

# ODNS: Oblivious DNS

Anne Edmundson, **Paul Schmitt**, Nick Feamster, Jennifer Rexford, *Princeton University*  
Allison Mankin, *Salesforce*

OARC 28  
9 March 2018



# Conventional DNS



Client



Recursive



Root Server

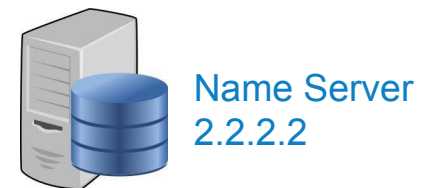


TLD Server  
1.1.1.1

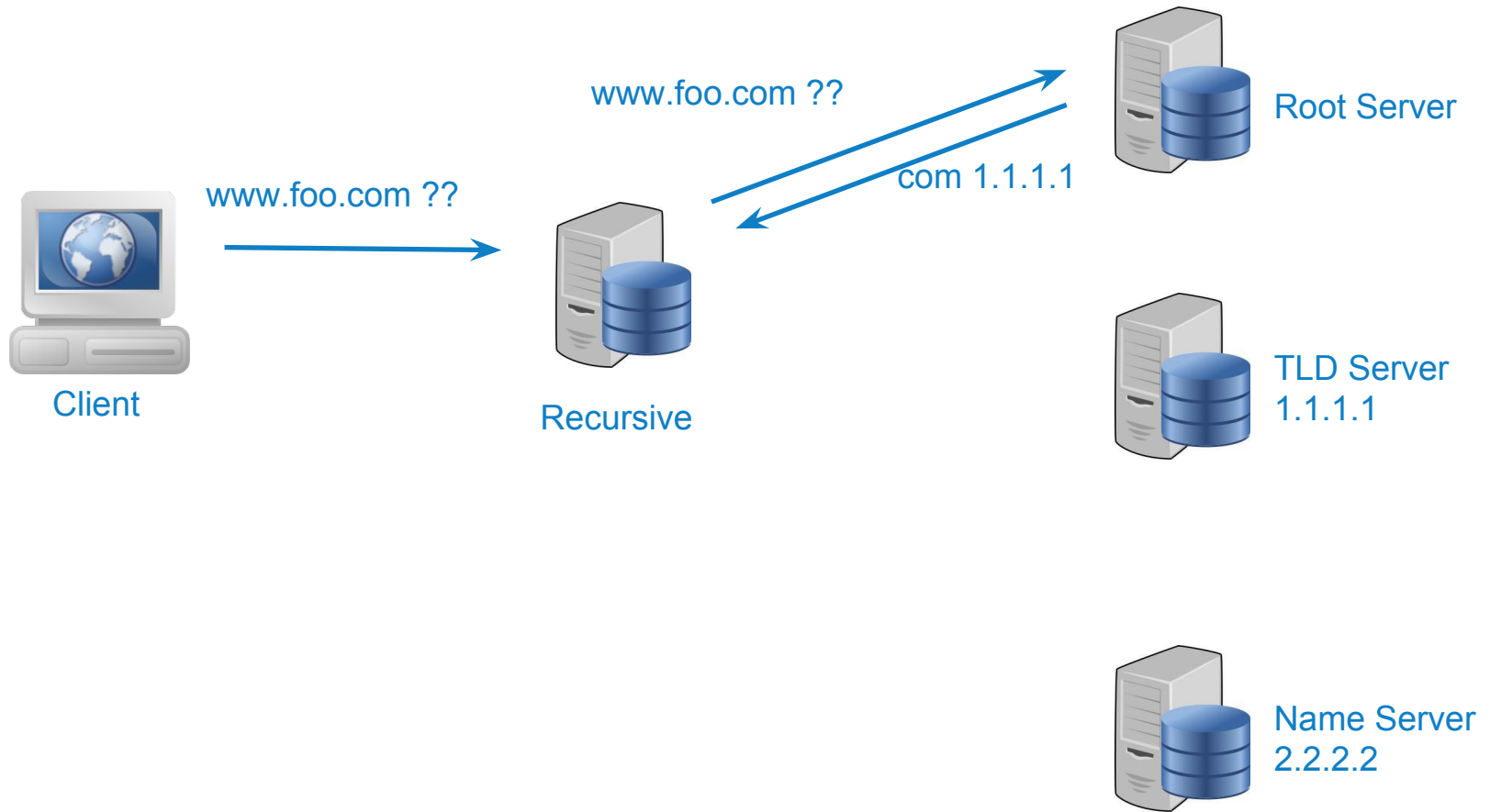


Name Server  
2.2.2.2

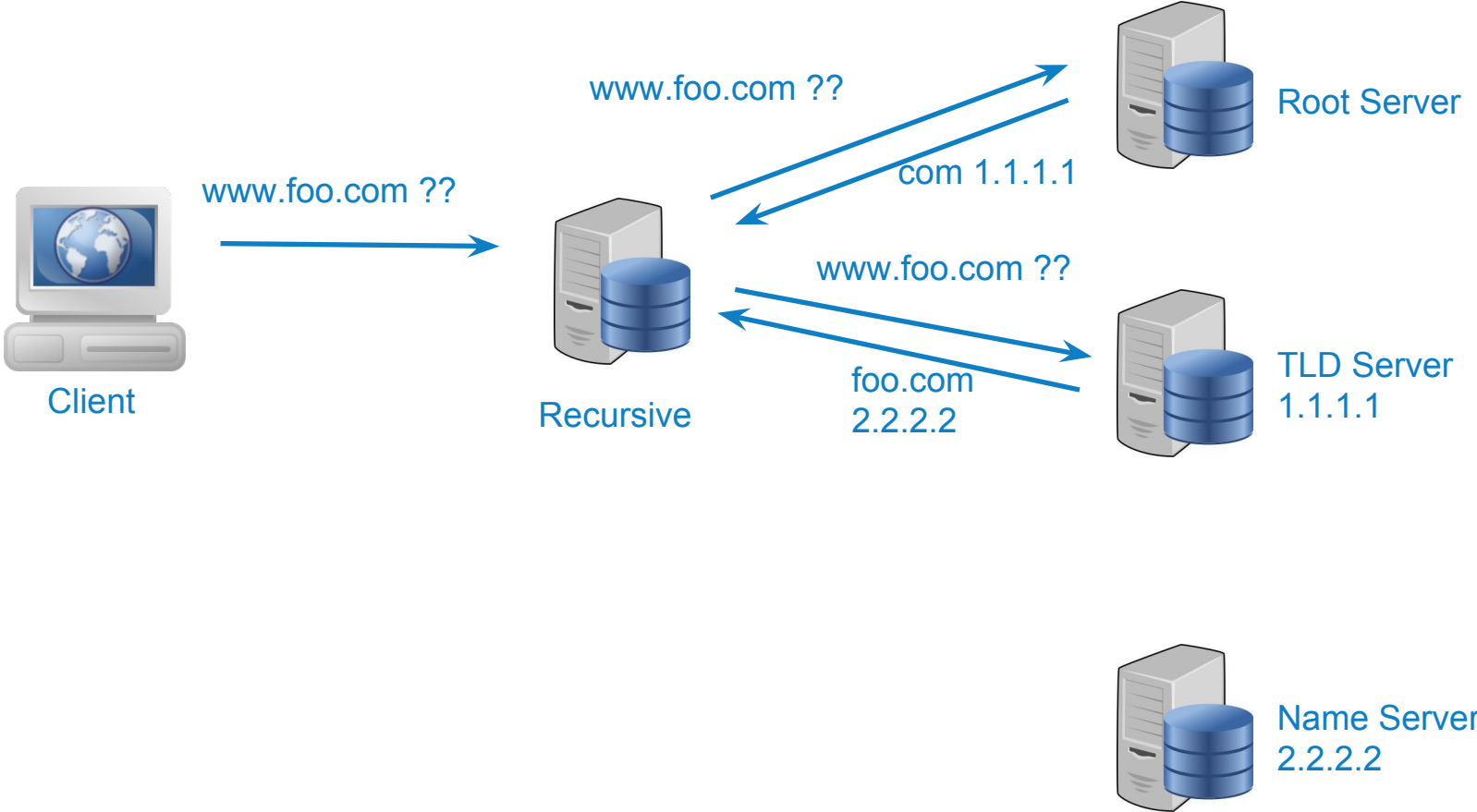
# Conventional DNS



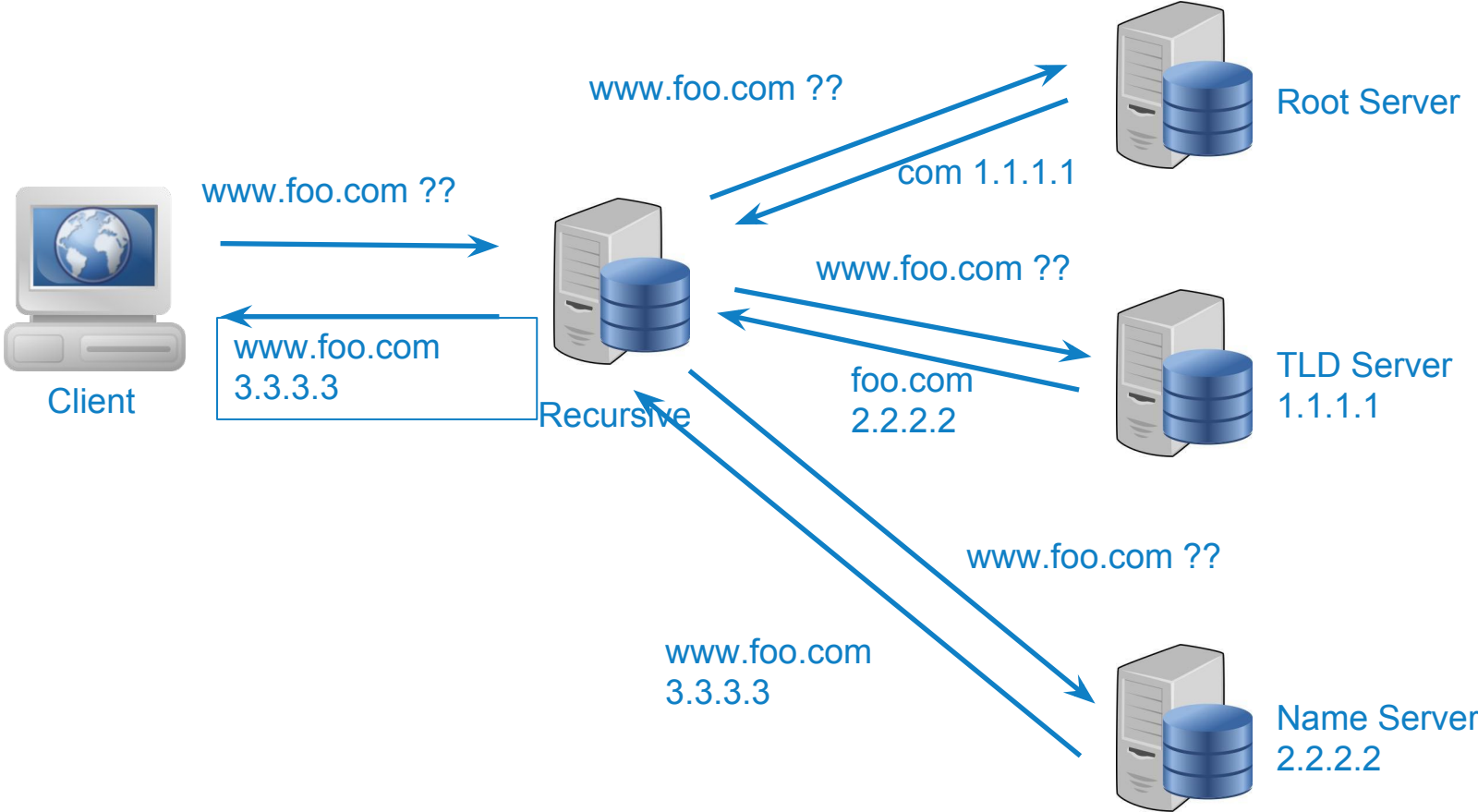
# Conventional DNS



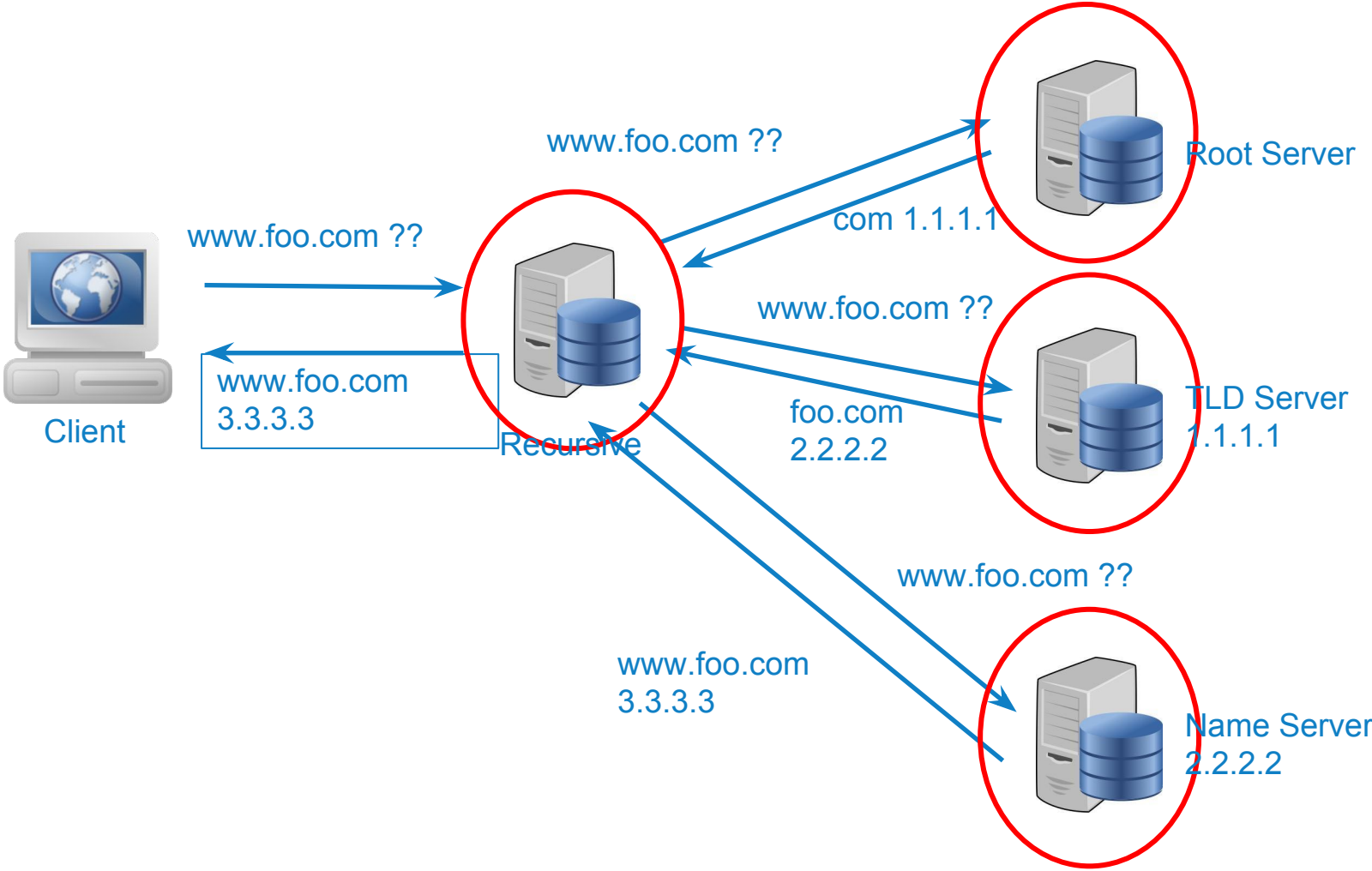
