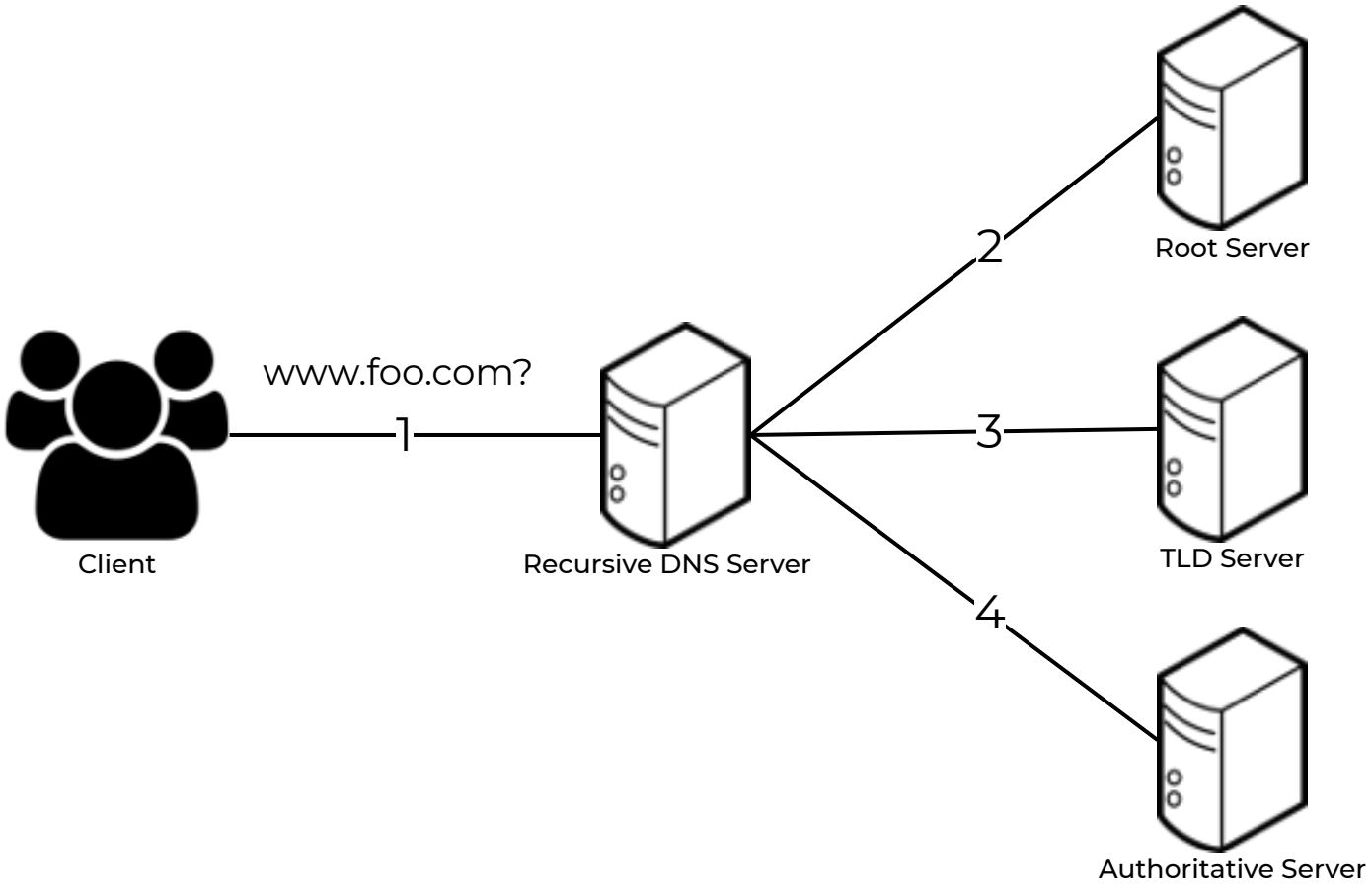
Oblivious DNS: Practical Privacy for DNS Queries

Paul Schmitt (Princeton)
Anne Edmundson (Princeton)
Allison Mankin (Salesforce)
Nick Feamster (Princeton)

