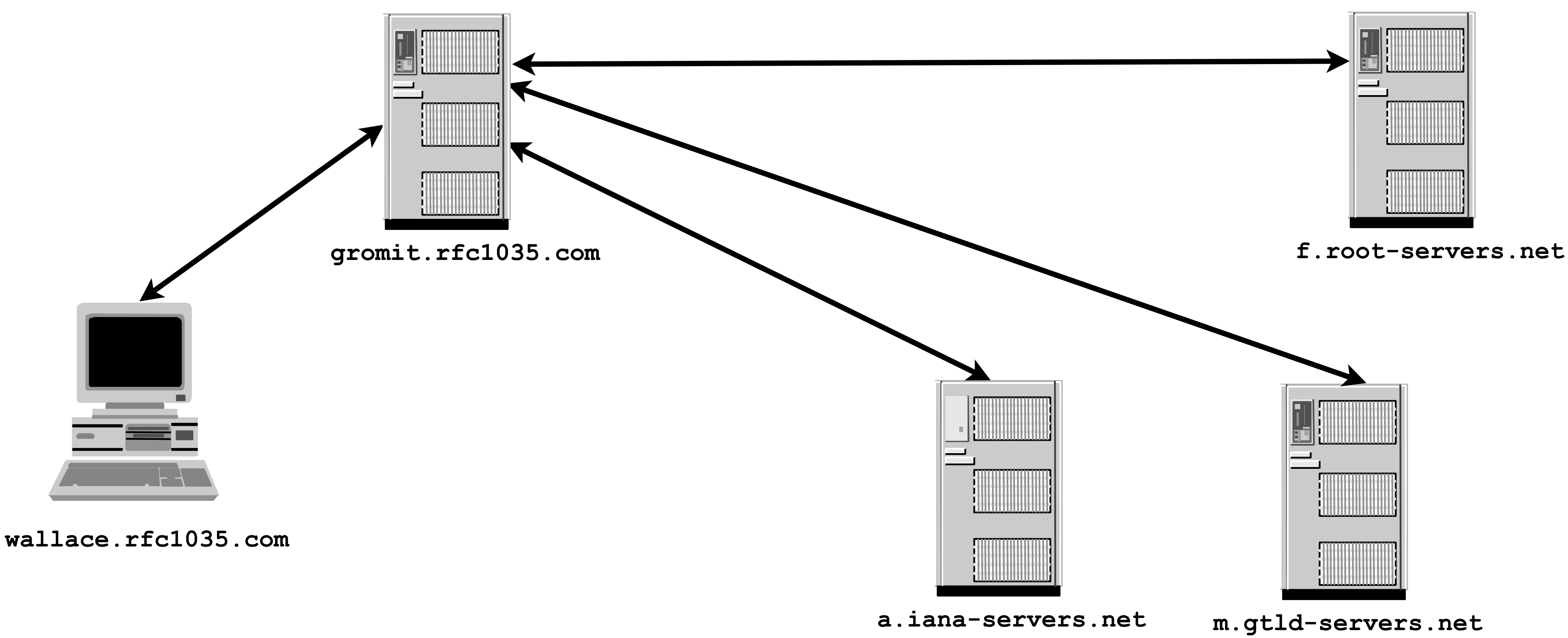
- Internet Engineering Task Force
  - No legal identity (by design)
- Develops almost all Internet protocol standards:
  - Routing, addressing, naming, etc.
- Self-organising into Working Groups
  - No membership criteria or voting
  - Decisions made by consensus on mailing lists
    - “rough consensus and running code”
  - WGs define a problem, find a solution and then disband

DNS at the IETF

- Several DNS-related working groups:
  - DNSOP - DNS operations
  - DPRIVE - DNS Privacy (DNS over (D)TLS)
  - DOH - DNS over HTTP(S)
- Now closed WGs:
  - DNSEXT - DNS Extensions (Secure DNS)
  - DANE - DNS-Based Authentication of Named Entities