# Conventional DNS



# Conventional DNS



# Conventional DNS



# Threat Model

- **User Data:** a user's identity, queries, browsing patterns
- **Attacker's Goal:** learn about user data
- **Attacker's Capabilities:**
  - monitor traffic
  - gain access to DNS logs



# Threat Model

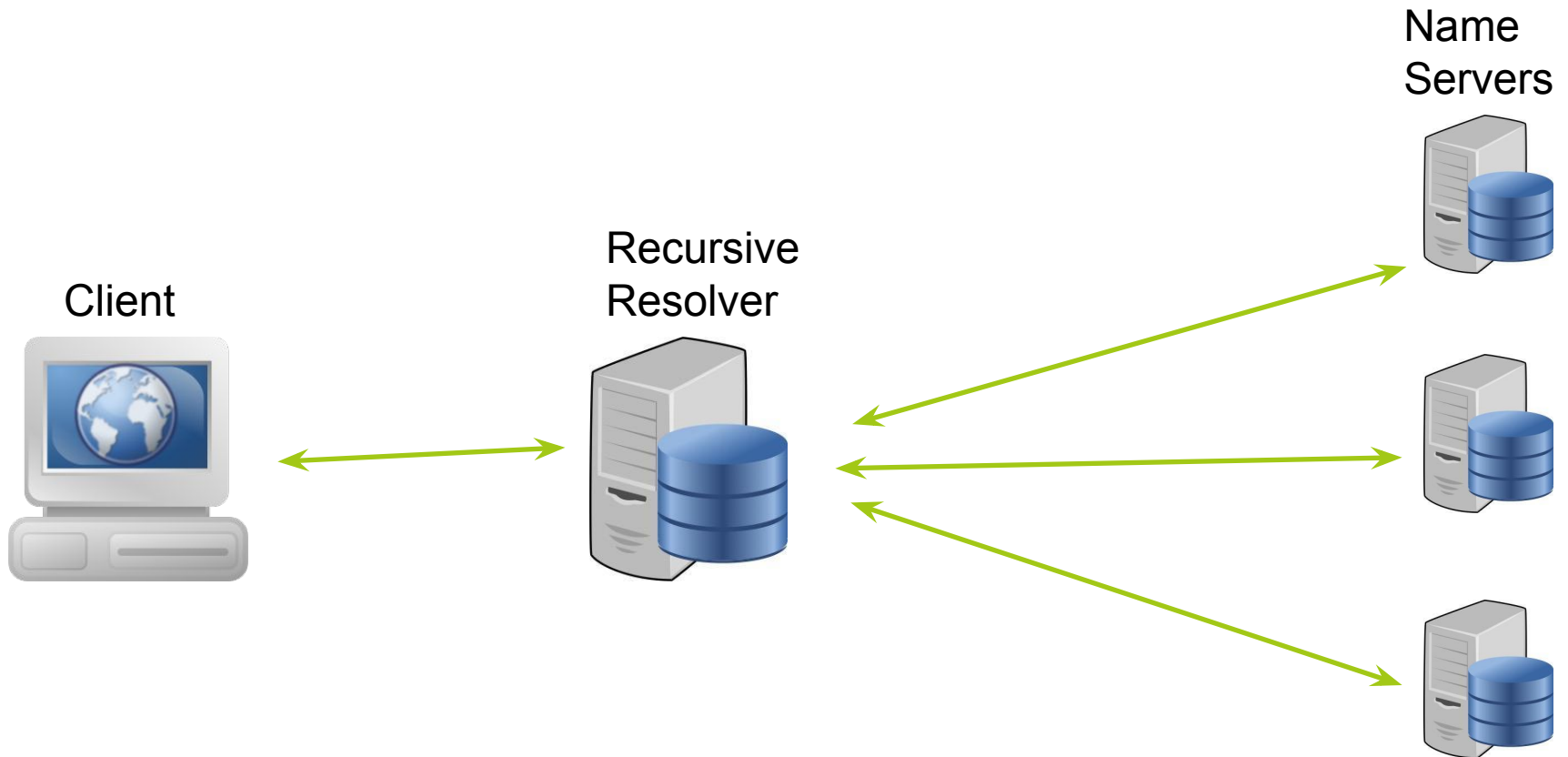
- **User Data:** a user's identity, queries, browsing patterns
- **Attacker's Goal:** learn about user data
- **Attacker's Capabilities:**
  - monitor traffic
  - gain access to DNS logs

