

DNS/DNSSEC/DANE/DNS-over-TLS etc. Team – IETF95 Hackathon

In-Person:

Ray Bellis, Sebastian Castro, Sara Dickinson, John Dickinson, Ralph Dolman, Robert Edmonds, Evan Hunt, Shumon Huque, Daniel Kahn Gillmor, Shane Kerr, Dave Lawrence, Allison Mankin, Benno Overeinder, Jan Včelák, Dan York

Remote:

Linus Nordberg, Melinda Shore, Marek Vavruša, Gowri Visweswaran

Following slides represent some of the efforts.
Check with individuals for more details.

Varied Projects

Sources

- RFC 7766 (DNS-over-TCP)
- draft-ietf-dprive-dns-over-tls
- draft-ietf-dnsop-edns-chain-query
- Draft-shore-tls-dnssec-chain-extension (DANE/TLS)

Platforms

- BIND9
- Unbound
- Knot Recursive
- getdns

Other Topics

- Performance
- Security hardening

BIND9 - Chain Query — Dave Lawrence

- Added EDNS CHAIN option (DNSOP draft in RFC-Editor) to dig (+chain or +chain=closest.trust.point).
- Added named options to allow chain as a server or request chain when forwarding.
- Only replies with chain when over TCP or with valid cookie.
- DOESN'T ACTUALLY YET INCLUDE THE OTHER DNSSEC RECORDS
- Added subsystem test.
- Example screenshot shows the current tests along with one dig showing
- CHAIN option in request and reply

Chain Query Screen Shot

```
1. dhcp-89e5.meeting.ietf.org (zsh)
(ssh) emacs zsh zsh zsh
~/src/bind-9.10.3-P4/bin/tests/system
: dhcp-89e5:tale 32; sh run.sh chain
S:chain:Sun Apr 3 13:28:03 ART 2016
T:chain:1:A
A:System test chain
I:chain over UDP, no SIT
I:chain over TCP,
I:chain over UDP, with initial SIT
I:chain over UDP, with full SIT

; <<>> DiG 9.10.3-P4 <<>> +qr +chain=com +sit=3f8e3faaca73ab839d3b1e0857014495d9
92729fe90cc1e8 www.example.com -p 5300 @10.53.0.1
;; global options: +cmd
;; Sending:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 821
;; flags: rd ad; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; CHAIN: 03 63 6f 6d 00 (".com.")
;; COOKIE: 3f8e3faaca73ab839d3b1e0857014495d992729fe90cc1e8
;; QUESTION SECTION:
;; www.example.com. IN A

;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 821
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 5

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; COOKIE: 3f8e3faaca73ab83268763ac57014495f644f46c33cd6e97 (good)
;; CHAIN: 03 63 6f 6d 00 (".com.")
;; QUESTION SECTION:
;; www.example.com. IN A

;; ANSWER SECTION:
www.example.com. 86400 IN A 93.184.216.34

;; AUTHORITY SECTION:
example.com. 172799 IN NS a.iana-servers.net.
example.com. 172799 IN NS b.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net. 172800 IN A 199.43.132.53
a.iana-servers.net. 1800 IN AAAA 2001:500:8c::53
b.iana-servers.net. 1800 IN A 199.43.133.53
b.iana-servers.net. 1800 IN AAAA 2001:500:8d::53

;; Query time: 0 msec
;; SERVER: 10.53.0.1#5300(10.53.0.1)
;; WHEN: Sun Apr 03 13:28:05 ART 2016
;; MSG SIZE rcvd: 233

I:exit status: 0
R:PASS
E:chain:Sun Apr 3 13:28:06 ART 2016
~/src/bind-9.10.3-P4/bin/tests/system
: dhcp-89e5:tale 32; █
```

Unbound – Chain Query – Ralph Dolmans

- Partially completed implementation of EDNS0 chain query in the Unbound recursive open source
- To be continued, including interoperability testing with implementations in BIND9, dig etc.