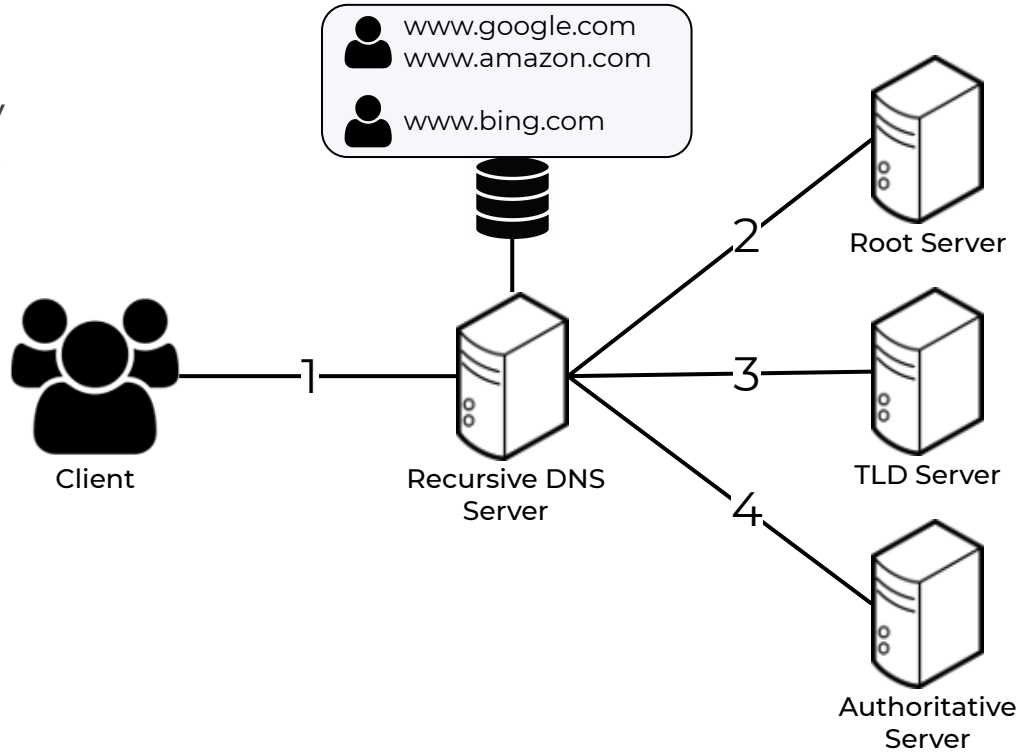
PRINCETON
UNIVERSITY

Conventional DNS



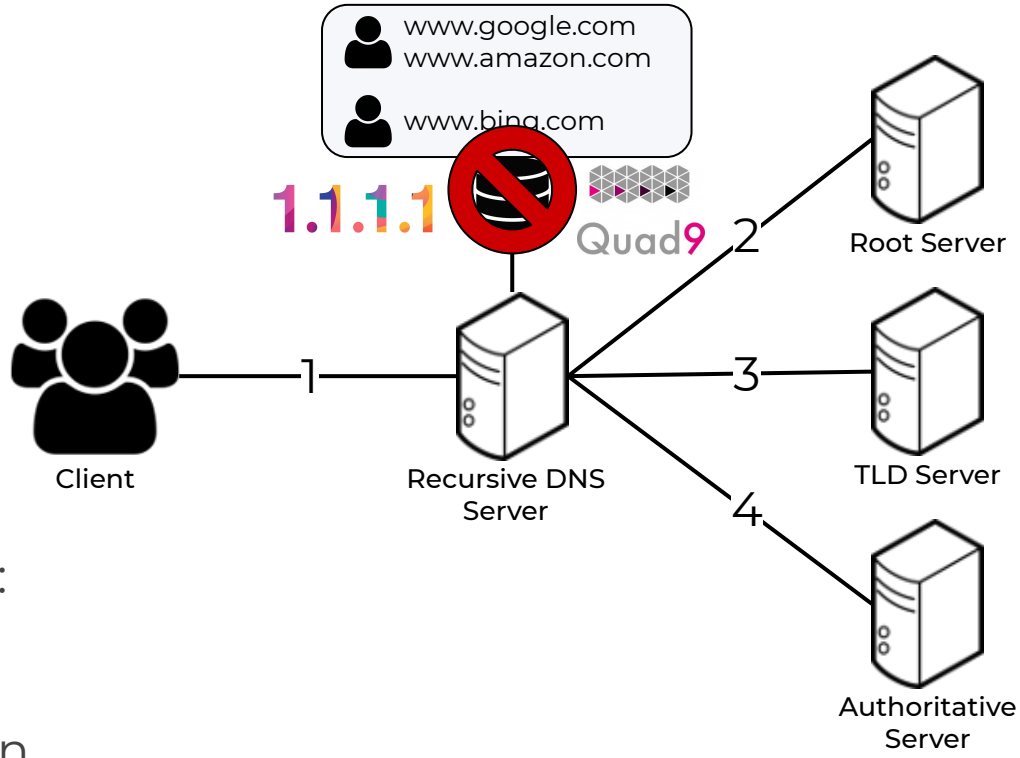
Conventional DNS

- Client identity and query are viewable at and prior to the recursive (ISP) server
- DNS operators can be targets of data requests



Conventional DNS

- Services now offer open DNS resolvers with promise of deleting logs
- Shifts trust to these providers
- Other techniques do not fully protect user privacy:
  - DNS-over-TLS
  - DNS-over-HTTPS
  - QNAME minimization



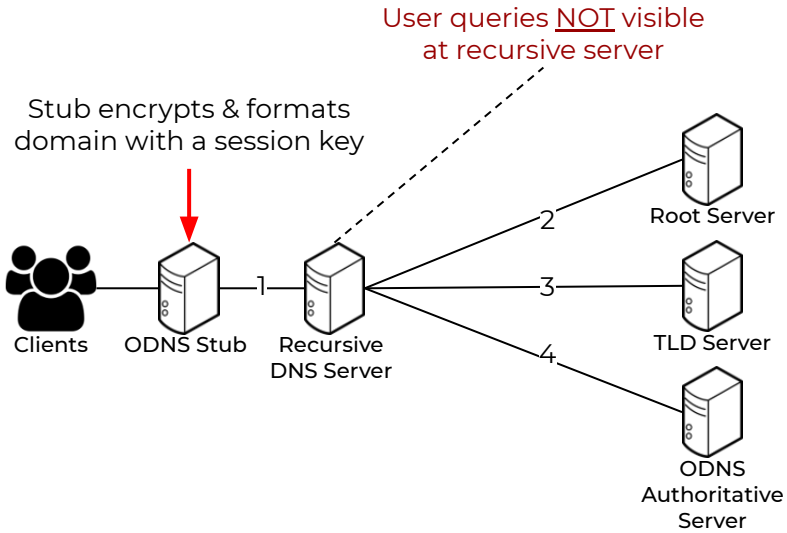
Oblivious DNS

Goal:

- Separate user identity from query

Requirements:

- Compatible with existing infrastructure
- Minimize overhead



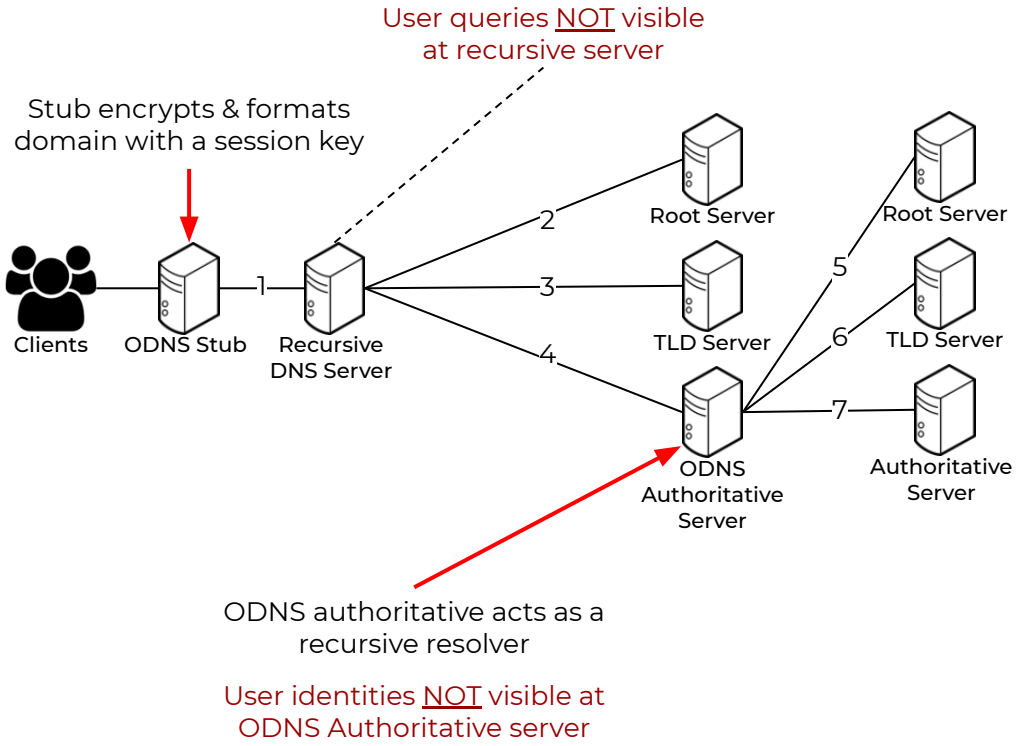
Oblivious DNS

Goal:

- Separate user identity from query

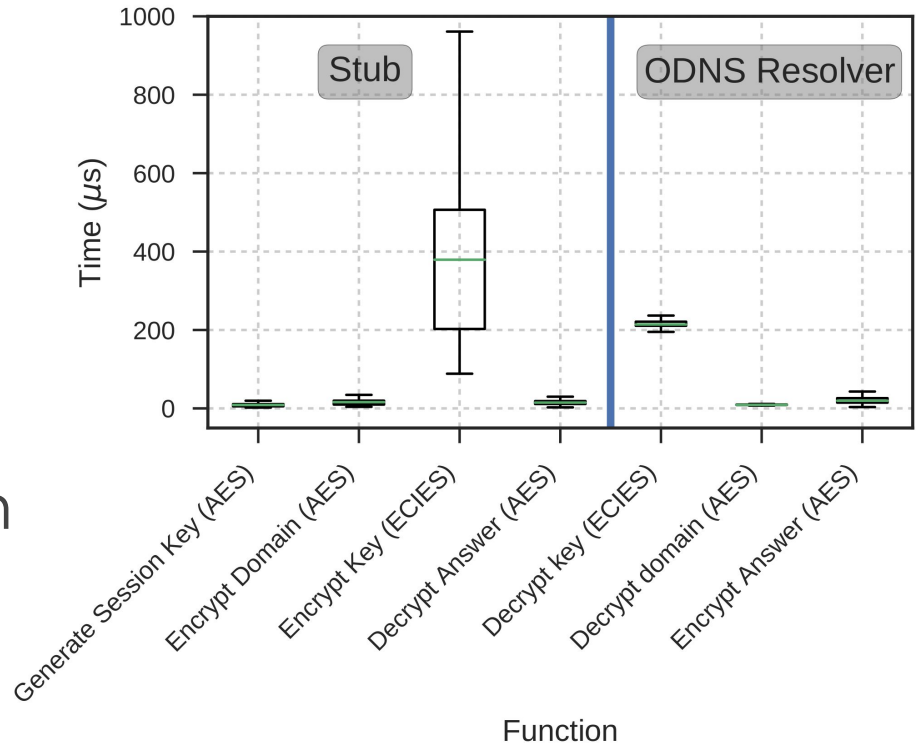
Requirements:

- Compatible with existing infrastructure
- Minimize overhead



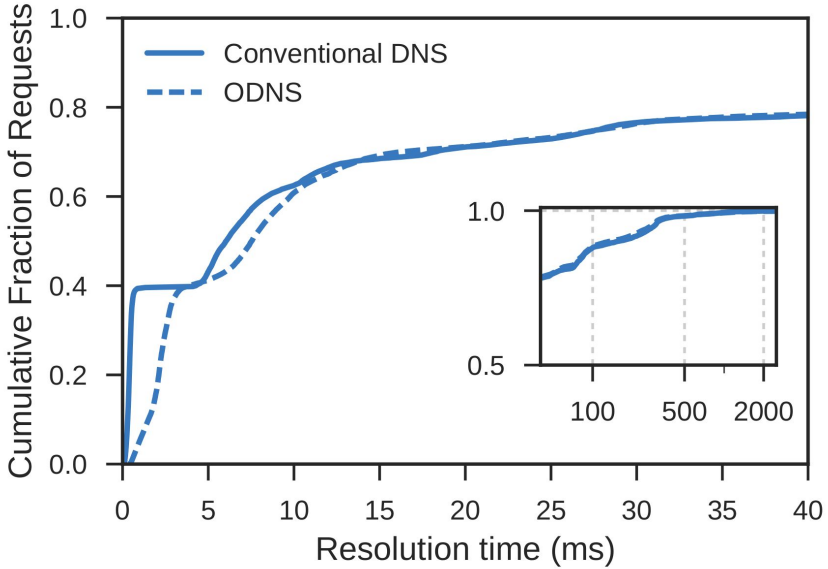
ODNS Crypto Overhead

- Roughly ~1-2 ms for crypto operations using standard libraries
- Symmetric encryption/decryption is lightweight