***DNS operators could be targets of data requests***

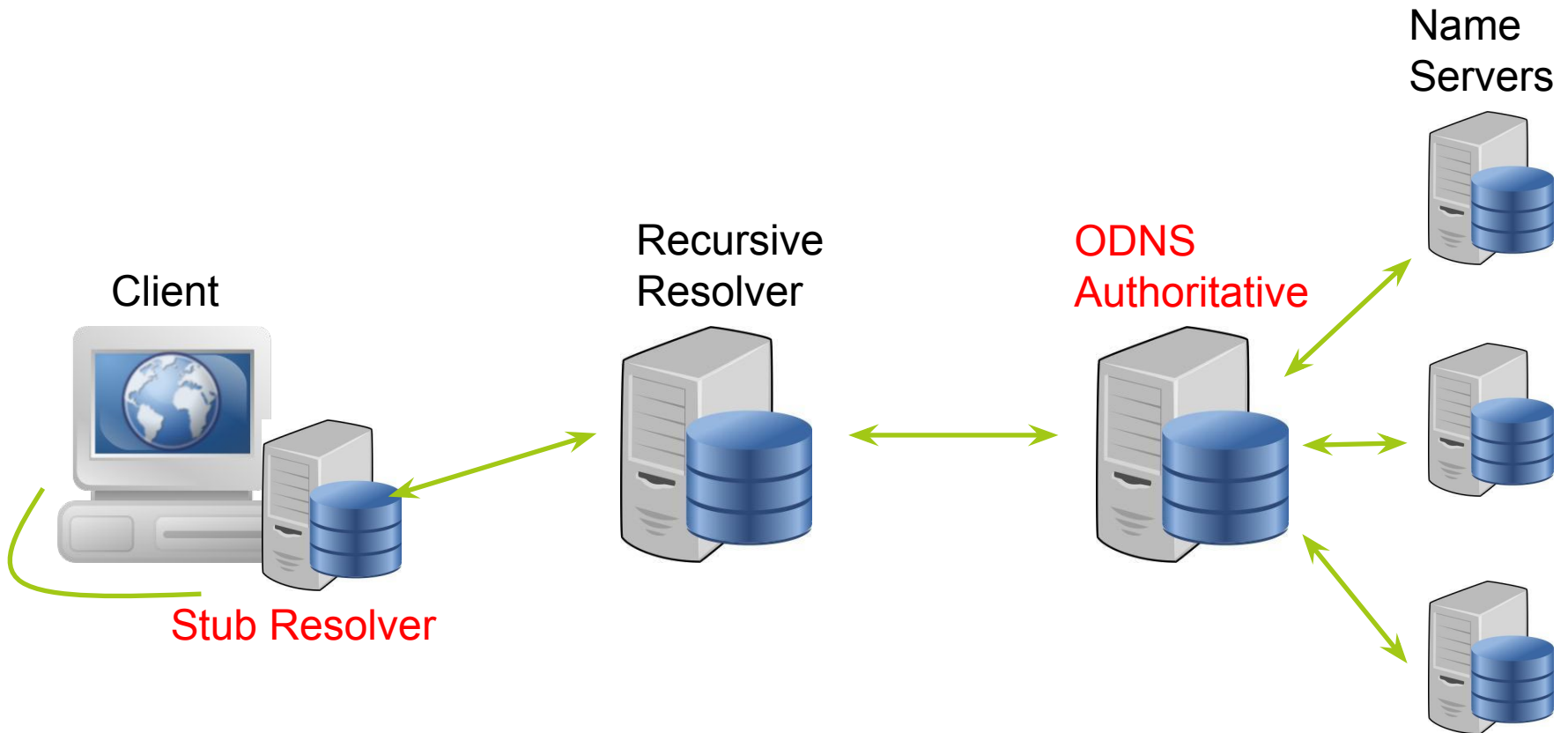
# Existing Approaches

- DNS Query Name Minimization
- DNS-Over-TLS
- Quad9
- Onion Services (via Tor)

