EDU Tutorial:
DNS Privacy

Sara Dickinson
Sinodun
sara@sinodun.com
Overview

• Goal:
  • Give audience historical background on why DNS Privacy is an important topic
    • Internet Privacy - presented by dkg
  • Chart progress during last 3-4 years (DPRIVE)
  • Present current status and tools
Internet Privacy

Daniel Kahn Gillmor
ACLU
DNS Privacy
- A brief history
## IETF Privacy activity

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>March 2011</td>
<td><strong>I-D:</strong> Privacy Considerations for Internet Protocols (IAB)</td>
</tr>
<tr>
<td>June 2013</td>
<td>Snowdon revelations</td>
</tr>
<tr>
<td>July 2013</td>
<td><strong>RFC6973:</strong> Privacy Considerations for Internet Protocols</td>
</tr>
<tr>
<td>May 2014</td>
<td><strong>RFC7258:</strong> Pervasive Monitoring is an Attack</td>
</tr>
<tr>
<td>August 2015</td>
<td><strong>RFC7624:</strong> Confidentiality in the Face of Pervasive Surveillance: A Threat model and Problem Statement</td>
</tr>
<tr>
<td></td>
<td>Much other ongoing work.....</td>
</tr>
</tbody>
</table>
“PM is an attack on the privacy of Internet users and organisations.”

“…that needs to be mitigated where possible, via the design of protocols that make PM significantly more expensive or infeasible.”
DNS Privacy in 2013?

• DNS [RFC1034/5 - 1987] - original design availability, redundancy and speed! (DNS is an enabler)

• DNS standards:
  • UDP (99% of traffic to root)
  • TCP only for ‘fallback’ when UDP MTU exceeded and XFR (support only mandatory from 2010)

• Perception: The DNS is public, right? It is not sensitive/personal information….it doesn’t need to be protected/encrypted

DNS sent in clear text
  -> NSA: ‘MORECOWBELL’
DNS Disclosure Example 1

Leak information → datatracker.ietf.org

Root

Auth for .org

datatracker.ietf.org

datatracker.ietf.org

datatracker.ietf.org

Auth for ietf.org

Rec
DNS Privacy in 2013?

- **RFC6891**: Extension Mechanisms for DNS (EDNS0)

  Intended to enhance DNS protocol capabilities

- But…. mechanism enabled addition of end-user data **into** DNS queries (non-standard options)

  - ISP justification: Parental Filtering (per device)
  - CDN justification: Faster content (geo location)
DNS Disclosure Example 2

[User src address] MAC address or id in DNS query

Client Subnet (RFC7871) contains source subnet in DNS query

Stub

Rec

Auth

CPE

[ietf.org ? [00:00:53:00:53:00]]

[? ietf.org ? [192.168.1]]
DNS Disclosure Example 2

Even behind a NAT, do not have anonymity!

Even behind a recursive do not have anonymity!
DNS Disclosure Example 3

- (AUTH) Who monitors or has access here?
- (UNAUTH) How safe is this data?

When at home…
When in a coffee shop…

Who monitors or has access here?
DNS - complications

• Basic problem is leakage of meta data
  • Allows re-identification of individuals

• Even without user meta data traffic analysis is possible based just on timings and cache snooping

• **DNS Filtering** is becoming more prevalent
## DNS Risk Matrix

<table>
<thead>
<tr>
<th>Risk</th>
<th>In-Flight</th>
<th>At Rest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Stub =&gt; Rec</td>
<td>Rec =&gt; Auth</td>
</tr>
<tr>
<td>Passive Monitoring</td>
<td></td>
<td></td>
</tr>
<tr>
<td>At Rest</td>
<td></td>
<td></td>
</tr>
<tr>
<td>At Authoritative</td>
<td></td>
<td></td>
</tr>
<tr>
<td>At Recursive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other Disclosure Risks e.g. Data breaches</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
DNS Service Discovery

- Devices advertise services on local network (DNS, mDNS)
- Other devices then discover the service and use it

- Alice's Images . _imageStore._tcp . local
- Alice's Mobile Phone . _presence._tcp . local
- Alice's Notebook . _presence._tcp . local
DNS-SD Privacy

• Advertising leaks information about:
  • User - ‘name’, devices, services (user tracking)
  • Devices - services & attributes (port, priorities)
    • Device fingerprinting possible

=> Software or specific device identification

• Discovery leaks info about preferred services
DNS Privacy options (2013)

- **DNSCurve**
  - Daniel J. Bernstein, initial interest but not adoption

- **DNSCrypt**
  - Many implementations, several open DNSCrypt Resolvers ([OpenDNS](https://www.opendns.com)), [Yandex browser](https://yandex.com/)