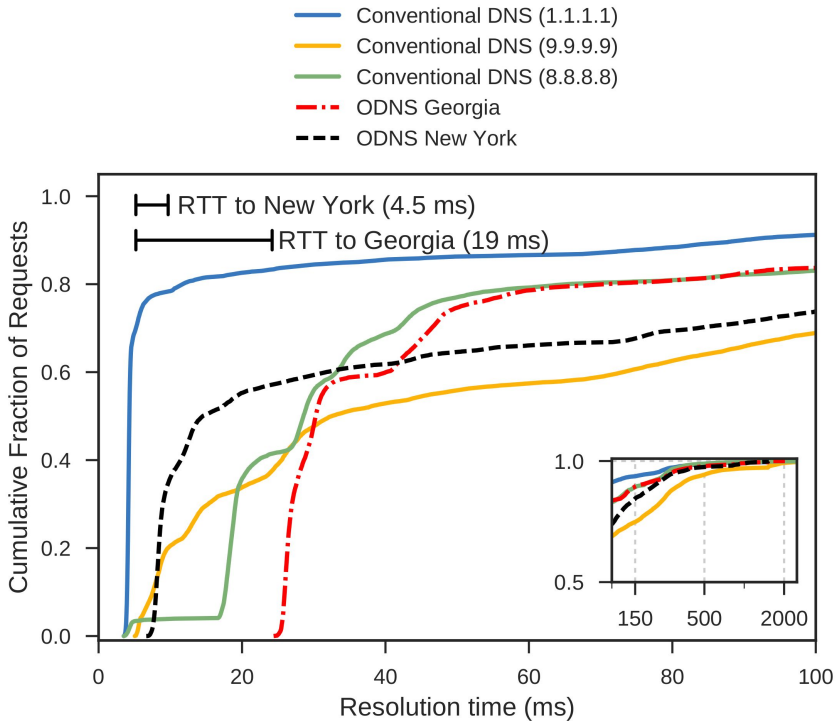
ODNS Crypto Overhead

- Roughly ~1-2 ms for crypto operations using standard libraries
- Symmetric encryption/decryption is lightweight



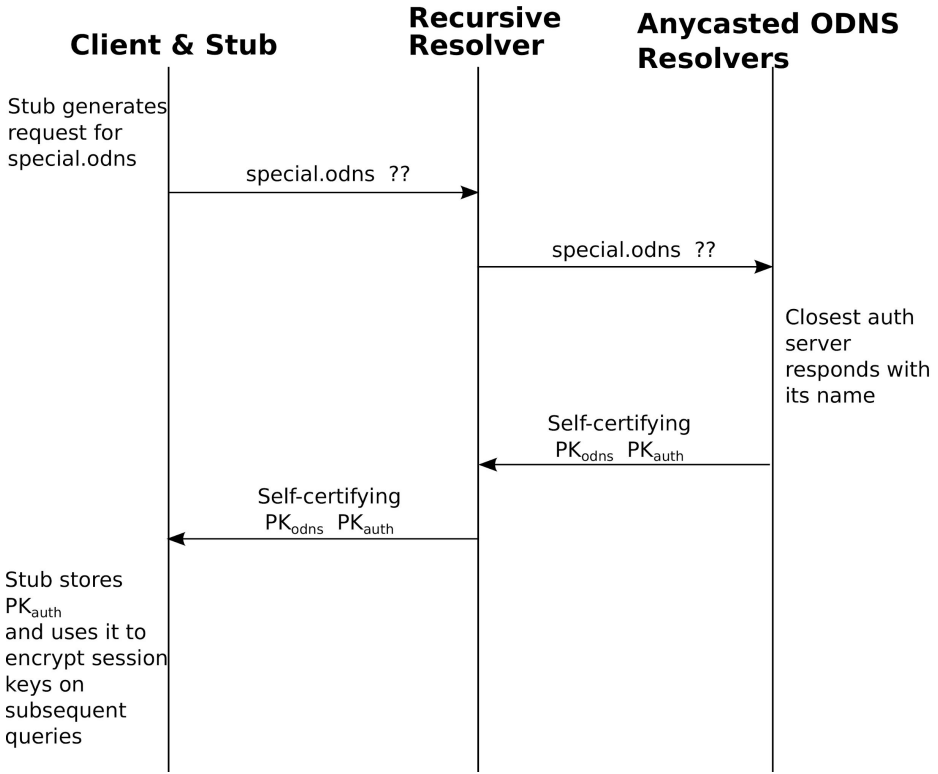
ODNS WAN Latency

- Latency to ODNS Resolver added to each query
- Widespread anycast deployment to mitigate WAN latency