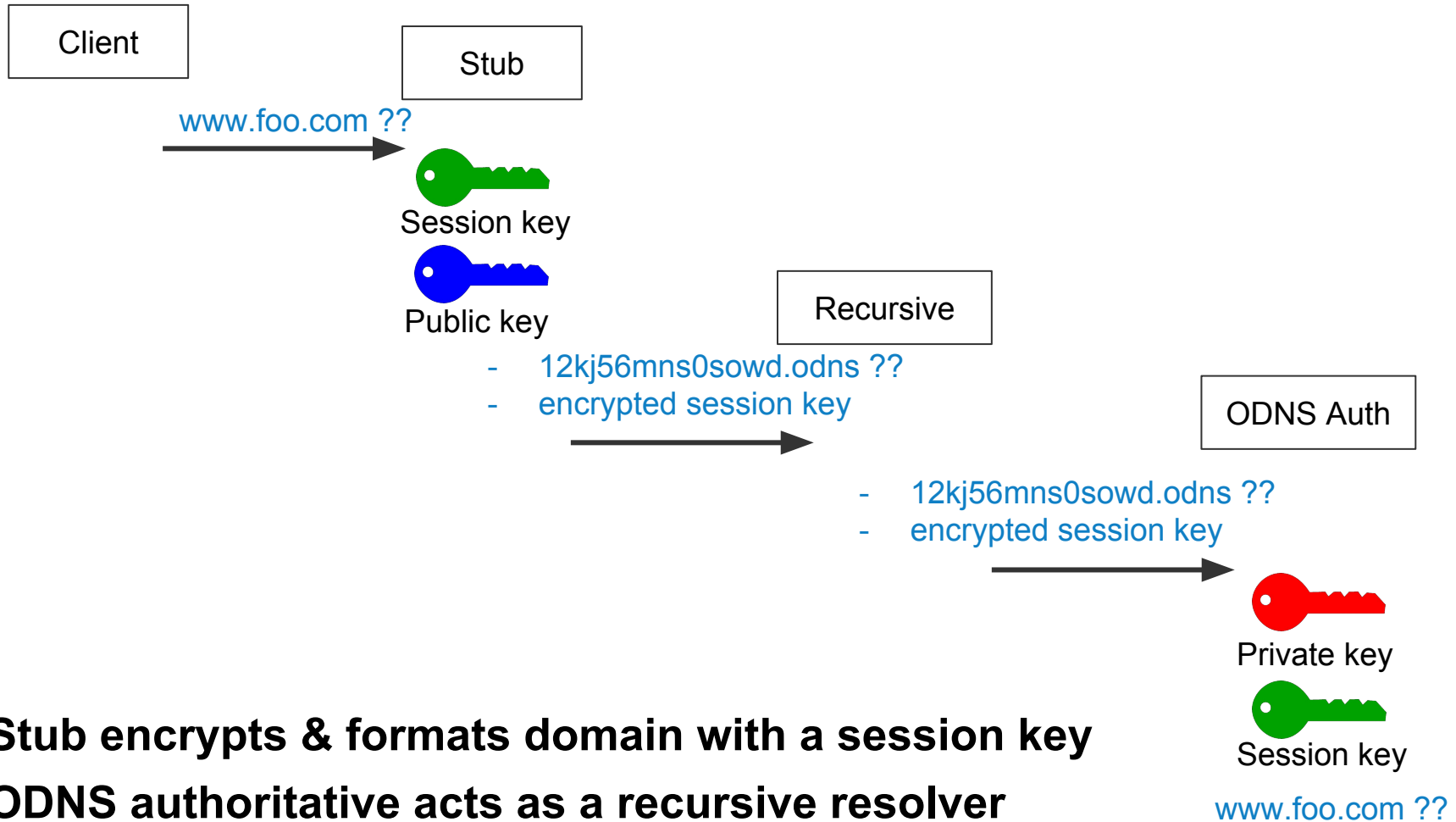
# ODNS Overview



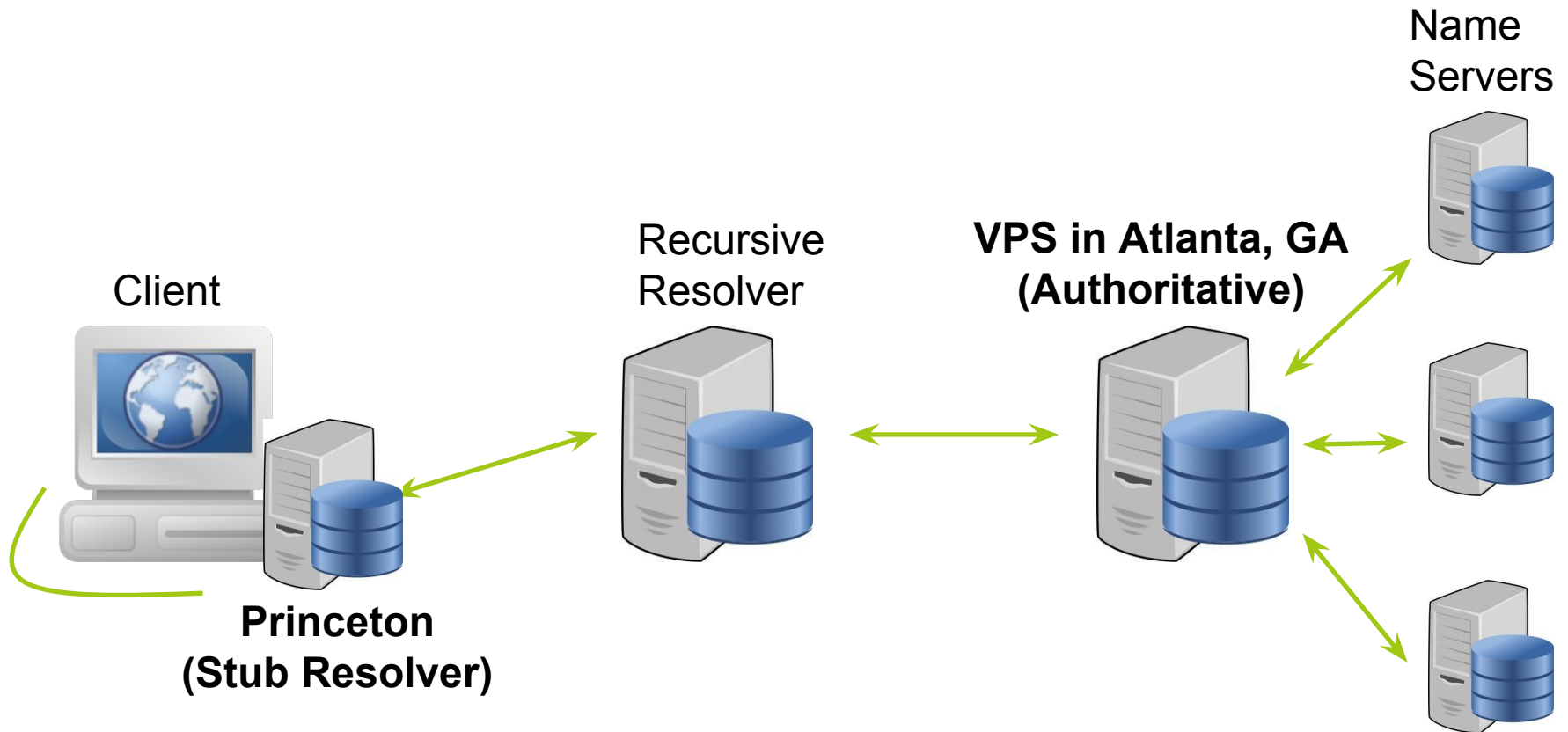
