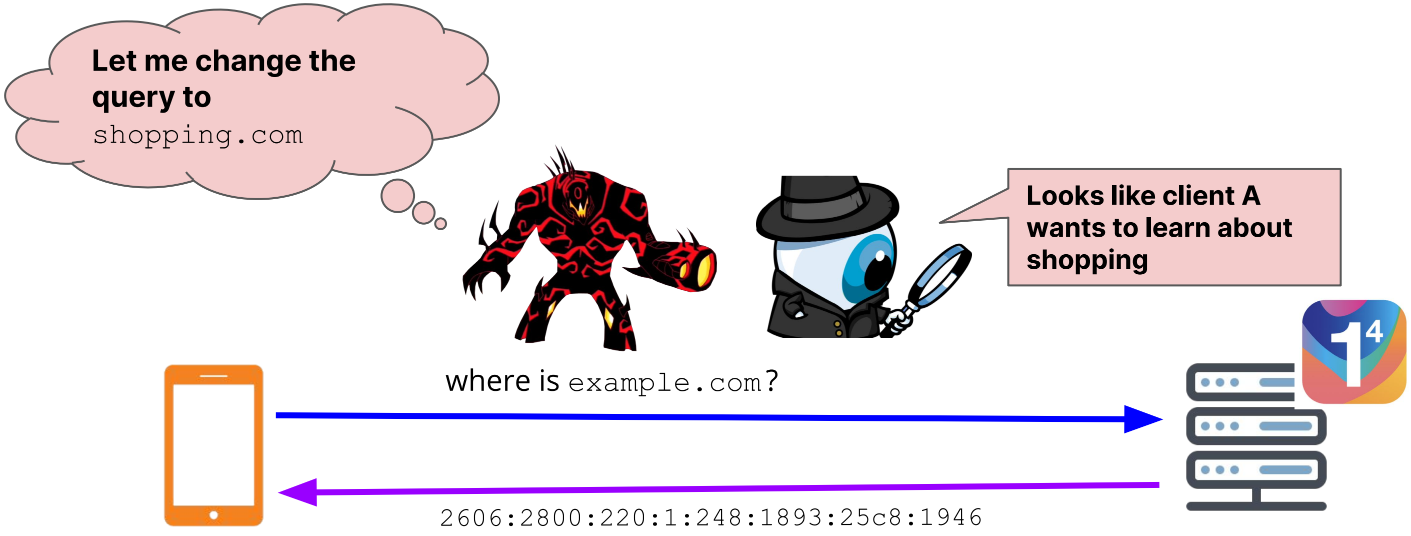
Presented at Proceedings on Privacy Enhancing
Technologies Symposium - 2021

CLOUDFLARE

Oblivious DNS Over HTTPS (ODoH): A Practical Privacy Enhancement to DNS

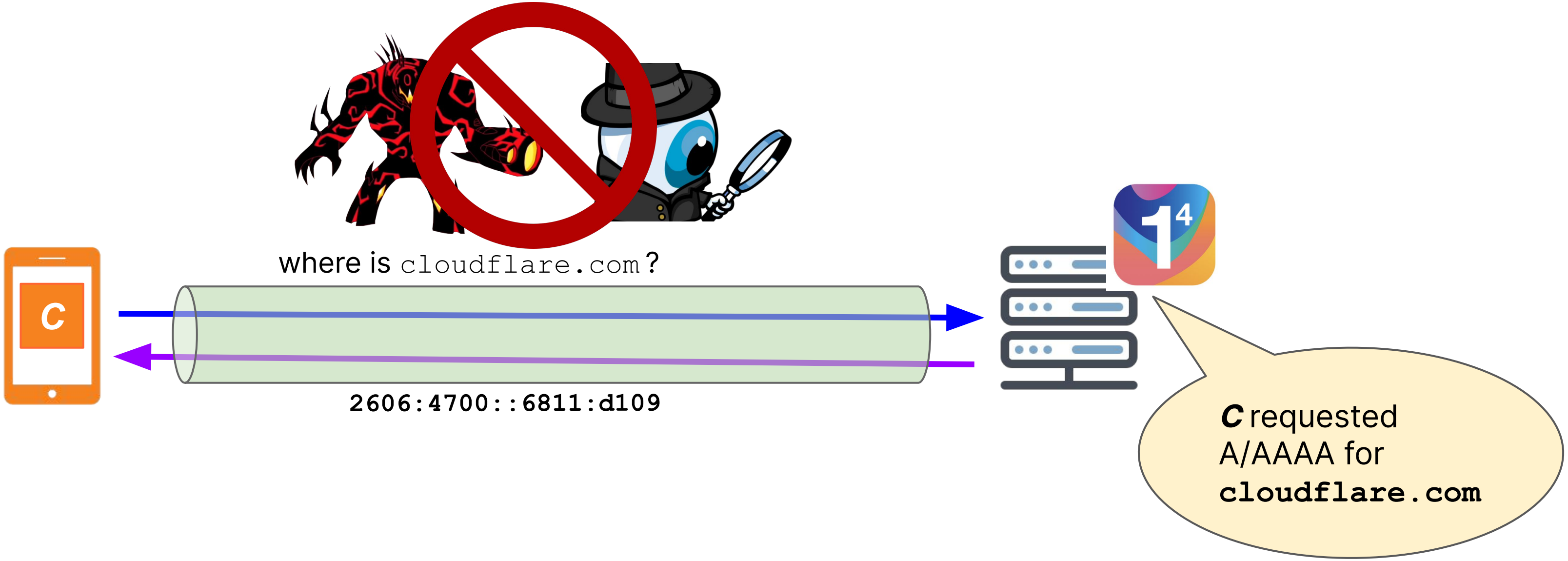
Sudheesh Singanamalla
Suphanat Chunhapanya
Jonathan Hoyland
Marek Vavruša
Tanya Verma
Peter Wu
Marwan Fayed
Kurtis Heimerl
Nick Sullivan
Christopher Wood

Do53: Plain-text UDP exposes DNS Messages

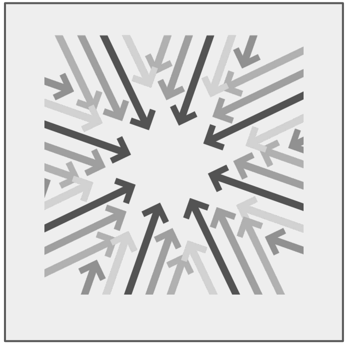


Most Widely Used Variant of the Protocol
(92% daily traffic to *1.1.1.1*)