- **Authentication** with some privacy
  - Documented but not standard

- **Recursive-Auth**
- **Stub-Recursive**
- **Anti-spoofing, anti DoS**
DNS Privacy options (2014)

- Run a local resolver (Unbound)
- DNSTrigger (NLNet Labs)
- Client software to enable DNSSEC
- Used TLS on port 443 as last ditch attempt to enable DNSSEC (DNS-over-TLS impl)

Goal was DNSSEC, not Privacy!
DPRIVE WG

- DPRIVE WG create in 2014

Charter: Primary Focus is Stub to recursive

- Why not tackle whole problem?
  - Don’t boil the ocean, stepwise solution
  - Stub to Rec reveals most information
  - Rec to Auth is a particularly hard problem
DNS Privacy problem

Relationship: 1 to ‘a few’ some of whom are known (ISP)

Relationship: 1 to many most of whom are not known => Authentication is hard
RFC 7626 - DNS Privacy Considerations

• Problem statement: Expert coverage of risks throughout DNS ecosystem

• Rebuts "alleged public nature of DNS data"

  • The data may be public, but a DNS ‘transaction’ is not/should not be.

"A typical example from outside the DNS world is: the web site of Alcoholics Anonymous is public; the fact that you visit it should not be."
Choices, choices...

- So... we know the problem but what mechanism to use for encrypting DNS?
  - STARTTLS
  - TLS
  - DTLS
  - Confidential DNS draft

Drafts submitted on all these solutions to the working group
## Encryption Options

<table>
<thead>
<tr>
<th></th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>STARTTLS</strong></td>
<td>• Port 53</td>
<td>• Downgrade attack on negotiation</td>
</tr>
<tr>
<td></td>
<td>• Known technique</td>
<td>• Port 53 - middleboxes blocking?</td>
</tr>
<tr>
<td></td>
<td>• Incrementation deployment</td>
<td>• Latency from negotiation</td>
</tr>
<tr>
<td><strong>TLS (new port)</strong></td>
<td>• New DNS port (no interference with port 53)</td>
<td>• New port assignment</td>
</tr>
<tr>
<td></td>
<td>• Existing implementations</td>
<td>• Scalability?</td>
</tr>
<tr>
<td><strong>DTLS (new port)</strong></td>
<td>• UDP based</td>
<td>• Truncation of DNS messages (just like UDP)</td>
</tr>
<tr>
<td></td>
<td>• Not as widely used/deployed</td>
<td>◐ Fallback to TLS or clear text</td>
</tr>
<tr>
<td></td>
<td></td>
<td>✗ Can’t be standalone solution</td>
</tr>
</tbody>
</table>
Encrypted DNS ‘TODO’ list

- Get a new port
- DNS-over-TCP/TLS: Address issues in standards and implementations
- Tackle authentication of DNS servers (bootstrap problem)
- What about traffic analysis of encrypted traffic - msg size & timing still tell a lot!
Get a new port!

- One does not simply get a new port…

- Oct 2015 - 853 is the magic number

Your request has been processed. We have assigned the following system port number as an early allocations per RFC7120, with the DPRIVE Chairs as the point of contact:

<table>
<thead>
<tr>
<th>Service</th>
<th>Port</th>
<th>Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain-s</td>
<td>853</td>
<td>tcp</td>
<td>DNS query-response protocol run over TLS/DTLS</td>
</tr>
<tr>
<td>domain-s</td>
<td>853</td>
<td>udp</td>
<td>DNS query-response protocol run over TLS/DTLS</td>
</tr>
</tbody>
</table>
DNS + TCP/TLS?

- DNS-over-TCP history:
  - typical DNS clients do ‘one-shot’ TCP
  - DNS servers have very basic TCP capabilities
  - No attention paid to TCP tuning, robustness
  - Performance tools based on one-shot TCP
## Fix DNS-over-TCP/TLS

<table>
<thead>
<tr>
<th>Goal</th>
<th>How?</th>
</tr>
</thead>
</table>
| Optimise set up & resumption              | TFO Fast Open  
TLS session resumption  
[TLS 1.3]                             |
| Amortise cost of TCP/TLS setup            | RFC7766 (bis of RFC5966) - March 2016:  
Client pipelining (not one-shot!),  
Server concurrent processing,  
Out-of-order responses |
| Servers handle many connections robustly  | RFC7828: Persistent connections (Keepalive)                           |
|                                           | Learn from HTTP world!                                              |
Performance (RFC7766)

**Client** - pipeline requests, keep connection open and handle out-of-order response

**Server** - concurrent processing of requests sending of out of order responses

- **in-order**
  - q1, q2
  - q1, q2 delayed waiting for q1 (+1 RTT)
  - q2
  - q1
  - a1
  - a2

- **concurrent, OOOR**
  - q1, q2
  - 0 extra RTT
  - reply as soon as possible
  - q1
  - q2
  - a2
  - a1

---

DNS Privacy Tutorial @ IETF 97

Nov 2016, Seoul
Authentication in DNS-over-(D)TLS

2 Usage Profiles:

• Strict
  • “Do or do not. There is no try.”