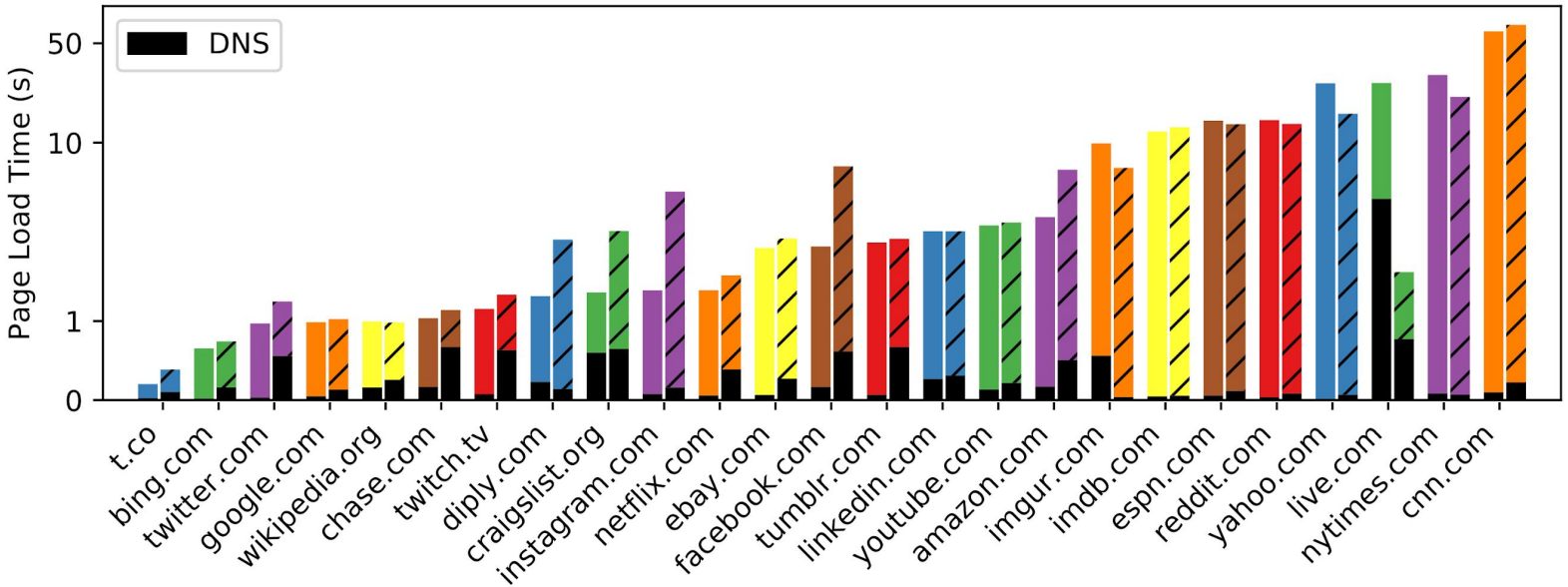


Key Distribution

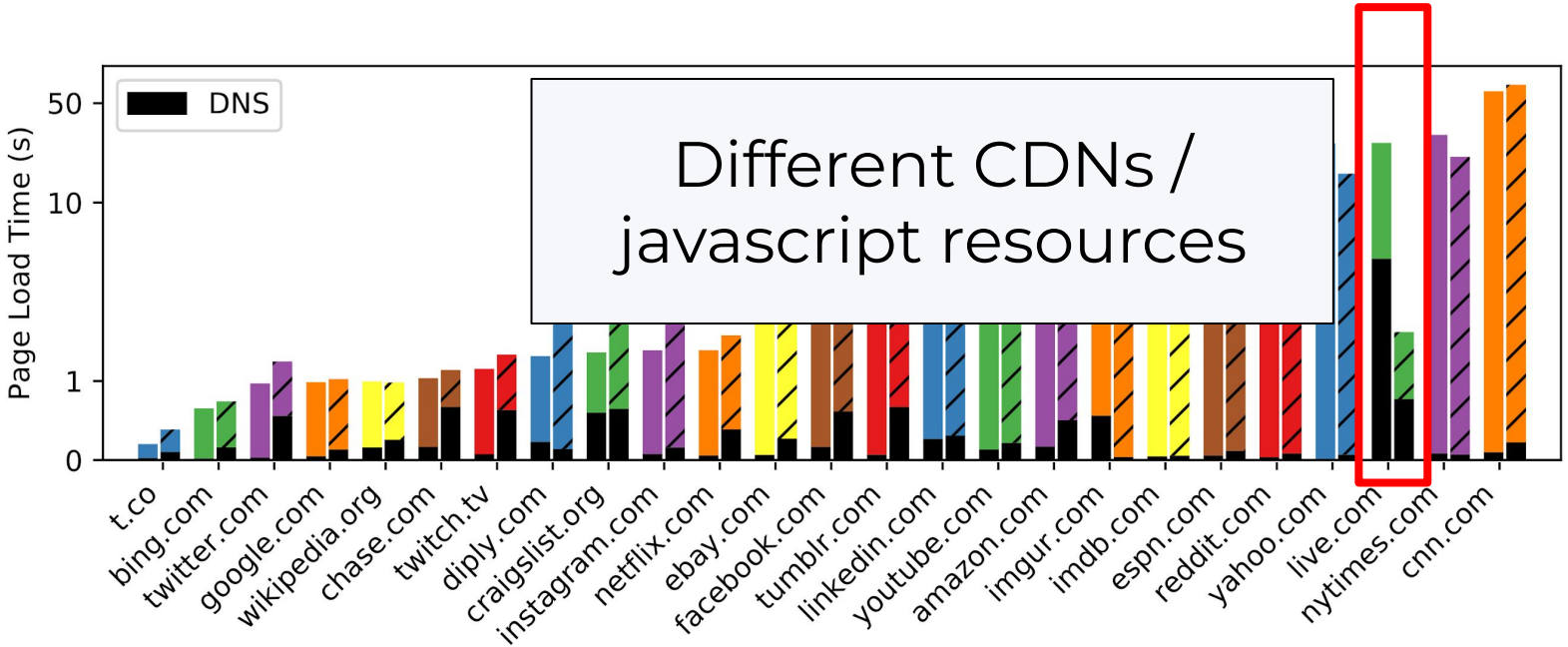
- Anycast for scalability
- Special query reaches the nearest anycast server
- Server responds with public key and name