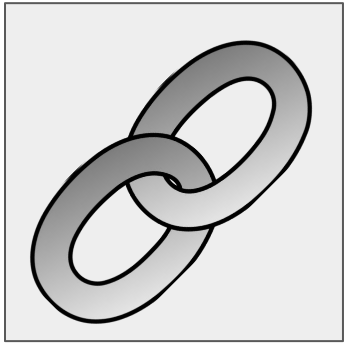
DoH: Encrypts Stub-to-Resolver Link



The Gaps in DoH that ODoH Fills



Centralization of Services

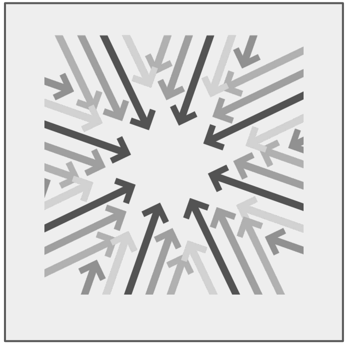


Association of query to clients

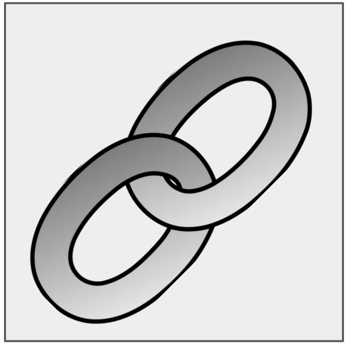
Privacy by Policy

Regulatory Concerns

The Gaps in DoH that ODoH Fills



Centralization of Services

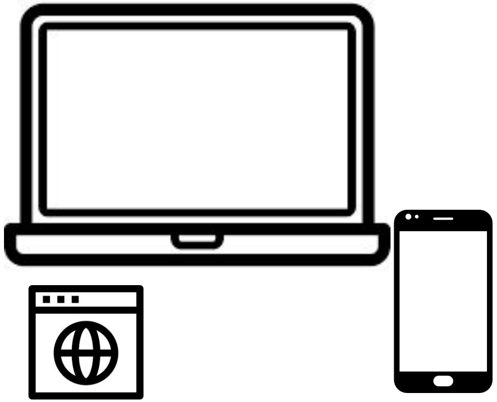


Association of query to clients

Privacy by Policy

Regulatory Concerns

Components of ODoH (1/3)

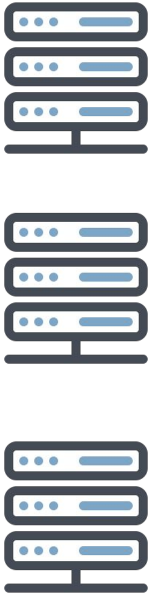


- Prepare DNS Query requests
- Receive DNS Answer responses

Goals:

1. Be able to successfully encrypt and decrypt DNS messages
2. Be unable to decrypt incorrectly received messages.
3. Identify maliciousness or attacks when they occur.

Components of ODoH (2/3)

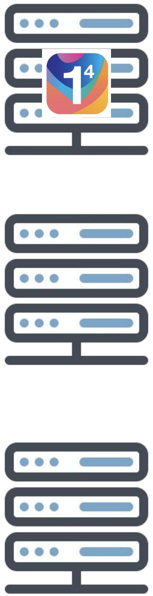


- Relay the encrypted requests to target
- Relay the encrypted responses to client
- Remove client IP addresses

Goals:

1. Remove client identifying information
2. Be unable to decrypt any messages from either the client or the target instances
3. Operated by an organization different from the target resolver

Components of ODoH (3/3)



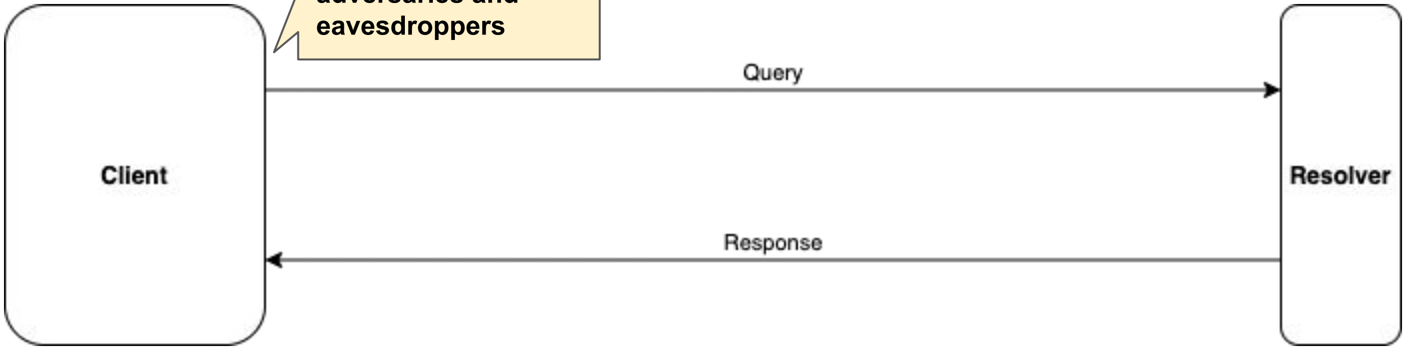
- Receive the encrypted requests from proxy
- Decrypt the query and Encrypt the answer

Goals:

1. Successfully decrypt the query
2. Obtain the answer from a resolver
3. Encrypt the answer and respond to proxy
4. Be unable to identify the actual client requesting the information.

Building the ODoH Protocol - Starting at DoH

DNS Query is secure from Network adversaries and eavesdroppers



Network Security ✅
Query Privacy ❌

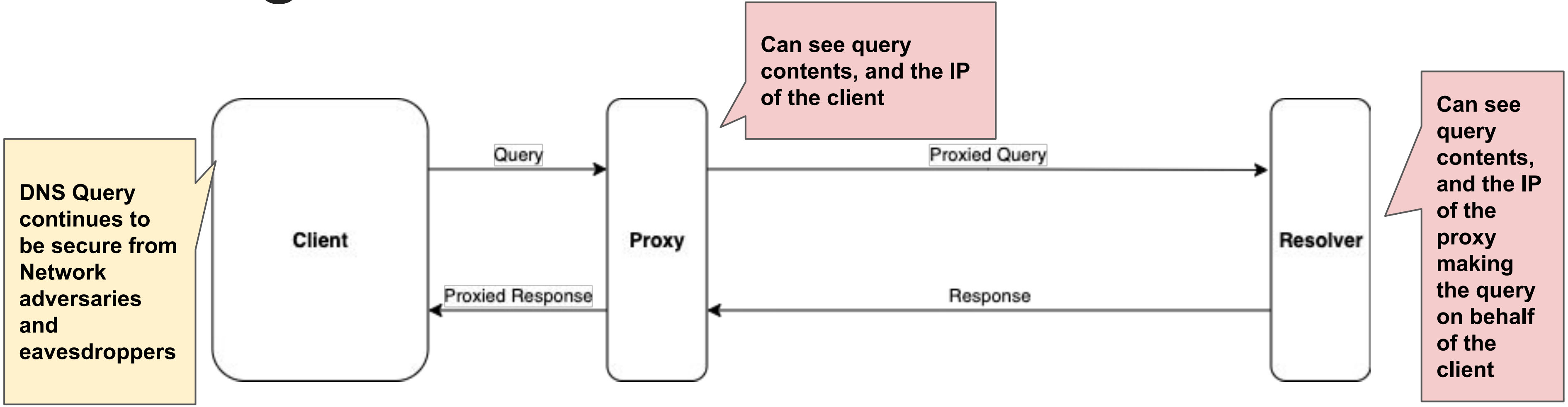
Client IP Address ✅
Client Query ✅

Can see query contents, and the client IP making the query.

