

# Where's my DNS?

Sara Dickinson [sara@sinodun.com](mailto:sara@sinodun.com)

# The DNS protocol is evolving

DoT: DNS-over-TLS  
DoH: DNS-over-HTTPS (WIP)

- **DoT** RFC7858 standard May 2016
  - Implemented to-date in ‘standard’ open source DNS software
- **DoH** draft-ietf-doh-dns-over-https is in (through?) WGLC
  - Draft deals mainly with protocol, not
    - That DoH facilitates specific use cases: “via existing browser APIs”
    - Discovery of DoH servers (DRUI) - must have a URL

# What will this change?

- DoT/DoH will change stub to recursive DNS....
  - Use of encrypted DNS transports
  - System-wide resolution or per app?
    - Multiple resolvers per device?
  - Resolver: choice, discovery or defaults?

# What will this change?

- DoT/DoH will change stub to recursive DNS....
  - Use of encrypted DNS transports
  - System-wide resolution or per app?
    - Multiple resolvers per device?
  - Resolver: choice, discovery or defaults?

End User

Application Dev

Network Op

Resolver Op

# What will this change?

- DoT/DoH will change stub to recursive DNS....
  - Use of encrypted DNS transports
  - System-wide resolution or per app?
    - Multiple resolvers per device?
  - Resolver: choice, discovery or defaults?

Concentrate  
on this here





“What will I see?”

“...things that were... things that are...  
and some things...  
that have not yet come to pass.”

# Open Source Implementations Today

	Client	Recursive Resolver
<b>DoT</b>	<ul style="list-style-type: none"> <li>• <b>getdns</b> library &amp; <b>Stubby</b> (fwd)</li> <li>• Unbound/Knot resolver (fwd)</li> <li>• Android P: system config (dev)</li> <li>• <b>systemd</b> support</li> </ul>	<ul style="list-style-type: none"> <li>• <b>Unbound, Knot Resolver, dnsmdist</b></li> <li>• BIND on the way</li> </ul>
<b>DoH*</b>	<ul style="list-style-type: none"> <li>• getdns/Stubby (next release)</li> <li>• Android 'Intra' App</li> <li>• <b>Firefox Nightly</b> config option</li> <li>• Chrome/Bromite</li> <li>• Various experimental*</li> </ul>	<ul style="list-style-type: none"> <li>• <b>dnsmdist</b> (WIP)</li> <li>• Various experimental*</li> </ul>

\* 10+ implementations (see DoH mailing list and IETF 101 Hackathon)

# Recursive Resolver Deployment

	Standalone	Large Scale
DoT	<ul style="list-style-type: none"><li>• <u>20 test servers</u></li></ul>	<ul style="list-style-type: none"><li>• Quad9 (9.9.9.9)</li><li>• Cloudflare (1.1.1.1)</li></ul>
DoH*	<ul style="list-style-type: none"><li>• Google <a href="https://dns.google.com/experimental">https://dns.google.com/experimental</a></li><li>• <u>Few other test servers</u></li></ul>	<ul style="list-style-type: none"><li>• Cloudflare<ul style="list-style-type: none"><li>• <a href="https://cloudflare-dns.com/dns-query">https://cloudflare-dns.com/dns-query</a></li><li>• <a href="https://mozilla.cloudflare-dns.com/dns-query">https://mozilla.cloudflare-dns.com/dns-query</a></li></ul></li></ul>

\* Experimental, some support JSON as well as wireformat



# Encrypted DNS, what's not to love?

- Defeat **passive surveillance** ✓
- Can **authenticate** the server ✓
  - Prevents redirects
  - 'Increases' trust
- DoH - less susceptible to port and traffic **blocking** ✓

# Encrypted DNS, what's not to love?

- Defeat **passive surveillance** ✓
- Can **authenticate** the server ✓
  - Prevents redirects
  - 'Increases' trust
- DoH - less susceptible to port and traffic **blocking** ✓





# Encrypted DNS, reality check....?



- Increased **tracking** of user
  - Fixed resolver & connections, session resumption
  - DoH headers....? (e.g. user-agent)
- **Limited choice** of resolvers right now:
  - Breaks VPN/Split horizon DNS
  - SNI still leaks to network
- **Resolver** still sees all the traffic (Oblivious-DNS anyone?)
  - Choice of 1 resolver better than many (which one)?

Probably not one on the local network...

# System or App?

If in App...  
system or own settings?