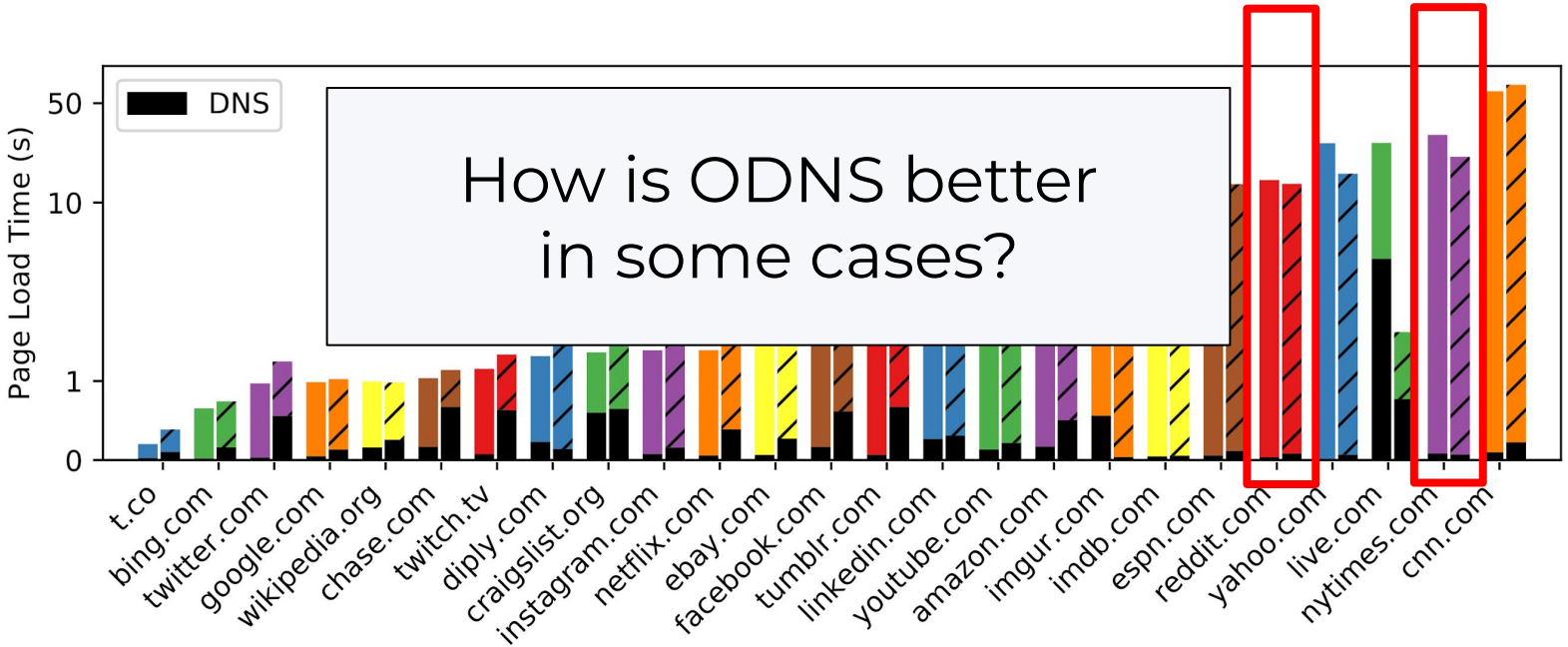
ODNS Overhead: Page Load Time



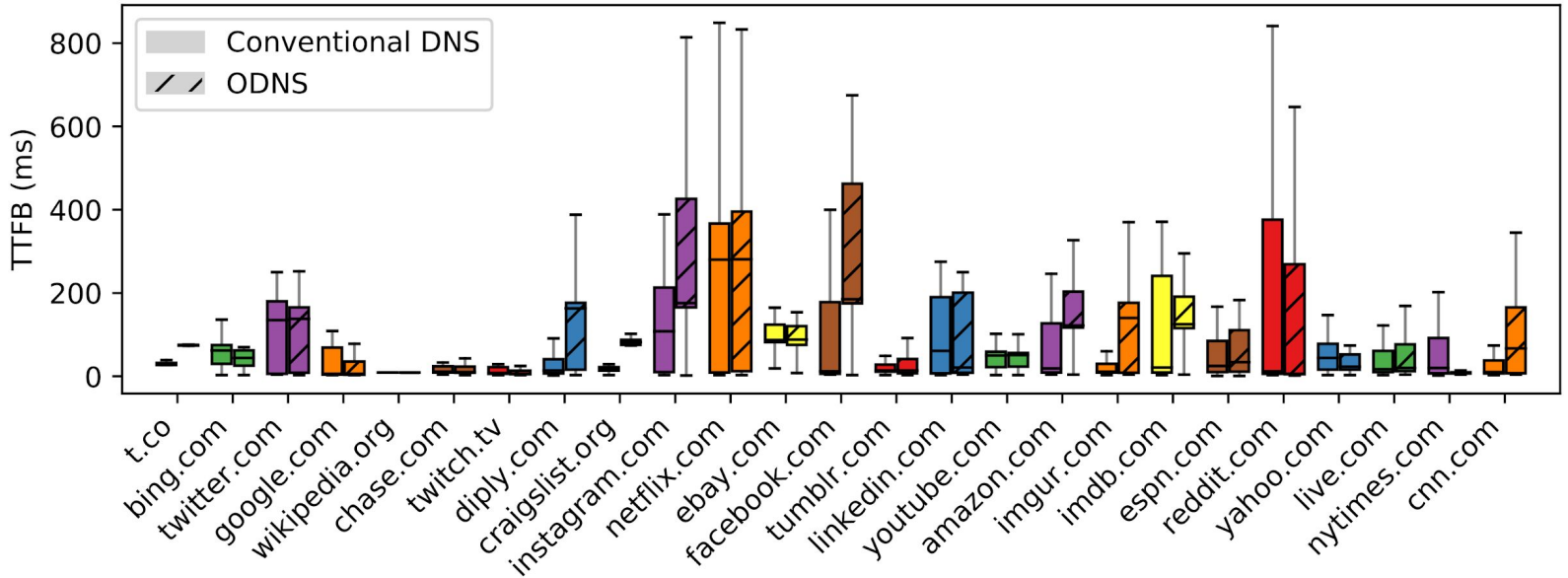
ODNS Overhead: Page Load Time



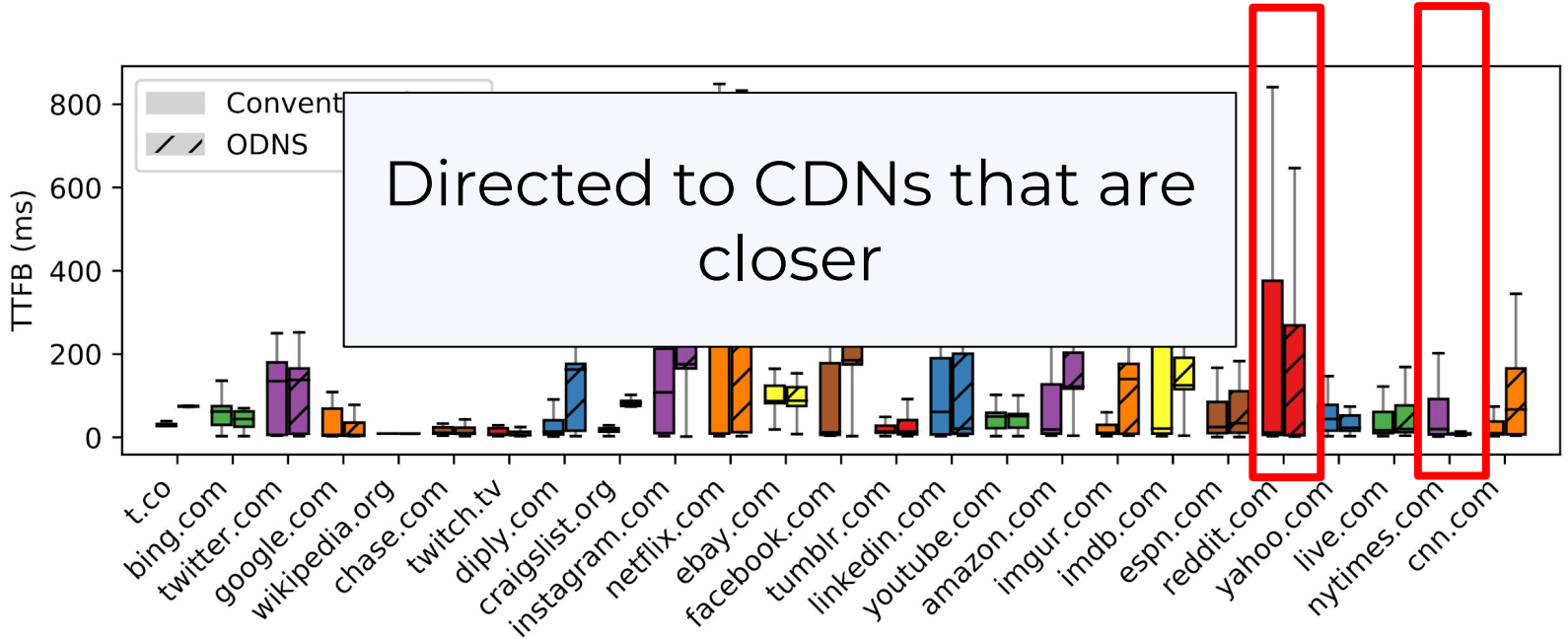
ODNS Overhead: Page Load Time



ODNS Overhead: Page TTFB

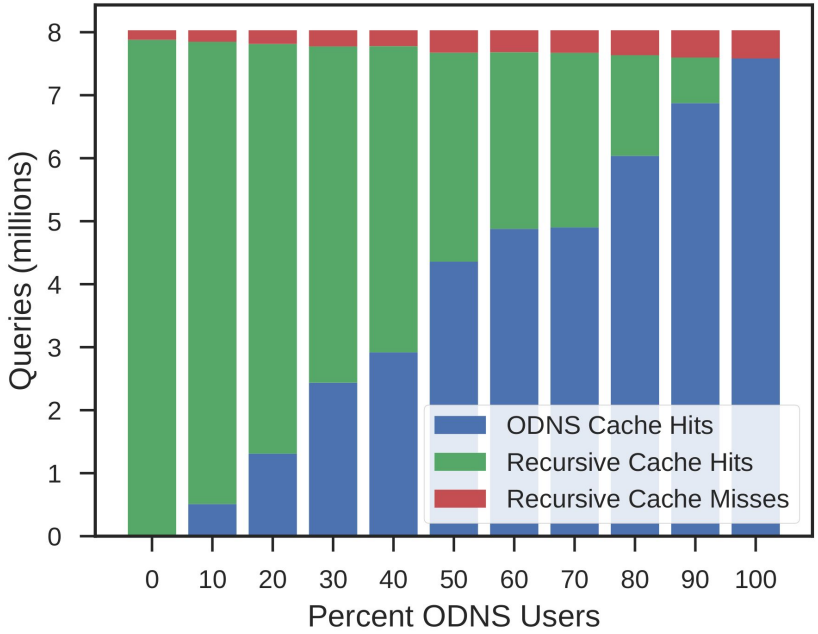


ODNS Overhead: Page TTFB