Similar to DoH and uses an encrypted communication channel

CLOUDFLARE

Building the ODoH Protocol - Proxied DoH

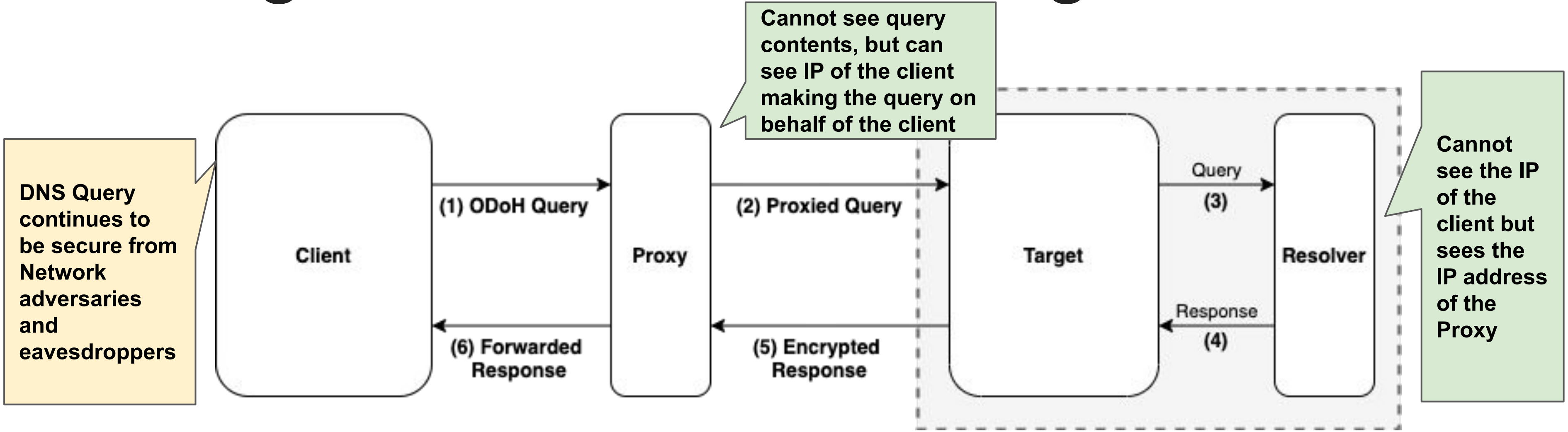


Network Security ✅
Query Privacy ❌

Client IP Address ✅
Client Query ✅

Client IP Address ❌
Client Query ✅

Building the ODoH Protocol - High Level View



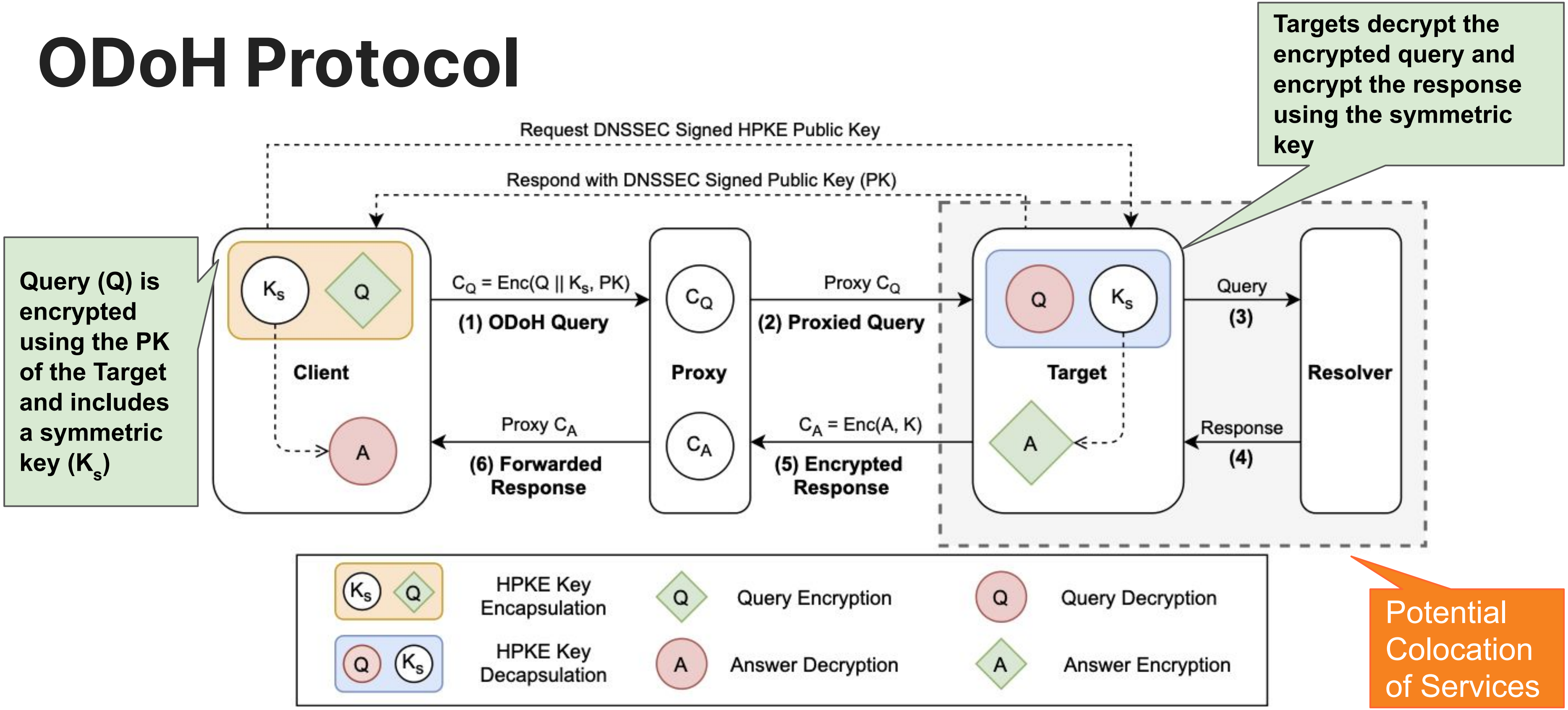
Network Security ✅
Query Privacy ✅

Client IP Address ✅
Client Query ❌

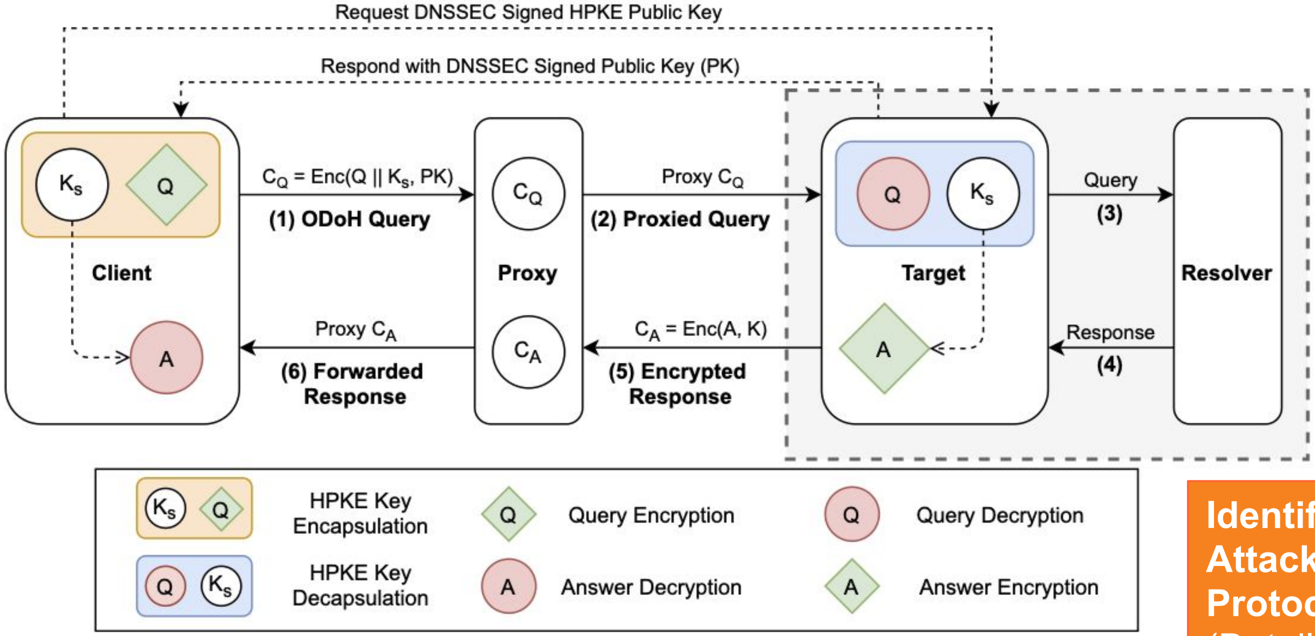
Client IP Address ❌
Client Query ✅

Requirements: Proxy and Target are Non-Colluding

ODoH Protocol



Formal Analysis

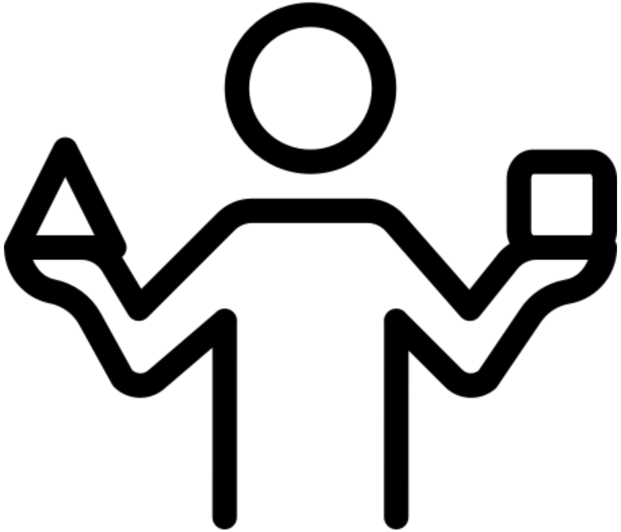