getdns - DNSSEC Transparency –

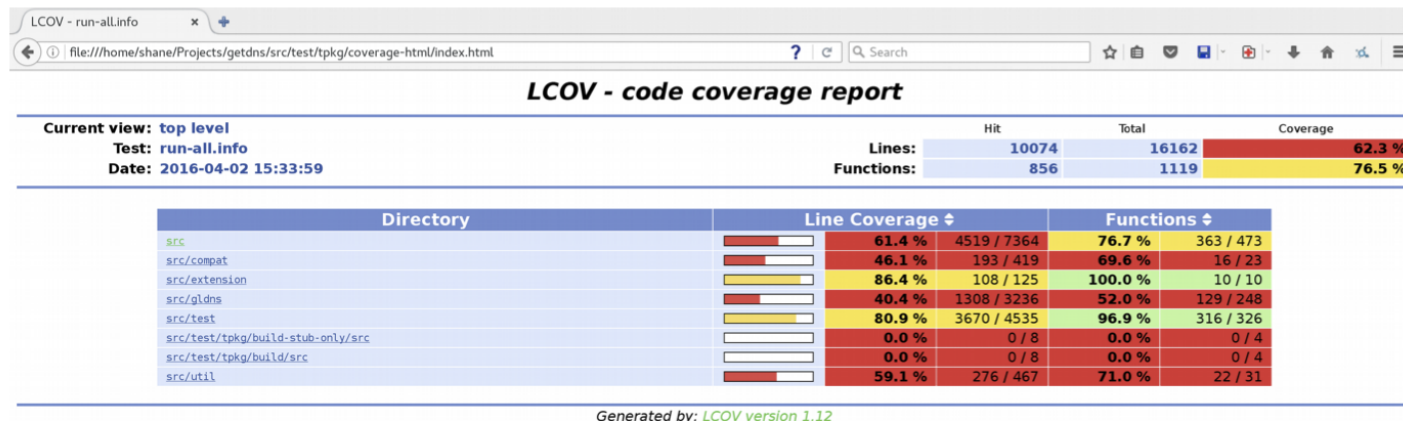
Linus Nordberg

- TRANS WG effort, see reporting (to come) on dnssec-trans mailing list
- Working tools test out well so far
- Erlang port in C_src/dnssec.c is the untested part
- Please get in touch if you are a fiendish DNSSEC tester and would like to contribute more tests

getdns – API Hardening – Shane Kerr

getdns API hardening

- getdns 1.0 approaching
- Don't want a repeat of glibc DNS problem! ;)
- Improve testing
 - Code coverage
 - Integrate with Deckard



Knot Recursive - DNS-over-TLS (and TCP OOOOP) - Jan Včelák, Daniel Kahn Gillmor, Marek Vavruša

- Knot Recursive DNS Server
<http://knot-resolver.readthedocs.org/en/latest/build.html>
- Added support for DNS over TLS (DPRIVE draft in RFC-Editor Queue)
- For this to perform well, needed TCP out-of-order processing (OOOP), and this was added too.

BIND9 - dnssec-keymgr — Evan Hunt and Sebastian Castro

- **Code available at <https://github.com/each/bind9-collab>**
- Defined a list of features/bug fixes/documentation we wanted to achieve this weekend
 - Features: Generate new keys based on a policy (DNSSEC bootstrapping).
 - Flags to make it more verbose
 - Bugfixes: Lots of changes to comply with PEP8 (coding guideline for Python). More robust error handling. Policy validation
 - Documentation: How to use the tool to fully sign and manage a zone with DNSSEC
- Lessons learned: How to write better Python code, cleaner, following guidelines. Better documentation. Lots of new features. Discovered bugs associated to new tools.

getdns – Performance Testing – John Dickinson

- Plan was to exercise different transport modes (UDP, TCP, TLS)
- Wanted to test DNS name server performance
- Ended up profiling *getdns* performance instead!
Discovered some limitations that need investigating...
 - (File desc limits, TCP 0 Window size).

getdns – Google Public DNS-over-HTTPS (and HTTP) – Sara Dickinson

- Announced April 1st (not a joke!)
 - Top tweet among those from #OARC24
- Not based on a standard but investigated behaviour.
Report at link below