"...allowing web applications to access DNS information via existing browser APIs"

# System or App?

- **Always been technically possible** for apps to do their own DNS but has:
  - largely been the **exception** (except some browsers)
  - have typically used the **system resolver** (8.8.8.8?)
  - **not been encrypted** (so still fully visible to user)
- Nothing to say an app ‘must use system **library and/or resolver**’
  - Just traditional architecture of end user devices
  - Easy for simple apps: one library call, no frills, reliable

**WHAT IF I TOLD YOU BROWSERS**

**ARE GOING TO DO THEIR OWN DOH**

# DNS in Browsers

- Some have always had their own DNS stub (e.g. Chrome)
- Some already use encrypted DNS
  - Yandex (DNSCrypt), Tenta (DNS-over-TLS)
- **Firefox Nightly already does DoH**
- **Firefox 62 (Sept 2018) will support DoH (by default?)**
- Chrome has a DoH implementation (not exposed)
  - Used in Bromite



# DoH in Browsers

- Why encrypt directly from the browser?
- Why DoH, not DoT? [Mozilla's answer.](#)



# DoH in Browsers

- Why encrypt directly from the browser?

OS's are slow to offer new DNS features (DoT/DoH)

Selling point: “we care about the privacy of our users”

Performance: “reduce latency within browser”

- Why DoH, not DoT? [Mozilla's answer.](#)

# DoH in Browsers

- Why encrypt directly from the browser?

OS's are slow to offer new DNS features (DoT/DoH)

Selling point: “we care about the privacy of our users”

Performance: “reduce latency within browser”

- Why DoH, not DoT? Mozilla's answer.

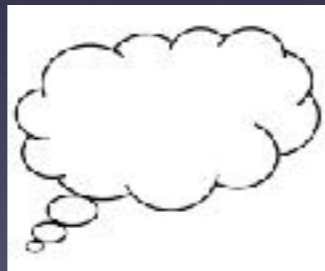
Integration: “leverage the HTTPS ecosystem”

HTTPS everywhere: “it works... just use port 443, mix traffic”

Cool stuff: “JSON, Server Push, ‘Resolverless DNS’....”

# DoH in Browsers

- Makes sense from a purely browser (application) view point
- But... bigger shift from 'traditional' DNS (including DoT)
- Unlikely browsers will change direction now....



- Thought experiment:
  - If DoH had been proposed in DPRIVE back in 2014... where would we be now (many solutions were considered)?

# DoH in Firefox



- Right now: Firefox Nightly 'experiment' (half of users, opt-out)
    - Use DoH to send all queries to Cloudflare **as well** as default resolver, compare the results
  - Overview of future plans, details of config & how it works
  - Plan:
- 
- Chrome, Safari, IE/Edge plans?



# DoH in Firefox



- Right now: Firefox Nightly 'experiment' (half of users, opt-out)
  - Use DoH to send all queries to Cloudflare **as well** as default resolver, compare the results

- Overview of future plans, details of config & how it works



- Plan:

- **“We’d like to turn this [DoH] on as the default for all of our users”**
- **Cloudflare is our ‘Trusted Recursive Resolver’ (TRR) - more later**



CLOUDFLARE

- Chrome, Safari, IE/Edge plans?

# Short term vs long term

- **Short term DoH in browsers:**
  - In reality, Cloudflare are the only **large scale DoH** provider today
  - Need ISPs, etc. to catch up
  - Cloudflare might be the default but user can configure their own resolver **if** they know where to look (Google, Quad9?)
  - No **discovery mechanisms** for DoH servers available
    - Pre-defined list/default/user override is only option

# Short term vs **long term**

- Consider end user workflows (on different devices):
  - **Browser** based *desktop* workflow (for cloud based data)
  - **App** based *mobile* workflow
- Split: **Browser/the rest?**
  - What will the default resolver model be for browsers?
  - ‘Opportunistic/Resolverless DNS’ Discover a DoH server within a domain (browser tab) and use that... *minimised leakage*
    - **Change of trust model or more?**

# Short term vs **long term**

- Consider end user workflows (on different devices):
  - **Browser** based *desktop* workflow (for cloud based data)
  - **App** based *mobile* workflow
- Split: **Browser/the rest?**
  - What will the default resolver model be for browsers?
  - ‘Opportunistic/Resolverless DNS’ Discover a DoH server within a domain (browser tab) and use that... *minimised leakage*
    - **Change of trust model or more?** **DoHNS?**



# Short term vs **long term**

- **Most other apps** also do their own DoH/DoT?
  - If OS's remain slow to update DNS (as with DNSSEC), this is likely...
  - Quality and range of DNS libraries improves e.g. getdns, Javascript libraries this is more likely....
  - Wide enough deployment of DoH/DoT servers (available everywhere or just a few big operators)?
- **Privacy increase plus individual apps see the gains, but will the overall 'user experience' suffer?**

# Short term vs long term

- **Most other apps** also do their own DoH/DoT?

Or a huge mixture...