Lemma: An adversary is unable to associate a connection between client and proxy with the corresponding query unless both the proxy and target are compromised.

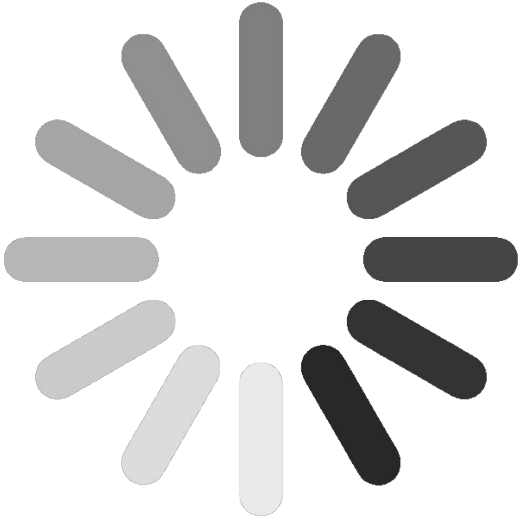
Identified and Fixed a Replay Attack in the IETF proposed ODoH Protocol
(Details in the paper)

Research Questions

What is the impact of ODoH on **DNS Response Times?**



How does ODoH **Compare to other privacy enhancing protocols?**



How does ODoH affect **Page Load Time experiences?**

CLOUDFLARE

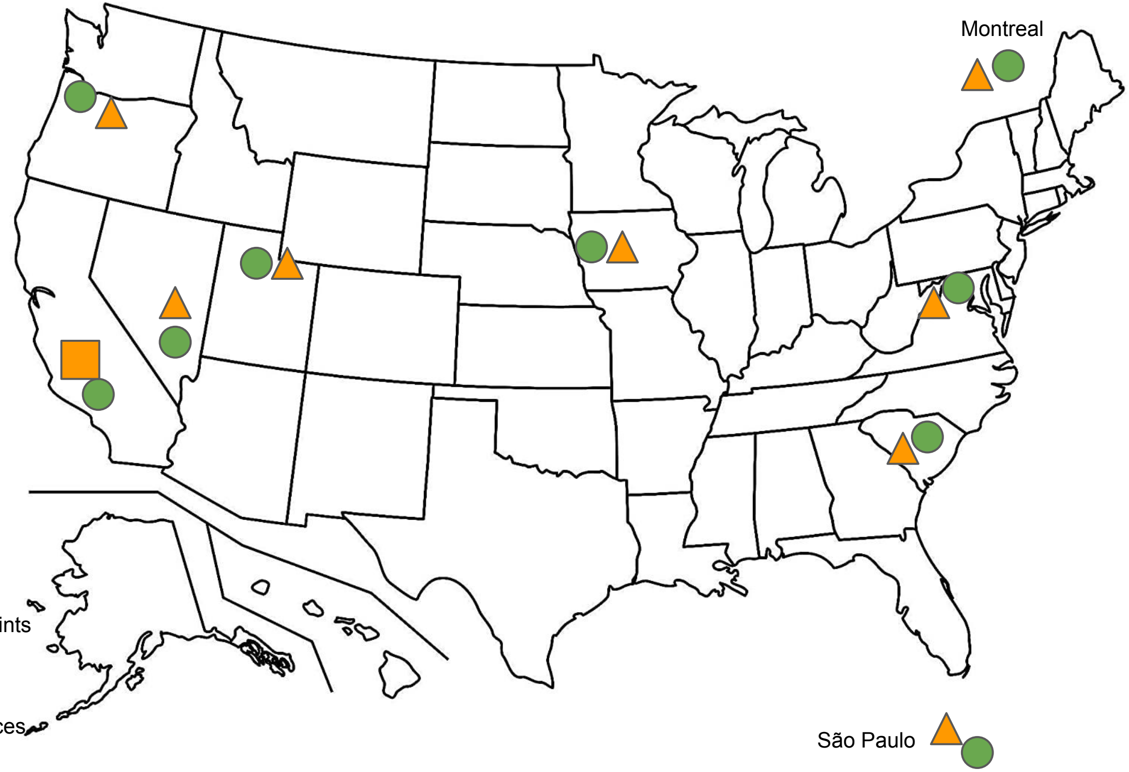
Measurement Setup and Deployments

Resolvers:

1.1.1.1

8.8.8.8

9.9.9.9



Client Vantage Points

GCP Instances

Serverless Instances

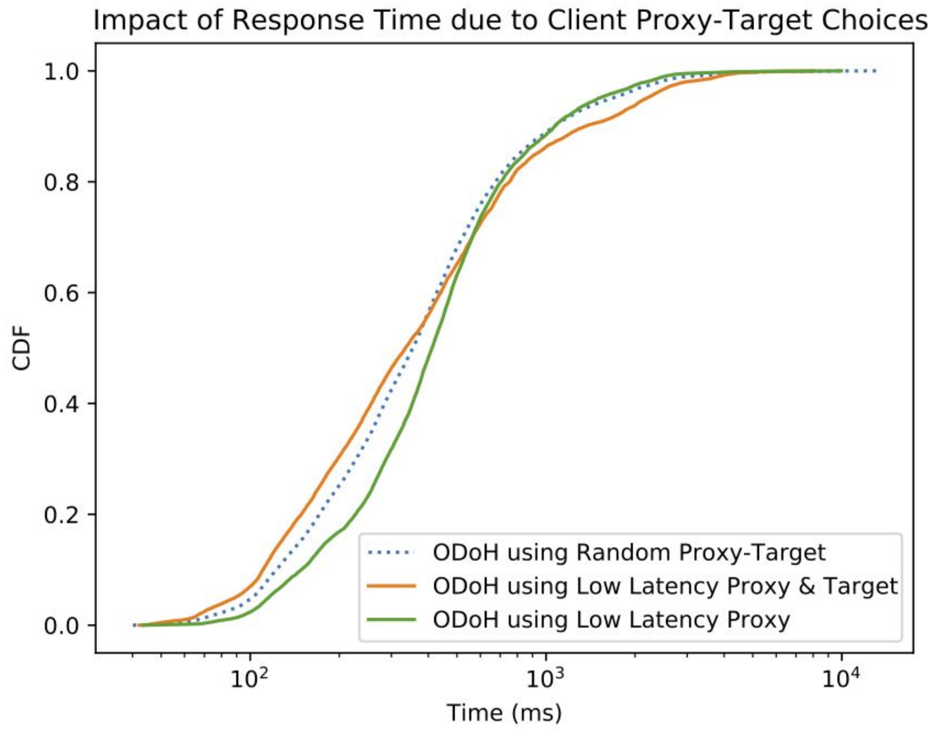
90 Client stubs
- 10 per vantage point

Experiment:
21,000 DNS req/day
or 15 requests/minute

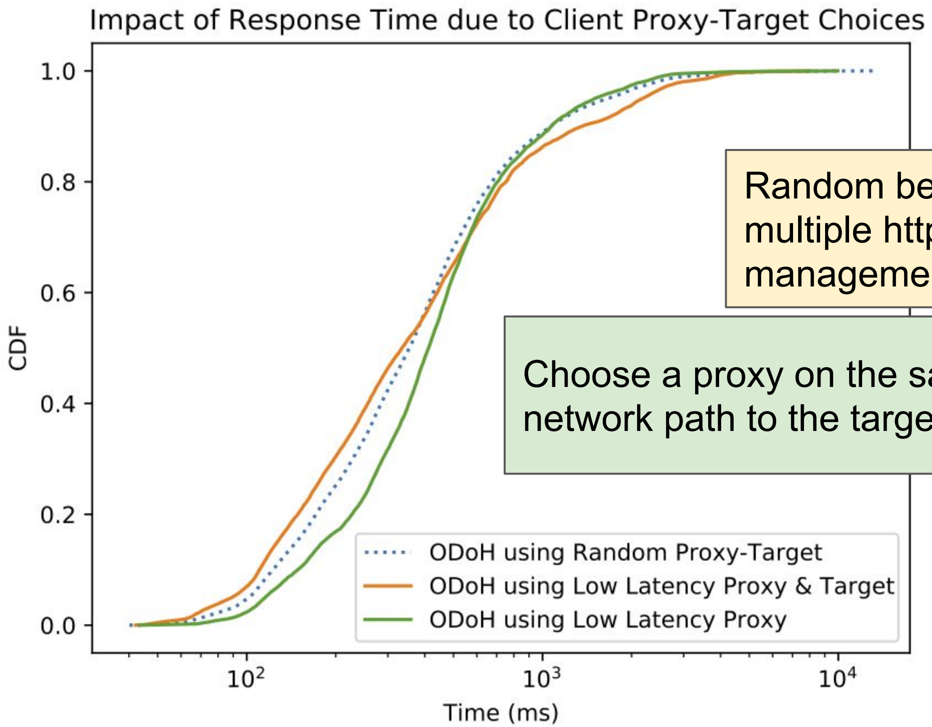
Average bandwidth:
480 Mbit/s

Clients:
1 core Intel Xeon 2 GHz CPU 3.75GB RAM x86_64

Takeaway 1: Choose Low-latency Proxy-Target



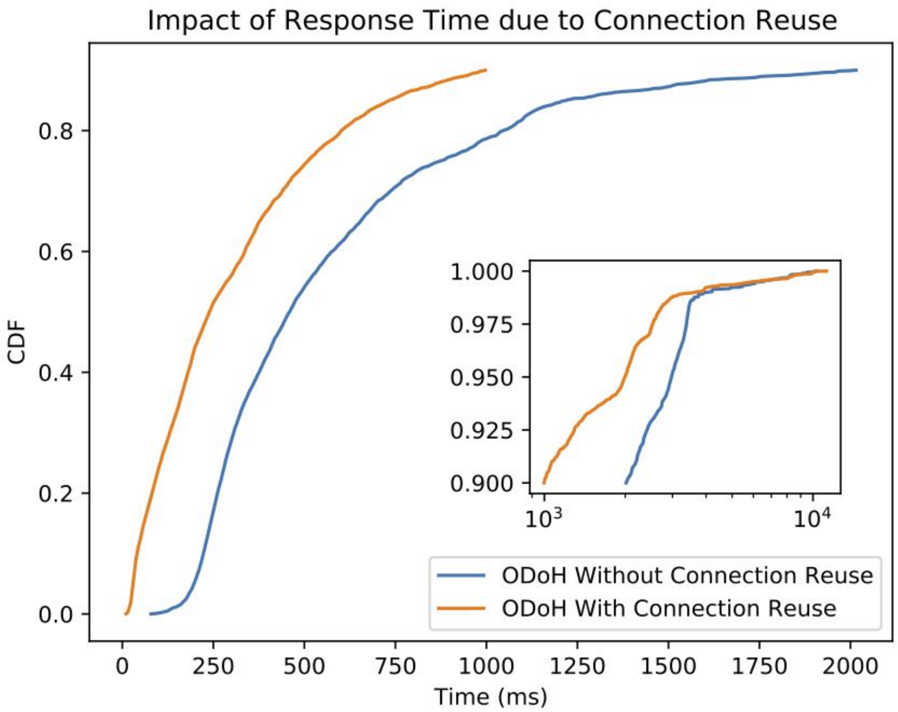
Takeaway 1: Choose Low-latency Proxy-Target



Random behaves better due to multiple http connection management of streams.

