DNS privacy at IETF

Stéphane Bortzmeyer

AFNIC

bortzmeyer@nic.fr

DNS privacy at IETF

Stéphane Bortzmeyer

AFNIC

bortzmeyer@nic.fr

Context

1. Unlike what many people say, IETF worked on privacy before Snowden: RFC 6973 (highly recommended reading)
2. The Snowden revelations: it's not longer possible to hide in the sand. Powerful orgs **massively** spy on us and less powerful try hard to do it, too.
3. Vancouver, november 2013: IETF pledged to "harden the Internet" `http://www.ietf.org/media/2013-11-07-internet-privacy-and-security.html`
4. Actual work: perpass list for general discussions, approval of the statement "Pervasive Monitoring is an Attack" (in RFC Editor queue), tutorial in meetings ("Privacy for engineers"), privacy reviews of old protocols (see the list ietf-privacy)...

The DNS case

1. Old protocol, designed in different times
2. Leaks a lot
3. Many strong constraints: latency, reliability, ubiquity
4. First work on its privacy at CENTR (Hi, Nathalie)

Open data

An actual DNS query reveals:

1. Who is requesting (yes, I know, the status of the source IP address is complicated. . .)
2. What is requested (the QNAME)

It may defeat, at least partially, some security measures (such as HTTP**S**)

QNAME is revealing

1. `www.political-party.example` ← Sensitive information
2. `_bittorrent-tracker._tcp.domain.example` ← MPAA may be interested
3. `le-pc-de-pascal.domain.example` ← Personal information
4. PGP keys in DNS (indexed by user's email, see DANE WG) ← More personal information

Who can listen?

1. Name servers (both recursors and authoritative) sysadmins. “Enablers” in RFC 6973 parlance.
2. Third-parties sniffing the cable

We need solutions for “on the wire” and “on the server”.

Two cases

May require different solutions

1. Client machine ↔ full resolver (no caching to protect you) (you talk only to a few resolvers)
2. Resolver ↔ auth. name server (some protection because of caching and relaying by the resolver) (needs scalability)

The two principles of privacy engineering

1. Send as little data as possible (RFC 6973, section 6.1)
2. Encrypt it

2) is necessary against third-party snoopers. 1) is necessary against PRISM (or similar) providers.

IETF Action

1. Problem statement discussed in the list dns-privacy
2. Many Internet-Drafts
3. One meeting of dnsop + one BoF in London, march 2014
4. No running code yet, except for DNSoverTLSoverTCP
5. No WG or official Internet-Drafts

Problem statement

1. `draft-bortzmeyer-dnsop-dns-privacy`

Solution? Minimizing the QNAME

1. No need to send the full QNAME to the authoritative name servers
2. Ask `NS com` to the root name servers instead of `A www.example.com`
3. Deployable unilaterally, conformant with RFCs, no change in protocol
4. Already discussed with the Unbound and BIND people. Seems realistic
5. Loss of data in the auth. name servers: a feature, not a bug
6. `draft-bortzmeyer-dns-qname-minimisation`

Solution? Encrypting data

1. [outside IETF] DNScurve/DNScrypt
2. IPsec (no enthusiasm)
3. "Confidential DNS" a DNS-specific encryption scheme `draft-wijngaards-dnsop-confidentialdns`
4. "Starting TLS over DNS" relies on the well-known TLS. Requires TCP and therefore persistent connections `draft-hzhwm-start-tls-for-dns` Implemented and lot of data available `ftp://ftp.isi.edu/isi-pubs/tr-688.pdf`
5. DNSoD "DNS over DTLS" relies on the existing DTLS protocol `draft-wing-dnsop-dnsodtls`

(Provisional) Conclusion

1. No overwhelming enthusiasm (or opposition)
2. Changing the DNS is hard

Merci !

www.afnic.fr
contact@afnic.fr