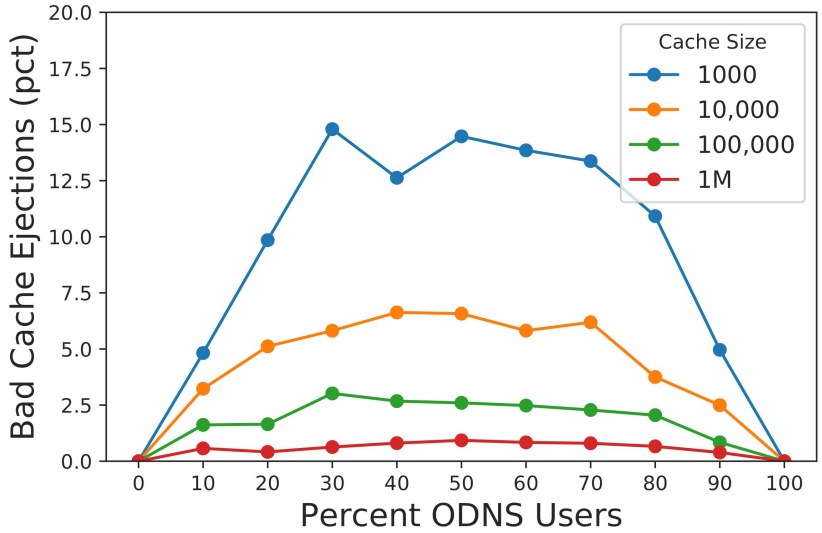
Impact on Recursive Cache

- Simulated with trace of ~8M queries
- If caching at stub, ODNS reduces traffic burden on the recursive resolver



Impact on Cache (2)

- Undesirable cache entries?
- Some resolvers ignore TTL = zero
- “Bad” == ODNS entry causing non-ODNS to be ejected