Choose a proxy on the same network path to the target.

Takeaway 2: Reuse Connections

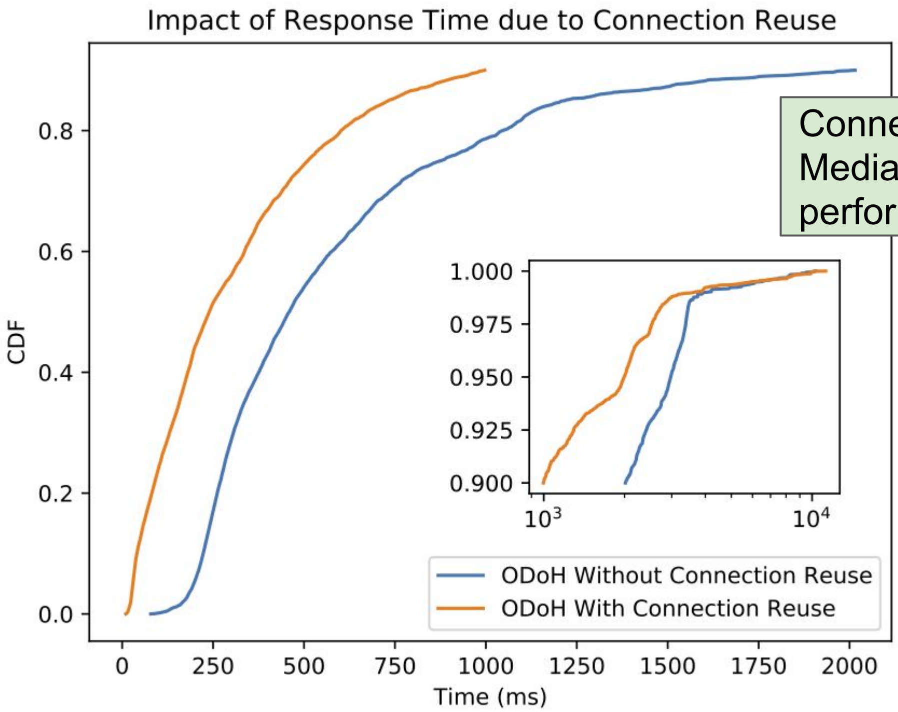


Takeaway 2: Reuse Connections

Some leakage of client identity due to reuse of session keys

- **No sensitive information** in either cleartext or encrypted form **is leaked**

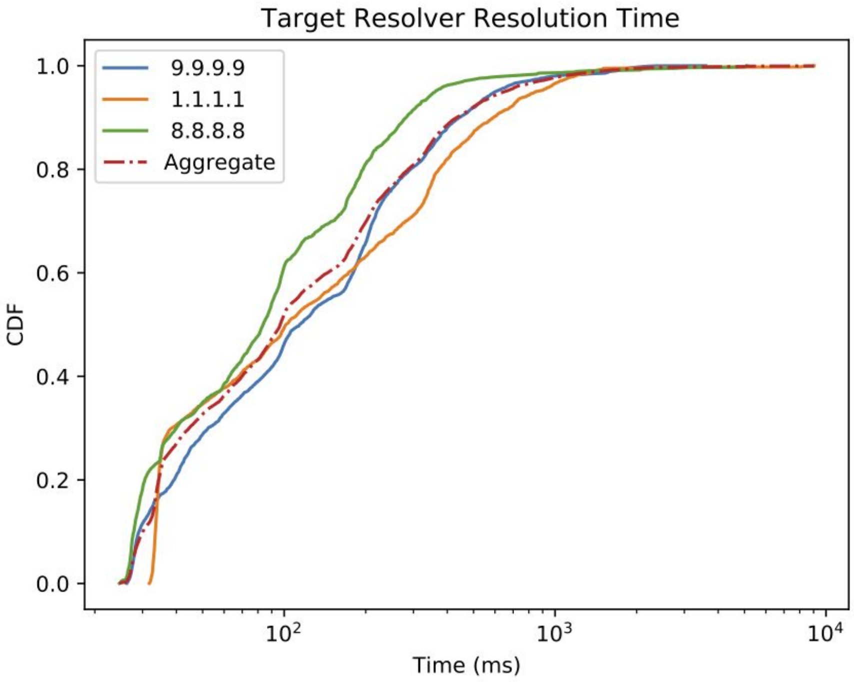
Possible for clients to configure and force new connections if necessary.



Connection reuse improves Median DNS response time performance by 48%.

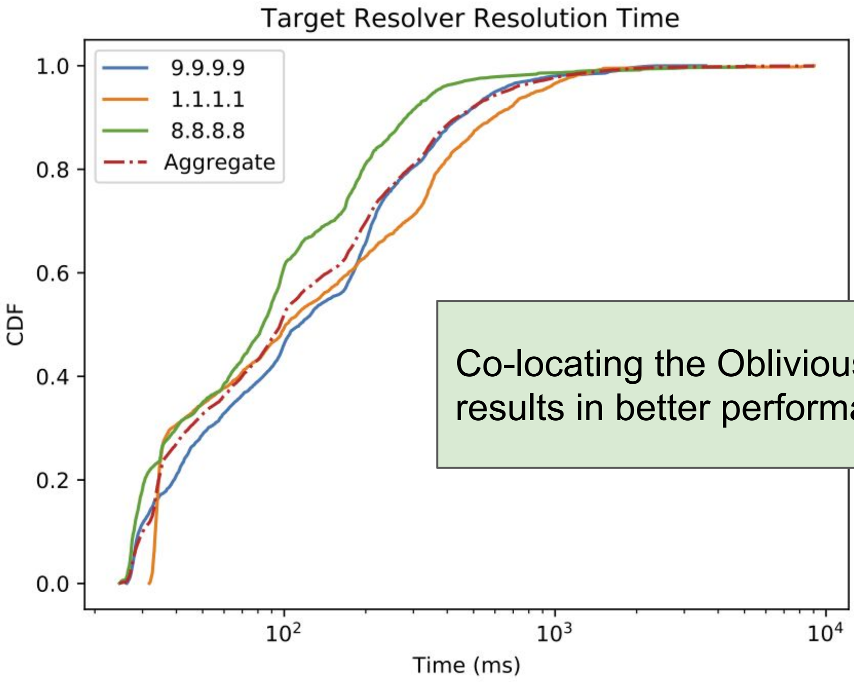
Takeaway 3: Colocation is Important

The targets hosted on Google Cloud have faster response times with Google DNS due to colocation of Google DNS within Google Cloud Services.