A Typical DNS Lookup

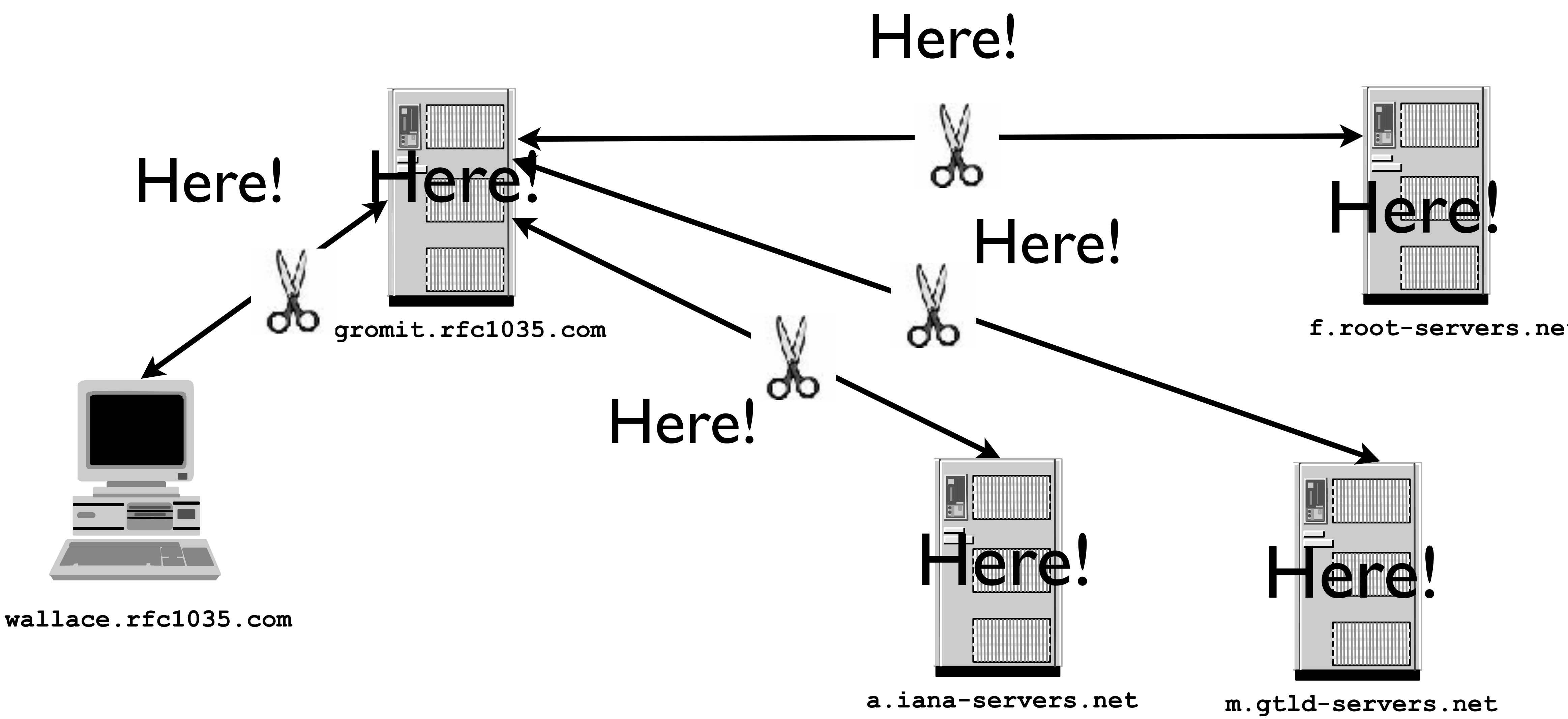
Resolving server **`gromit`** returns **`www.example.com`**'s address to the client **`wallace`**'s stub resolver, which has been patiently waiting for an answer to the DNS query it made



What's Wrong With That?

- Nothing: it all works just fine.....
- BUT there's no authentication at all!
- A client can't tell:
  - Where an answer **really** came from
  - If the server that replied is telling the truth or not
  - If it received **exactly** what the server sent
    - This applies to **`wallace.rfc1035.com`**'s query and the lookups **`gromit.rfc1035.com`** performed to resolve that query

So where are the vulnerabilities?



DNS Attack Vectors

- Bombard client or resolving server with forged answers or educated guesses
- Intercept a response packet and modify it
  - Tends to only work well if adjacent to client or server
- Inject bogus data into caches
- Take control of the name server(s) for some zone and make them tell lies
- Compromise the registry
- Evil routing/peering tricks to hi-jack traffic

The Solution: DNSSEC

- Weaknesses have been known for a **long** time
- IETF started work on DNS Security in late 1990s
  - DNS Security Extensions (DNSSEC)
- Design goals:
  - Authenticity and verification of DNS data
- Design exclusions:
  - Message authenticity/verification
  - Confidentiality & privacy
  - Server authenticity/verification

DNSSEC in a Nutshell

- Strong cryptographic hashes of DNS data
  - SHA-1, SHA-2
- Public-key crypto
  - RSA, DSA, ECDSA, Diffie-Hellman
- Digital signatures of hashes of DNS data
  - Signed with DNS zone's private key
- Signatures and public keys stored in the DNS as resource records