Discussion

- Challenges:
  - EDNS0 Client Subnet
  - QNAME length
  - 0x20 bit encoding

- Policy-based routing

Thank you

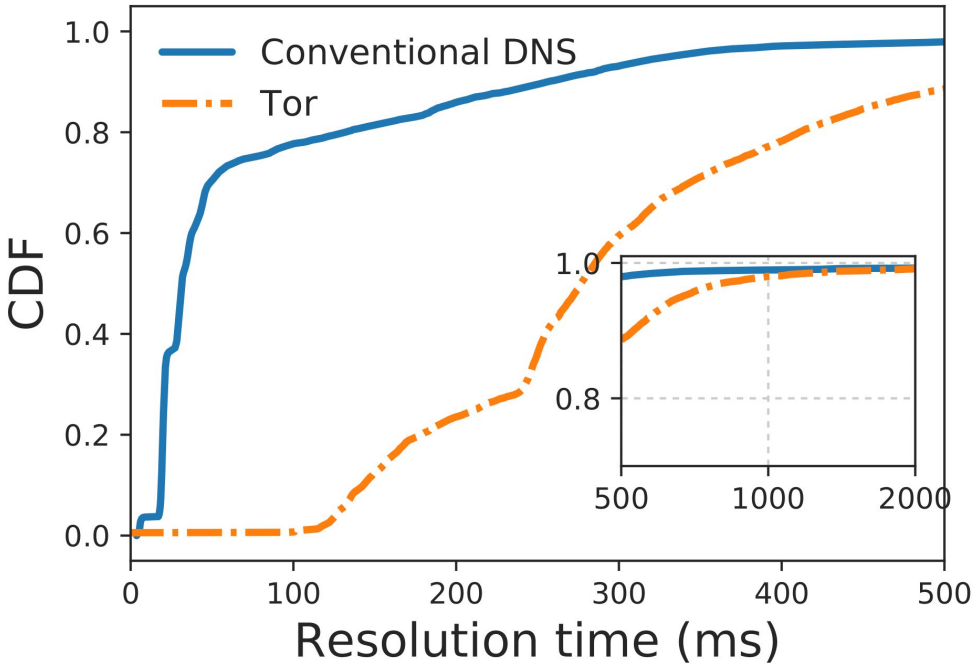
Paul Schmitt
pschmitt@cs.princeton.edu

PRINCETON
UNIVERSITY

Backup slides

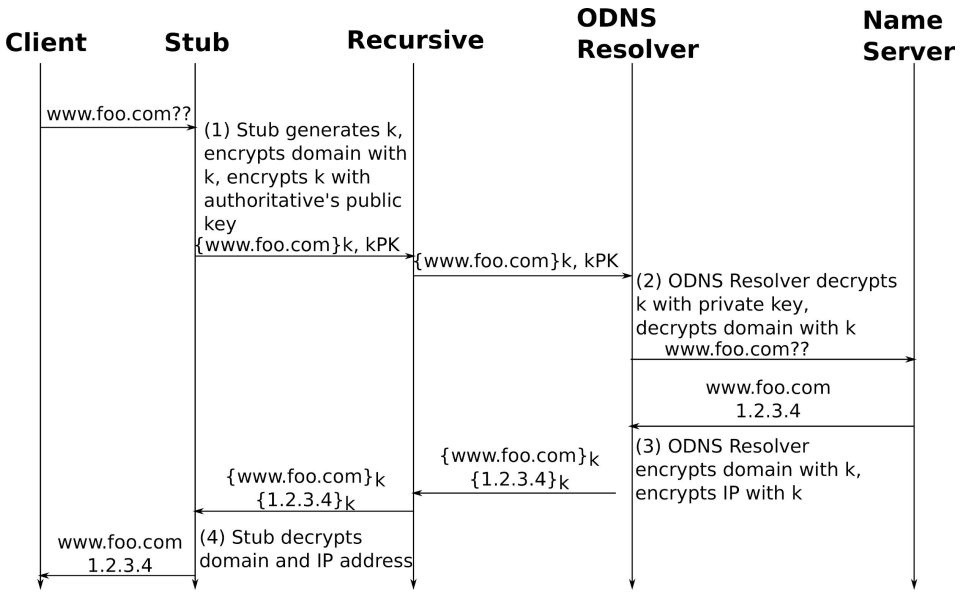
Why Not Tor?

- Latency (median)
  - ODNS: 31.31 ms
  - Tor: 276.76 ms
- Censorship concerns
- Exit node can be associated with traffic



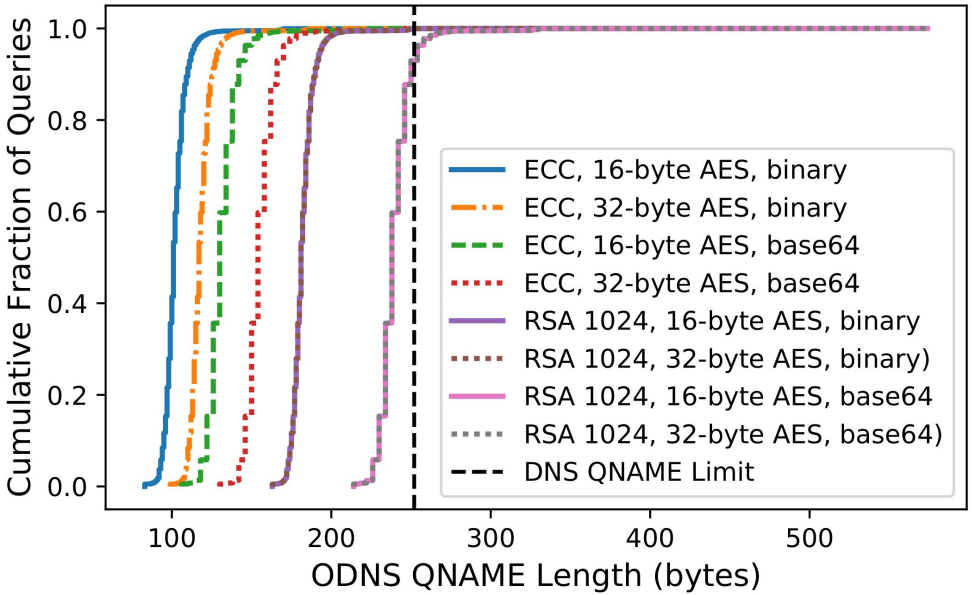
Protocol

- Stub encrypts query with session key and session key with resolver public key
- Stub appends resolver name to encrypted query
- ODNS resolver decrypts session key with private key, query with session key, and encrypts response



QNAME Length

- QNAME = 4 sets of 63 bytes
- base64 encoding
  - 0x20 bit encoding issue



EDNS0 Client Subnet

- Must avoid some recursive resolvers

Open Recursive Resolver (IP)	EDNS0 Client Subnet	0x20
Google (8.8.8.8)	✓	
Dyn (216.146.35.35)	✓	
Fourth Estate (45.77.165.194)	✓	
GreenTeamDNS (81.218.119.11)	✓	
Cloudflare (1.1.1.1)		✓
Verisign (64.6.64.6)		✓
Quad9 (9.9.9.9)		
Level3 (209.244.0.3)		
OpenDNS Home (208.67.222.222)		
Norton ConnectSafe (199.85.126.10)		
Comodo Secure DNS (8.26.56.26)		
DNS.WATCH (84.200.69.80)		
SafeDNS (195.46.39.39)		
FreeDNS (37.235.1.174)		
Hurricane Electric (74.82.42.42)		
Ultra (156.154.71.1)		