Takeaway 3: Colocation is Important

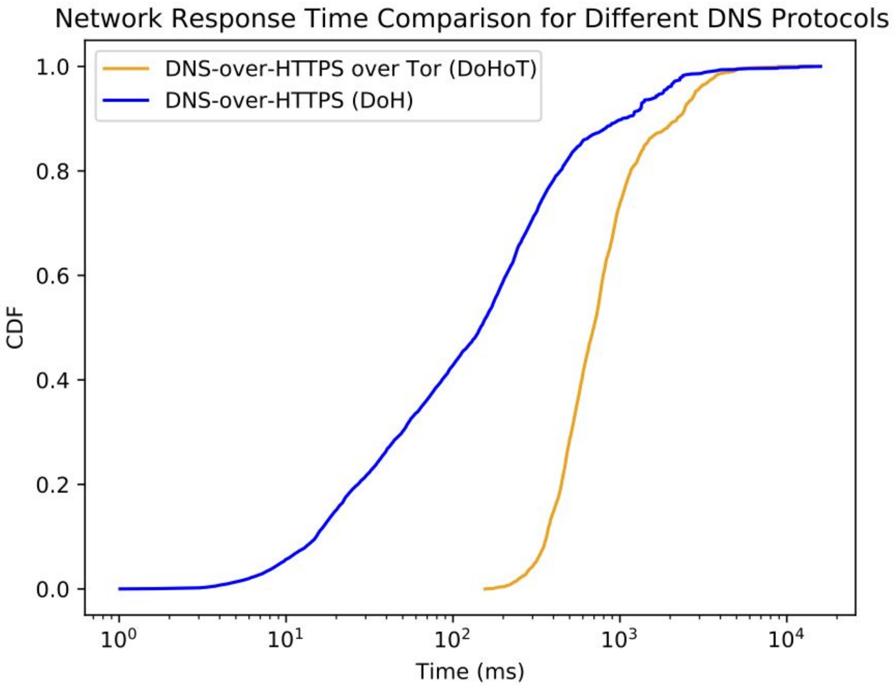
The targets hosted on Google Cloud have faster response times with Google DNS due to colocation of Google DNS within Google Cloud Services.



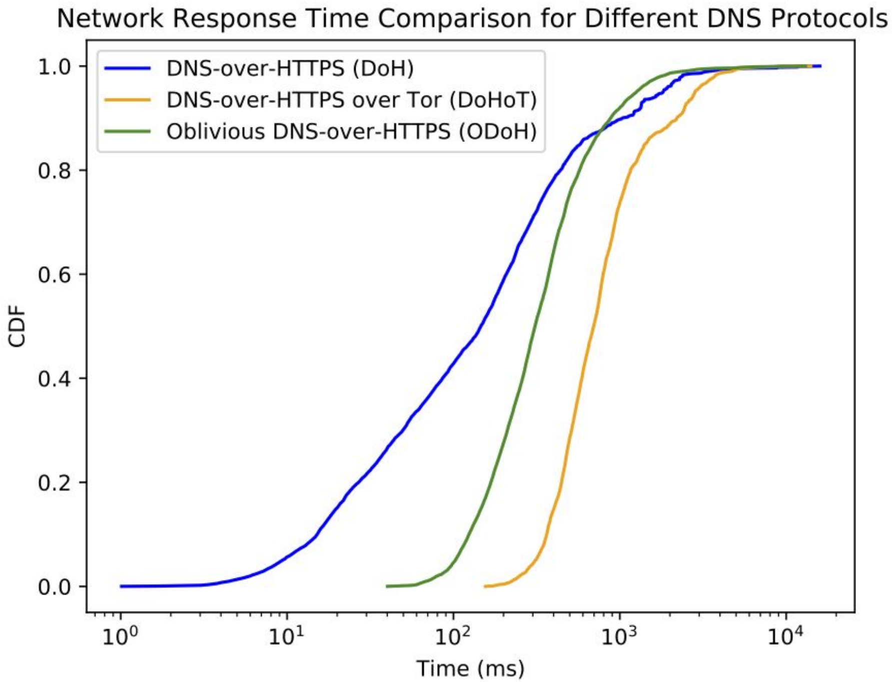
Co-locating the Oblivious Target and the Resolver results in better performance.