# ODNS Overview



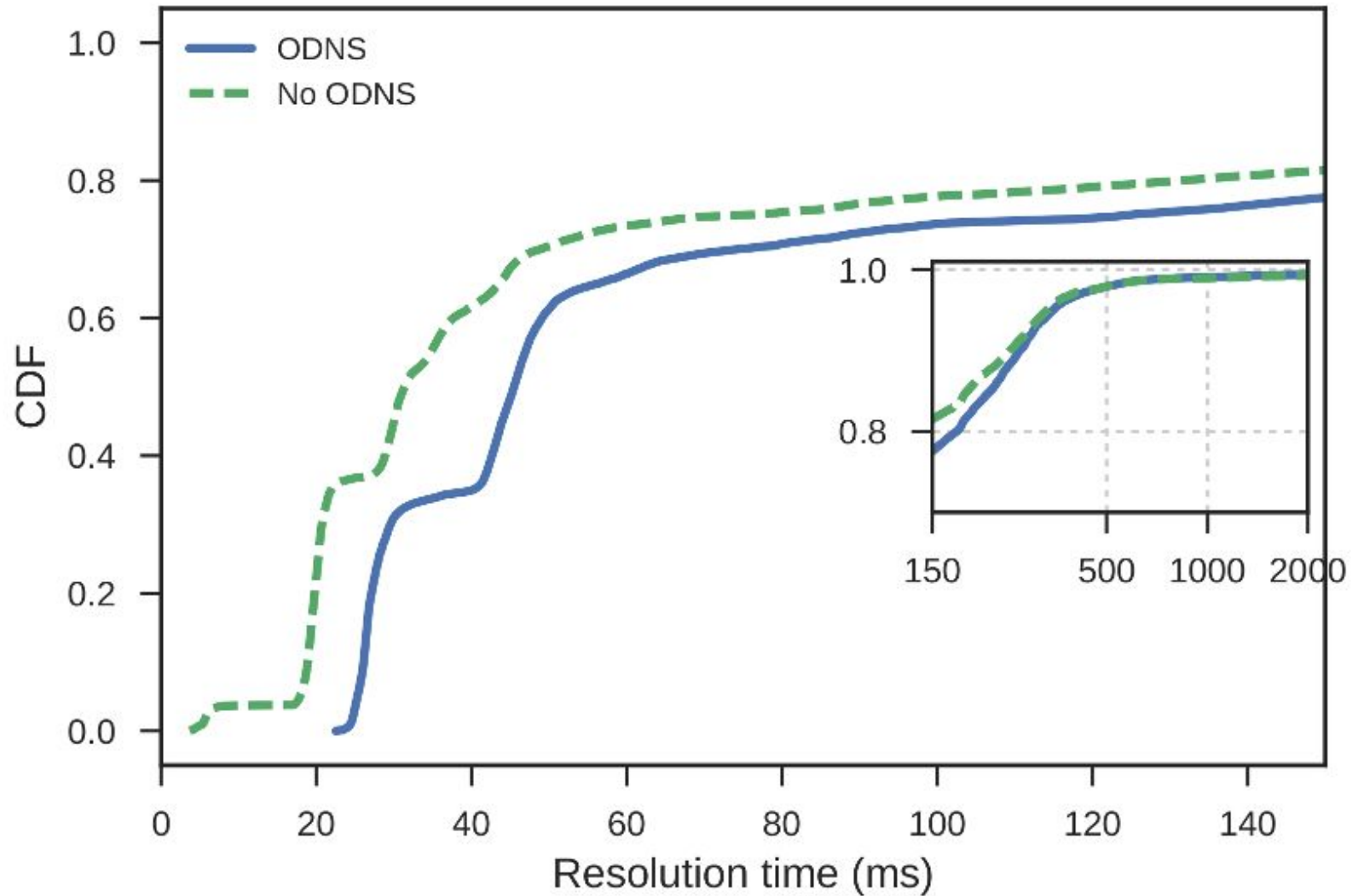
# ODNS Privacy Overview