• Opportunistic
  • “Success is stumbling from failure to failure with no loss of enthusiasm”

Try in order:

• Encrypt & Authenticate then
• Encrypt then
• Clear text

(Encrypt & Authenticate) or Nothing
Authentication in DNS-over-(D)TLS

- Authentication based on config of either:
  - Authentication domain name
  - SPKI pinset

- Shouldn’t DNS use DANE…? Well - even better:
  - I-D: TLS DNSSEC Chain Extension
DNS Auth using DANE

1: Obtain a Auth Domain name & IP address
(1a) • Configure Auth domain name
• Do Opportunistic SRV lookup

2a: • Opportunistic lookup of DANE records for server
• Validate locally with DNSSEC
1: Obtain a Auth Domain name & IP address

(1a)
- Configure Auth domain name
- Do Opportunistic SRV lookup

0 (or 2): Obtains DANE records for itself!

- Reduces Latency
- Eliminates need for intermediate recursive
# DPRIVE Solution Documents (stub to recursive)

<table>
<thead>
<tr>
<th>Document</th>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC7858</td>
<td>May 2016</td>
<td>DNS-over-TLS</td>
</tr>
<tr>
<td>RFC7830</td>
<td>May 2016</td>
<td>EDNS0 Padding Option</td>
</tr>
<tr>
<td>draft-ietf-dprive-dnsodtls*</td>
<td>Completed WGLC</td>
<td>DNS-over-DTLS</td>
</tr>
<tr>
<td>draft-ietf-dprive-dtls-and-tls-profiles</td>
<td>In WGLC</td>
<td>Authentication for DNS-over-(D)TLS</td>
</tr>
</tbody>
</table>

*Intended status: Experimental*
What about Recursive to Authoritative?

- DPRIVE - Re-charter? WG this Friday!
- I-D: Next step for DPRIVE: resolver-to-auth link
  - Presents 6 authentication options/models
  - Data on DNS-over-(D)TLS
- DNSOP - RFC7816: QNAME Minimisation
DNS Disclosure Example 1

Leaks information

Root

Auth for .org

Auth for ietf.org

datatracker.ietf.org

Rec

datatracker.ietf.org

datatracker.ietf.org

datatracker.ietf.org

datatracker.ietf.org

datatracker.ietf.org

datatracker.ietf.org

Leaks information

datatracker.ietf.org
QNAME Minimisation

datatracker.ietf.org
Rec
org
ietf.org
datatracker.ietf.org
Root
Auth for .org
Auth for ietf.org
Datatracker for ietf.org
Data handling policies

- Do you read the small print of your ISPs contract?
- More work/research needed in this area
  - Monitoring of government policy and practice
  - Transparency from providers on policy and breaches
  - Methods for de-identification of user data (e.g. DITL)
  - ‘PassiveDNS’ data used for research/security
DNS-over-HTTP(S)

- DNS-over-HTTP(S) has been around a while…
  - I-D: Review of DNS-over-HTTP
- Privacy (HTTPS authentication)
- Bypass port 53 interference (middlebox, captive portals)
- Higher level API
DNS-over-HTTP(S)

- Google: DNS-over-HTTPS (non-standard)
- I-D: DNS wire-format over HTTP
  - “Servers and clients SHOULD use TLS for communication.”
- I-D: DNS Queries over HTTPS
- Non-WG Mailing list and Bar BOF here (Tuesday)
# Risk Mitigation Matrix

<table>
<thead>
<tr>
<th>Risk</th>
<th>In-Flight</th>
<th>At Rest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Stub =&gt; Rec</td>
<td>Rec =&gt; Auth</td>
</tr>
<tr>
<td>Passive monitoring</td>
<td>Encryption (e.g. TLS, HTTPS)</td>
<td>QNAME Minimization</td>
</tr>
<tr>
<td>Active monitoring</td>
<td>Authentication &amp; Encryption</td>
<td></td>
</tr>
<tr>
<td>Other Disclosure Risks e.g. Data breaches</td>
<td></td>
<td>Data Best Practices (Policies) e.g. De-identification</td>
</tr>
</tbody>
</table>
DNS-SD