Engineered into:
https://odoh.cloudflare-dns.com

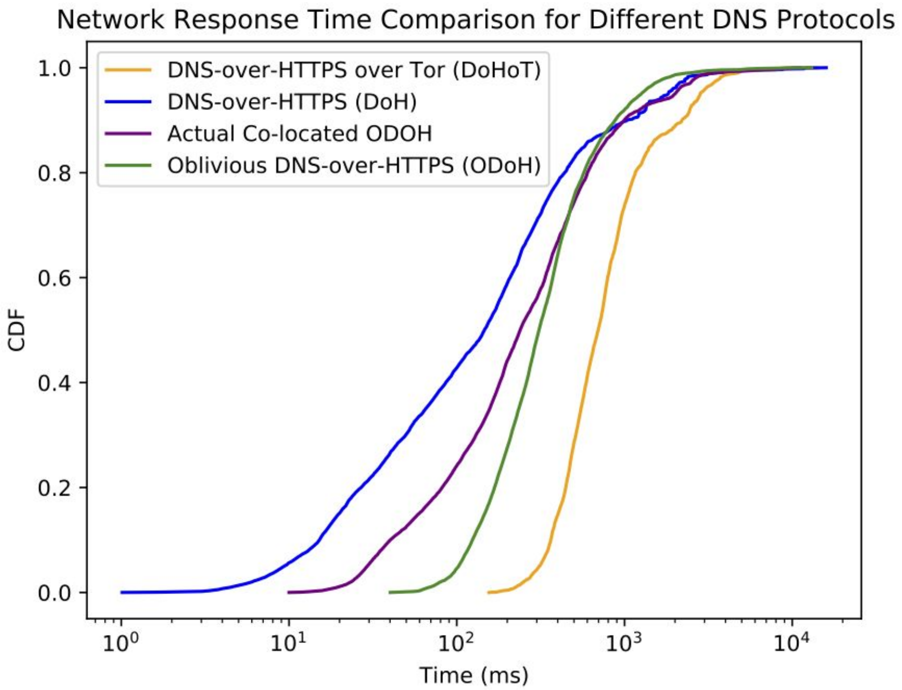
Comparing ODoH with Other DNS Protocols



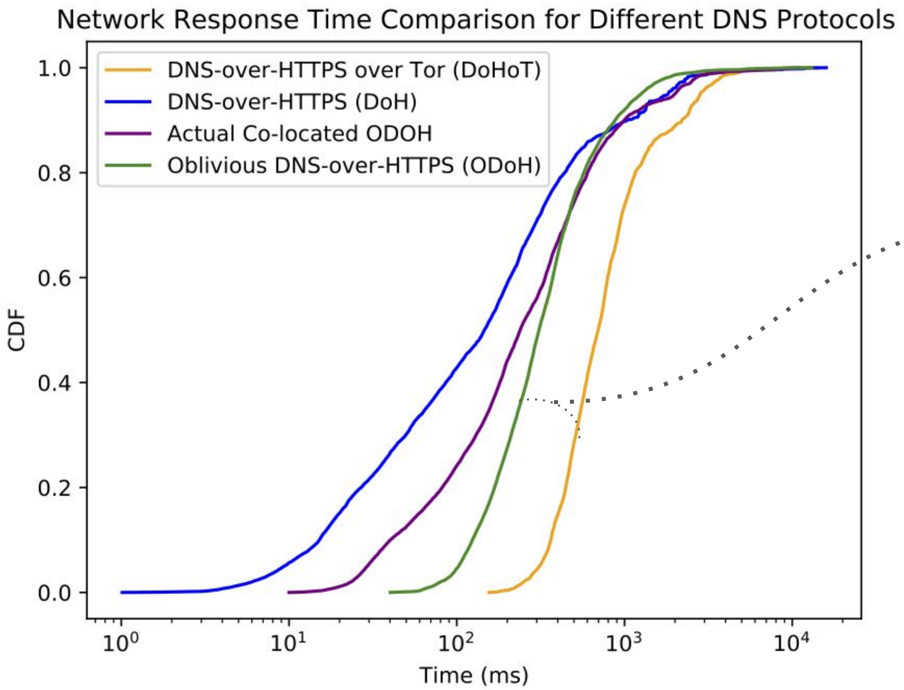
Comparing ODoH with Other DNS Protocols



Comparing ODoH with Other DNS Protocols



Comparing ODoH with Other DNS Protocols

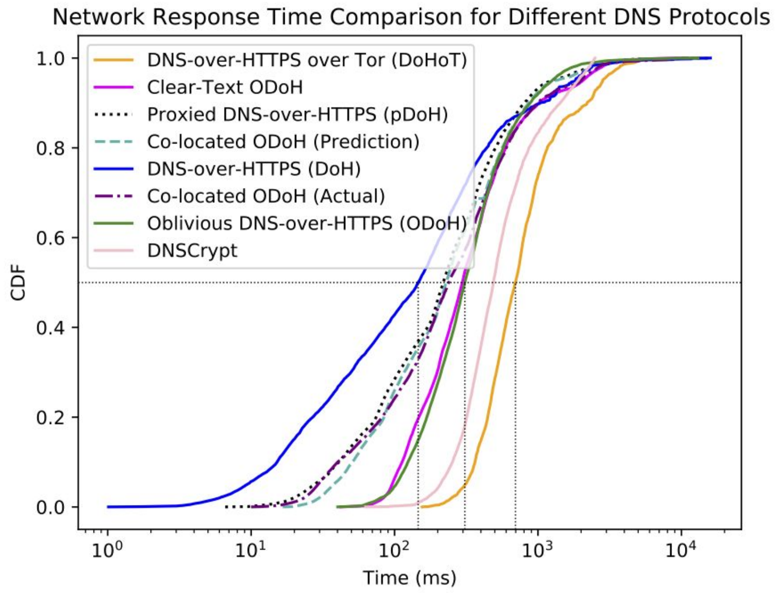


Encrypted DNS Protocols:

DNSCrypt and **Anonymous DNSCrypt**

CLOUDFLARE

Comparing Other Architectural Variants

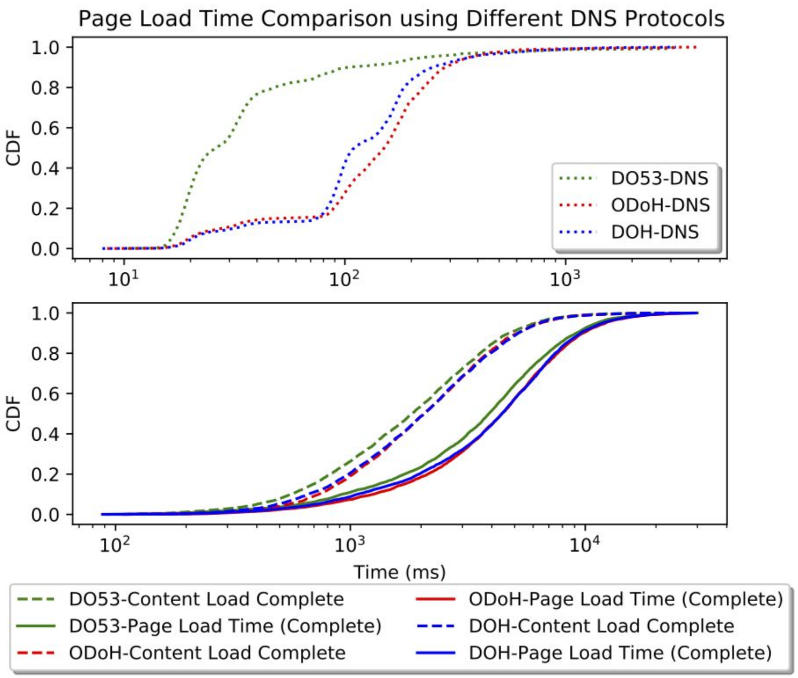


https://odoh.cloudflare-dns.com/

Protocol	Request Path	Security	Privacy
Plain DNS (Do53)	C → R	No	No
DNS over HTTPS (DoH)	C → R	Yes	No*
Proxied DoH	C → P → R	Yes	No
Oblivious DoH (ODoH)	**C → P → T → R**	**Yes**	**Yes**
Cleartext ODoH	C → P → T → R	Yes	No
Co-located ODoH	**C → P → (T+R)**	**Yes**	**Yes**
DNSCrypt	C → R	Yes	No*
Anonymous DNSCrypt	C → P → R	Yes	Yes
DoH over Tor (DoHoT)	C → Tor → R	Yes	Yes

[C: Client, R: Resolver, T: Target, P: Proxy] * Privacy Policy Based Privacy

In Browser Measurements



Measurements taken from a single vantage point (Chrome using Local Stub resolver[1]):

- Client node in a lab university wireless network (200 Mbps DL / 8Mbps UL)
- Experimental setup with on-path proxy
- 5000 random and top chosen websites from the Top 1M in Tranco dataset
- PLT taken after entire navigation page is rendered

Median Page load times increase by ~6.7% when using DoH and ~9.8% when using co-located ODoH services.

[1] https://github.com/cloudflare/cloudflared

Summary and Conclusion

1. Performance impacts in the protocol are **purely network topology effects**.
2. **Service co-location** will result in **increased response time performance**.
3. Client **choosing on-path proxy** results in higher response time performance.
4. Clients are encouraged to **reuse https connections** to avoid TLS+TCP handshake overheads.
5. ODoH has minimal total page load time impacts or perceivable user experience impacts.
6. **ODoH is a practical privacy enhancing protocol for DNS.**

Artifacts and Services Available

ODoH Rust Client	https://github.com/cloudflare/odoh-client-rs
ODoH Go Client	https://github.com/cloudflare/odoh-client-go
ODoH Go Target	https://github.com/cloudflare/odoh-server-go
ODoH Go Proxy	https://github.com/cloudflare/odoh-server-go
ODoH Rust	https://github.com/cloudflare/odoh-rs
ODoH Go	https://github.com/cloudflare/odoh-go
Production ODoH Target	https://odoh.cloudflare-dns.com/
Production ODoH Proxy	https://odoh1.surfdomeinen.nl/

Thank you

https://blog.cloudflare.com/oblivious-dns/

Paper: https://petsymposium.org/2021/files/papers/issue4/popets-2021-0085.pdf

A special shoutout to:

Eric Kinnear
Tommy Pauly
Wesley Evans
Patrick McManus
Edo Ryker
Alissa Starzak
John Graham-Cumming
Anbang Wen
Joost Van Dijk
Stephen Spencer
Tobias Pulls
IETF, IRTF, and DNS OARC Communities