Validation

- Validating resolver computes hash value of the returned DNS data that it requested
  - Response also includes the signature for that data
- Validator retrieves the corresponding public key and applies that public key to the signature to get the hash value that had been signed
  - If that hash matches the one it calculated itself, all is well
  - If not, Something Bad has happened

DNSSEC Deployment - 1

- Swedish ccTLD `.se` was first, September 2005
- Internet root got signed July 15th, 2010
  - A very, very cautious roll-out for obvious reasons
    - Awkward political problems too
    - No one organisation has the "master key"
- Most of the popular TLDs are now signed
  - `.com`, `.net`, `.uk`, `.info`, `.org`, `.de`, etc.
- All of ICANN's new gTLDs must use DNSSEC

DNSSEC Deployment - 2

- Very little adoption or interest
- Only 2 of the top 100 Alexa websites have signed domains
- Survey found uptake in `.com` was < 1% and ~30% of them had DNSSEC setups that failed to validate
- ~12% of DNS queries use a validating resolver
  - Most of them come via google's 8.8.8.8 and Comcast
- Some ccTLDs have got most delegations signed but almost none of the nation's ISPs validate

Catch 22

- Why incur the cost and hassle of signing if nobody is validating?
- Why incur the cost and hassle of validating if nobody is signing?
- Where are the use cases and killer apps?
  - Nobody's seriously developing these
  - Some proof of concept browser plugins
- Probably need all three groups to act in concert at the same time
  - Good luck with that…

Externalities

- Signers get no benefit from doing that, validators do
  - If the organisations doing validation screw up, signed zones fall off the net
- Anyone doing DNSSEC validation loses out if/when those who are signing make a mistake
  - ISP A loses when validation fails for ***`important.com`*** while there's no problem at ISP B which does not validate
- Why take the risk?
- DNSSEC adopters take on risks and costs for no real gains for themselves, just for others

DNSSEC in Enterprises

- No killer app yet
- No convincing use cases or business justification
  - Serious DNS spoofing attack might change minds
- Why add the complexity and risks for very little benefit?
- DNSSEC can interfere with on-the-fly DNS response rewriting systems
  - Blocking access to malware & smut, load balancers, geo-specific redirection, high availability middleboxes, etc.

Key Rollover in Pictures

Key Rollover

- DNSSEC keys will need to be changed from time to time
  - Sensible cryptographic practice
- This should happen at regular, planned intervals
  - Might have to happen sooner in an emergency
- How is this best done?
- Principle is clear enough, doing it right isn't
  - Too many easily broken moving parts
  - A "one size fits all" approach is impossible

The DNSSEC Treadmill

- DNS admins need to re-sign their zones and keep doing that forever
  - They need to change keys regularly too
- Need to use latest DNS software:
  - Bug fixes, new crypto support, add/drop algorithms, etc.
- Lots of last mile issues
- Open-ended and hard to quantify costs for support, operations, troubleshooting and tooling
  - Few organisations know what DNS costs them anyway

DNSSEC: A Never Ending Task?

DPRIVE - DNS Privacy

- WG set up as a result of Snowden revelations
- Initially aimed at DNS traffic between stub resolvers and resolving servers
  - About to consider resolving server traffic with authoritative servers
- Conceptually simple: DNS over (D)TLS
  - (Datagram) Transport Layer Security
  - Encrypted traffic uses port 853 rather than port 53

DPRIVE & Enterprise Networks - 1

- DNS traffic goes dark (sort of)
  - No visibility of what's in port 853 traffic
  - Can't intercept or eavesdrop on that
  - Obvious implications for DNS rewriting and blocking systems
- Not such a Big Deal for enterprise nets
  - Resolving DPRIVE server decrypts incoming queries (and logs them?) before making plaintext queries to authoritative servers

DPRIVE & Enterprise Networks - 2

- Enterprise IT management remains in control
- DNS over (D)TLS unlikely to be enabled by default
  - Conscious decision needed to switch this on
- Can check for port 853 traffic in the network
  - Tripwire(s) at firewalls and DMZ?
- Little client software so far
  - No killer app or use cases yet

DPRIVE Server-side Implementations

- Native support in two open-source resolving servers, **`unbound`** and **`knot`**
- No current plans to support this in BIND9
- Handful of experimental public servers - mostly for testing - on volunteer, best efforts basis
- Quad9 started in Q4 2017
  - Global and free anycast resolver service from PCH
    - Similar to 8.8.8.8, but on address 9.9.9.9
  - Offers service on port 53 and 853 (DNS over (D)TLS)