<https://portal.sinodun.com/wiki/display/TDNS/Google%27s+P>

- Started implementation in *getdns* but not finished

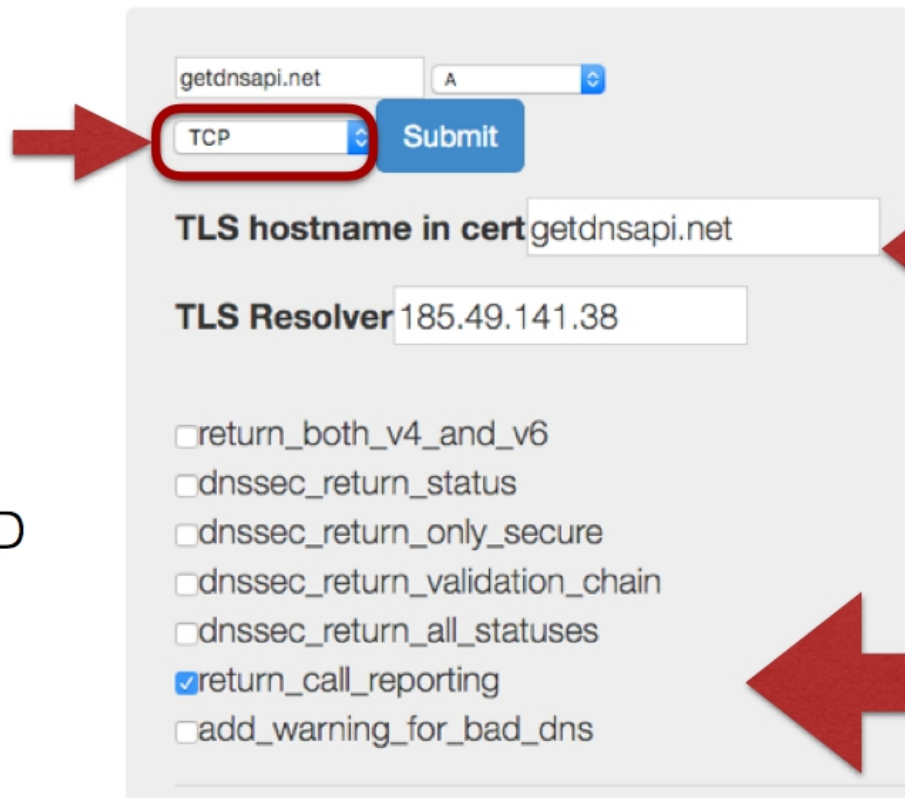
getdns – enhanced web-based query tool – Gowri Visweswaran

OFFICIAL: <https://getdnsapi.net/query.html>

TEST: <https://getdnsapi.net/gowri.html>

New Transport options:

- UDP
- TCP
- UCP, TCP
- TCP, UDP
- TLS
- TLS, TCP, UPD



The screenshot shows the web-based query tool interface. A red arrow points to the 'TCP' option in the transport dropdown menu, which is highlighted with a red circle. Another red arrow points to the 'TLS hostname in cert' field, which contains 'getdnsapi.net'. A third red arrow points to the 'return_call_reporting' checkbox, which is checked. The interface includes a 'Submit' button and various other options like 'TLS Resolver' and 'return_both_v4_and_v6'.