- I-D: Privacy Extensions for DNS-SD - adopted by WG

- 3 step design

  1. Offline pairing mechanism (shared secret)
  2. Discovery of the “Private Discovery Service”
  3. Actual Service Discovery (enc & auth conn)
Implementation Status
# Recursive implementations

<table>
<thead>
<tr>
<th>TCP/TLS Features</th>
<th>Features</th>
<th>Unbound</th>
<th>BIND</th>
<th>Knot Res</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TCP fast open</td>
<td>green</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Process pipelined queries</td>
<td>green</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Provide OOOR</td>
<td>yellow</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EDNS0 Keepalive</td>
<td>yellow</td>
<td></td>
<td>grey</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TLS Features</th>
<th>Features</th>
<th>Unbound</th>
<th>BIND</th>
<th>Knot Res</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TLS on port 853</td>
<td>green</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Provide server certificate</td>
<td>green</td>
<td></td>
<td>purple</td>
</tr>
<tr>
<td></td>
<td>EDNS0 Padding</td>
<td>grey</td>
<td></td>
<td>yellow</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rec =&gt; Auth</th>
<th>QNAME Minimisation</th>
<th>Unbound</th>
<th>BIND</th>
<th>Knot Res</th>
</tr>
</thead>
</table>

| Dark Green: | Latest stable release supports this |
| Light Green:| Patch available        |
| Yellow:     | Patch/work in progress, or requires building a patched dependency |
| Purple:     | Workaround available   |
| Grey:       | Not applicable or not yet planned |
Alternative server side solutions

- Pure TLS load balancer
  - NGINX, HAProxy
  - BIND article on using stunnel
- dnsdist from PowerDNS would be great…
- But no support yet

Disadvantages
- server must still have decent TCP capabilities
- DNS specific access control is missing
- pass through of edns0-tcp-keepalive option
## Stub implementations

<table>
<thead>
<tr>
<th>Features</th>
<th>ldns</th>
<th>digit</th>
<th>getdns</th>
<th>BIND (dig)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TCP/TLS Features</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCP fast open</td>
<td>*</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Connection reuse</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Pipelining of queries</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Process OOOR</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>EDNS0 Keepalive</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td><strong>TLS Features</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TLS on port 853</td>
<td>*</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Authentication of server</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>EDNS0 Padding</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

- Dark Green: Latest stable release supports this
- Light Green: Patch available
- Yellow: Patch/work in progress, or requires building a patched dependency
- Grey: Not applicable or not yet planned

*getdns uses libunbound in recursive mode*
Implementation Status

• Increasing uptake of better DNS-over-TCP, QNAME minimisation
• Several implementations of DNS-over-TLS
• None yet of DNS-over-DTLS
• BII has DNS-over-HTTP implementation

Key is enabling end users and application developers to easily adopt DNS Privacy
Deployment Status
## DNS-over-TLS Servers

<table>
<thead>
<tr>
<th>Hosted by</th>
<th>Software</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLnet Labs</td>
<td>Unbound</td>
</tr>
<tr>
<td>OARC</td>
<td>Unbound</td>
</tr>
<tr>
<td>Surfnet (Sinodun)</td>
<td>Bind + HAProxy Bind + nginx</td>
</tr>
<tr>
<td>dkg</td>
<td>Knot Resolver</td>
</tr>
<tr>
<td>IETF?</td>
<td></td>
</tr>
</tbody>
</table>

Find details at: [DNS Test Servers](#)

---

**DNS Privacy Tutorial @ IETF 97**

*Nov 2016, Seoul*
• Modern **async DNSSEC** enabled API
  
  • [https://getdnsapi.net](https://getdnsapi.net)

• Written in C, various bindings (Python, Java,...)

• DNS-over-TLS, validating DNSSEC stub

• ‘Stubby’ now available for testing
Stubby

- A privacy enabling stub resolver (based on getdns_query tool)
- 1.1.0-alpha3
- Run as daemon handling requests
- Configure OS DNS resolution to point at 127.0.0.1
Stubby In Action

- Reads config from /etc/stubby.conf
- domain name and SPKI pinset authentication
- Strict and Opportunistic profiles

- How to build and use Stubby
- Demos available: Sara, Willem Toorop, Allison Mankin
Stubby in Action
Ongoing and Future work

- Hacking this weekend at the IETF 97 Hackathon
  - lots of work on Stubby and test servers
- OS integration of client solutions
- More complete recursive implementations
- Increased deployment
- More DPRIVE work: Recursive to Auth….
Summary

• DNS Privacy is a real problem and more relevant than ever

• Active work on the large solution space

• Can test DNS Privacy today using Stubby & current test recursive servers

• More DNS Privacy services on the way…
Thank you!

Any Questions?

https://www.surveymonkey.com/r/97privacy