DPRIVE Client-side Implementations

- Only one: **`stubby`**
  - DNS proxy which takes incoming queries on loopback interface and forwards them using (D)TLS to port 853 somewhere
  - Currently uses (D)TLS1.2 - will work with (D)TLS1.3
  - Mostly aimed at experts
- Proof of concept app in Android development builds
  - Might move to production builds in Q3/4 2018
  - No decisions yet

DPRIVE Status

- Very little deployment and usage so far
- Quad9's only seen 5-10,000 unique IP addresses use DNS over (D)TLS
- **`stubby`** developers estimate a broadly similar number of downloads
- DPRIVE enthusiasts hope mobile apps will drive uptake
- Uncertain future because of other IETF work
  - DPRIVE may be overtaken by events
  - Could end up as the DNS equivalent of ToR

DNS over HTTP(S) - DOH

- WG formed last year: first meeting at IETF100
- Simple idea
  - Browsers send their DNS queries over HTTP(S) to a web server, web server does the resolution or gets a resolving DNS server to do that
  - Web server could "push" DNS data to browser to reduce latency and improve page load times
- Current thinking is this will be for HTTP/2
  - HTTP1.1 without TLS is possible, but should be discouraged

DOH Challenges & Issues

- HTTP has richer set of primitives than DNS
  - How well can these be aligned? Should they?
- Interactions between browser and DNS caches
- Server discovery: how does a DOH-capable browser find a DOH-capable web server?
- Use cases and best practices will need to get documented eventually
  - No deployment (or standardisation) of DOH yet

DoH & Enterprise Networks - 1

- Much DNS traffic could go **really** dark
  - Most browser DNS traffic would be encrypted and use port 443 (HTTPS), not port 53 (DNS)
  - DoH activity will be “buried” inside HTTPS connections
    - Can’t intercept or eavesdrop on that
    - Hard to find out who’s looking up what and when
    - Web servers handle the DNS queries sent by browers
  - Obvious implications for DNS response rewriting and blocking systems

DOH & Enterprise Networks - 2

- Arbitrary web servers get DOH traffic instead of queries to locally-run resolving DNS servers
  - DNS logs and analytics less useful
  - Monitoring or intercepting port 53 traffic at the DMZ or firewall will be less effective
  - Web server's DNS policies apply, not the enterprise's
- Address-based rewriting of DNS responses would apply to web server, not the orginating browser
  - Local DNS access control policy effectively bypassed

DOH & Enterprise Networks - 3

- Enterprise IT management potentially loses control
  - No need to set up DPRIVE-style DNS servers
  - Users get DOH-capable browsers by stealth
    - Just upgrade to the latest version - job done!
  - Disabling DOH in local web servers might not help much
    - Could make a difference when web proxies have to be used to reach the public Internet

DOH Status

- Work at the IETF has barely started
  - First consensus document towards Q4 2018?
- Strong support from key players
  - google, Mozilla Foundation, Apache(?)
  - Should mean very quick and uncontrolled adoption
    - Just install latest Firefox/Chrome/whatever
- Significant overlap with DPRIVE
  - A different way to encrypt DNS traffic from stub resolvers
  - Which approach will win?

QUIC

- New transport-layer protocol with (D)TLS baked in
  - Most significant IETF development in over a decade
- Initial hopes for everything-over-QUIC have faded
  - IETF was too optimistic/ambitious despite lots of goodwill and engineering effort from key players
  - Immediate priority is HTTP/2, revisit a generic solution for other protocols (DNS, SIP, etc) later
    - Not clear when that might work start
- Too early to tell what will happen next and when

ACME & DANE

- ACME working group is considering DANE as a way of authenticating phone numbers and SIP addresses
  - Very strong pressure from US authorities and telcos
- Could mean Secure DNS lookups to authenticate incoming call credentials which are provisioned in the DNS
  - Might be the use case to drive DNSSEC uptake
- Very much at the bleeding edge
  - Hard to suggest likely time-lines

Costs

- How long is a piece of string?
- (Incremental) hardware and software costs for DNSSEC, DOH, DRPIVE and QUIC are probably minimal
  - Bigger iron shouldn't be necessary
  - New functionality probably bundled in software "for free"
- Real costs lie elsewhere and are (a) enterprise specific; (b) probably hard to quantify:
  - Training, migration, testing, documentation, processes, changes to IT policies, legal/regulatory considerations, RoI, risk/threat analysis, impact on installed base

Summary

- Secure DNS (DNSSEC)
  - Still a solution in search of a problem
- DPRIVE - DNS over (D)TLS
  - Probably going to flop or be a very niche service
  - Mobile space could change this - and fast!
- DOH - DNS over HTTP(S)
  - Will be very disruptive
  - Likely to get quick adoption - significant vendor buy-in
- QUIC - too early to tell for DNS

QUESTIONS?