New Auth and Resolver options

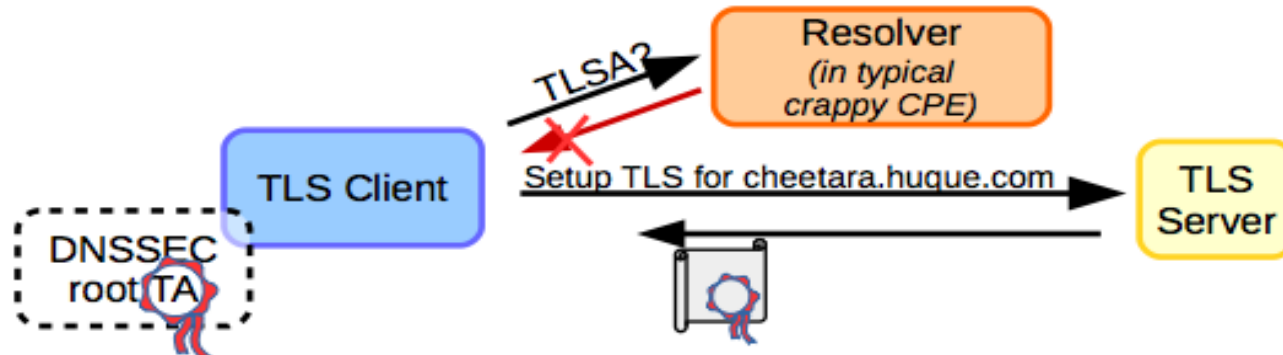
New extensions

Getdns – TLS Extension – Shumon Huque and Willem Toorop

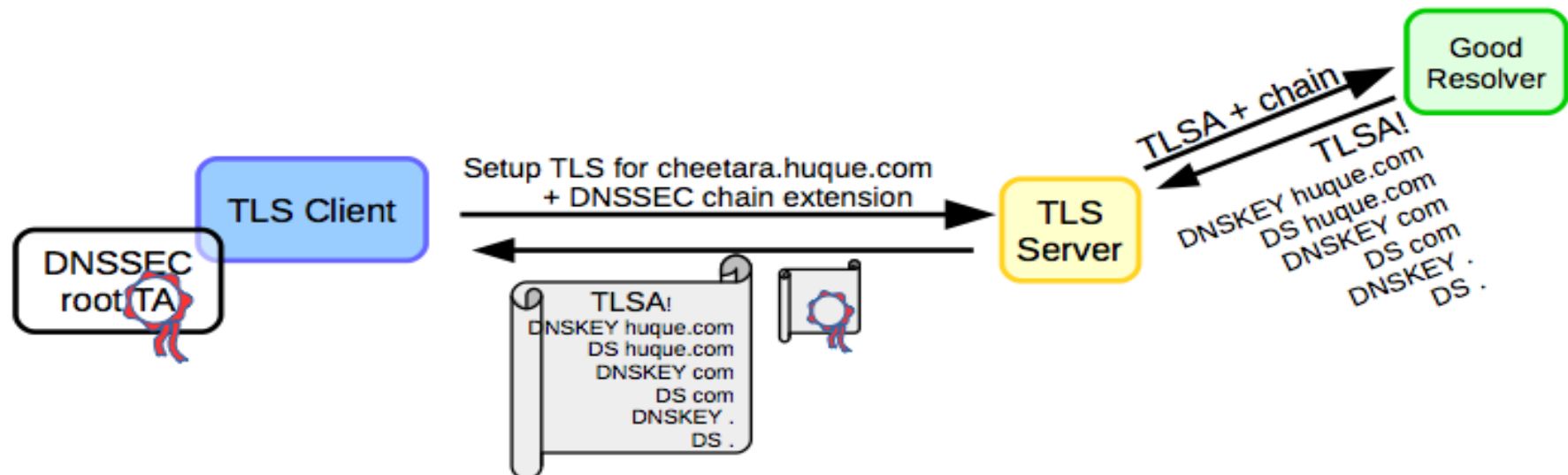
- Short presentation by Shumon and Willem
- Implementing draft proposed to TLS WG

the problem case

- No DNSSEC
- Only address queries
- Hideous to query complete chain



With the TLS DNSSEC Chain Extension



Languages used: **C**

Libraries used:

- * **OpenSSL 1.1.x:**

- * all TLS functions; custom extension construction and processing; DANE TLSA certificate authentication.

- * **getdns 1.0.0b1**

- * TLSA/DNSSEC chain query, generation, and verification.

Sample output from the server:

```
$ ./chainserver -d 5001
```

```
Built DNSSEC chain data for
```

```
_5001._tcp.cheetara.huque.com, size=3039 octets
```

```
Chain Server listening on port 5001
```

Sample run from the client:

```
$ ./chainclient -d cheetara.huque.com 5001
```

```
Connecting to IPv4 address: 50.116.63.23 port 5001
```

```
Sent DNSSEC chain extension (53)
```

```
Received DNSSEC chain extension (53). #octets: 3039
```

```
>> Debug: RR: _5001._tcp.cheetara.huque.com. TLSA
```

```
>> Debug: RR: huque.com. DNSKEY
```

```
>> Debug: RR: huque.com. DNSKEY
```

```
>> Debug: RR: huque.com. DS
```

```
>> Debug: RR: com. DNSKEY
```

```
>> Debug: RR: com. DNSKEY
```

```
>> Debug: RR: com. DS
```

```
[...]
```

```
DNSSEC status: SECURE
```

```
TLShv1.2 handshake succeeded.
```

```
Cipher: TLShv1.2 ECDHE-RSA-AES256-GCM-SHA384
```

```
DANE TLSA 3 1 1 [8477cdb8e095...] matched EE cert at  
depth 0
```

```
Validated Certificate chain:
```

```
0 Subject CN: cheetara.huque.com
```

```
Issuer CN: cheetara.huque.com
```

```
SAN dNSName: cheetara.huque.com
```