- If OS's remain slow to update DNS (as with DNSSEC), this is likely...
  - Quality and range of DNS libraries improves e.g. getdns, Javascript libraries this is more likely....
  - Wide enough deployment of DoH/DoT servers (available everywhere or just a few big operators)?
- **Privacy increase plus individual apps see the gains, but will the overall 'user experience' suffer?**

# Dude, where's my DNS?

- In an ideal world all apps that do their own DNS would consistently
  - Implement all DNS options (all transports, DNSSEC support, etc.)
  - Respect system settings (DHCP/user resolver, search domains, DNSSEC, etc.)
  - Be highly transparent about DNS settings (defaults, DoH headers, cookie use, etc..)
  - Expose low-level debugging of DNS queries (current debug in Firefox is limited...)

# Dude, where's my DNS?

- In an ideal world all apps that do their own DNS would consistently
  - Implement all DNS options (all transports, DNSSEC support, etc.)
  - Respect system settings (DHCP/user resolver, search domains, DNSSEC, etc.)
  - Be highly transparent about DNS settings (defaults, DoH headers, cookie use, etc..)
  - Expose low-level debugging of DNS queries (current debug in Firefox is limited...)

Can't enforce





Can't enforce

Can't enforce

Can't enforce





# Fragmented system DNS

DNS is no longer part of the device infrastructure with a single point of configuration....

- If not...
- Just another form of content? Possibly multiple name systems?
- Multiply config issues by number of devices a user has
- Multiple config points (transport, authentication)
  - Importantly DNSSEC 
- Multiple recursive resolvers (privacy gains)
  - Scatter queries/reduce leakage 
  - What if some fail, get blocking or attacked 
- Multiple points for monitoring/debugging? 

# Fragmented system DNS

DNS is no longer part of the device infrastructure with a single point of configuration....

- If not...
- Just another form of content? Possibly multiple name systems?
- Multiply config issues by number of devices a user has
- Multiple config points (transport, authentication)
  - Importantly DNSSEC 
- Multiple recursive resolvers (privacy gains)
  - Scatter queries/reduce leakage 
  - What if some fail, get blocking or attacked? 
- Multiple points for monitoring/debugging? 

Different failure mode than today..  
Wireshark/dig can't help you here

# Will users notice or care?

- If our muggle friends don't then we should!
- They won't notice if apps don't even expose that they do this.....
- They might depending on how transparent it is and the UX:

# Will users notice or care?

- If our muggle friends don't then we should!
- They won't notice if apps don't even expose that they do this.....
- They might depending on how transparent it is and the UX:

**Welcome to my\_app version X!**

Click to continue

(Terms & conditions)



# Will users notice or care?

- If our muggle friends don't then we should!
- They won't notice if apps don't even expose that they do this.....
- They might depending on how transparent it is and the UX:

## **Welcome to my\_app version X!**

In this release we are protecting your DNS  
- aren't we fab!

Click to continue

(Terms & conditions)

# Will users notice or care?

- If our muggle friends don't then we should!
- They won't notice if apps don't even expose that they do this.....
- They might depending on how transparent it is and the UX:

## **Welcome to my\_app version X!**

We are trying to improve the privacy of your DNS but do this we need to re-route all your DNS queries to a company based on Mars you probably haven't even heard of.

- Don't know what DNS is? Just click here to blindly accept our T&C's!
- Total geek? Click here to see the gory details...

(Terms & conditions)

# Trusted Recursive Resolver 'TRR'

# TRR

“ With this, we have a resolver that we can trust to protect users’ privacy. This means **Firefox can ignore the resolver that the network provides** and just go straight to Cloudflare.”

- Implicit consent model:
  - (Current) Log onto a network and use the DHCP provided resolver
  - (New?) Use an app and agree to app T&C’s (including DNS?)

# TRR

- Cloudflare are good (not perfect) - not all TRRs will be!
- Might end up with a few 'big' TRR providers
- Development companies set up own server (quality?)
- Applications be persuaded to use a certain 'TRR' in return for money?
- Work in progress on Best Current Practices for Operators...
  
- Bypassing network resolver (enterprise/user issue):
  - Breaks VPN, split horizon, leaks internal queries
  - Can use fallback but slow and still leaks queries

# TRR

- Cloudflare are good (not perfect) - not all TRRs will be!
- Might end up with a few 'big' TRR providers
- Development companies set up own server (quality?)
- Applications be persuaded to use a certain 'TRR' in return for money?
- Work in progress on Best Current Practices for Operators...
- Bypassing network resolver (enterprise/user issue):
  - Breaks VPN, split horizon, leaks internal queries
  - Can use fallback but slow and still leaks queries

Mitigation/motivation  
for operators to  
deploy

# Dude, where's my DNS?

I'm not judging...  
I'm just saying...

Mixed, uncertain future for the camel (1, 3, 30 yrs?)...

# Dude, where's my DNS?

I'm not judging...  
I'm just saying...

Mixed, uncertain future for the camel (1, 3, 30 yrs?)...



It's DNS Jim,  
but not as we know it



Thank you!