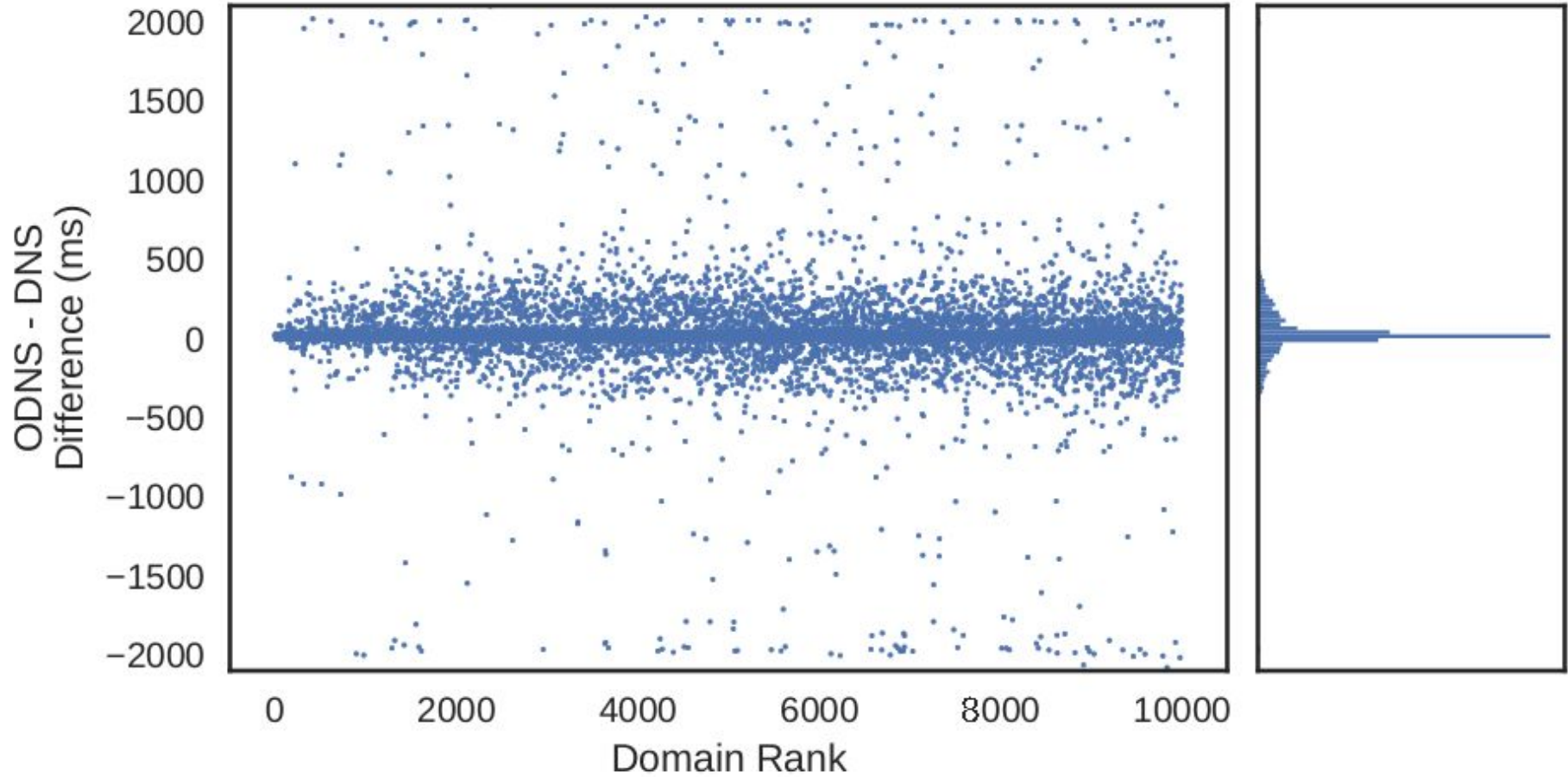
# Prototype & Testbed



# ODNS Overhead (1)

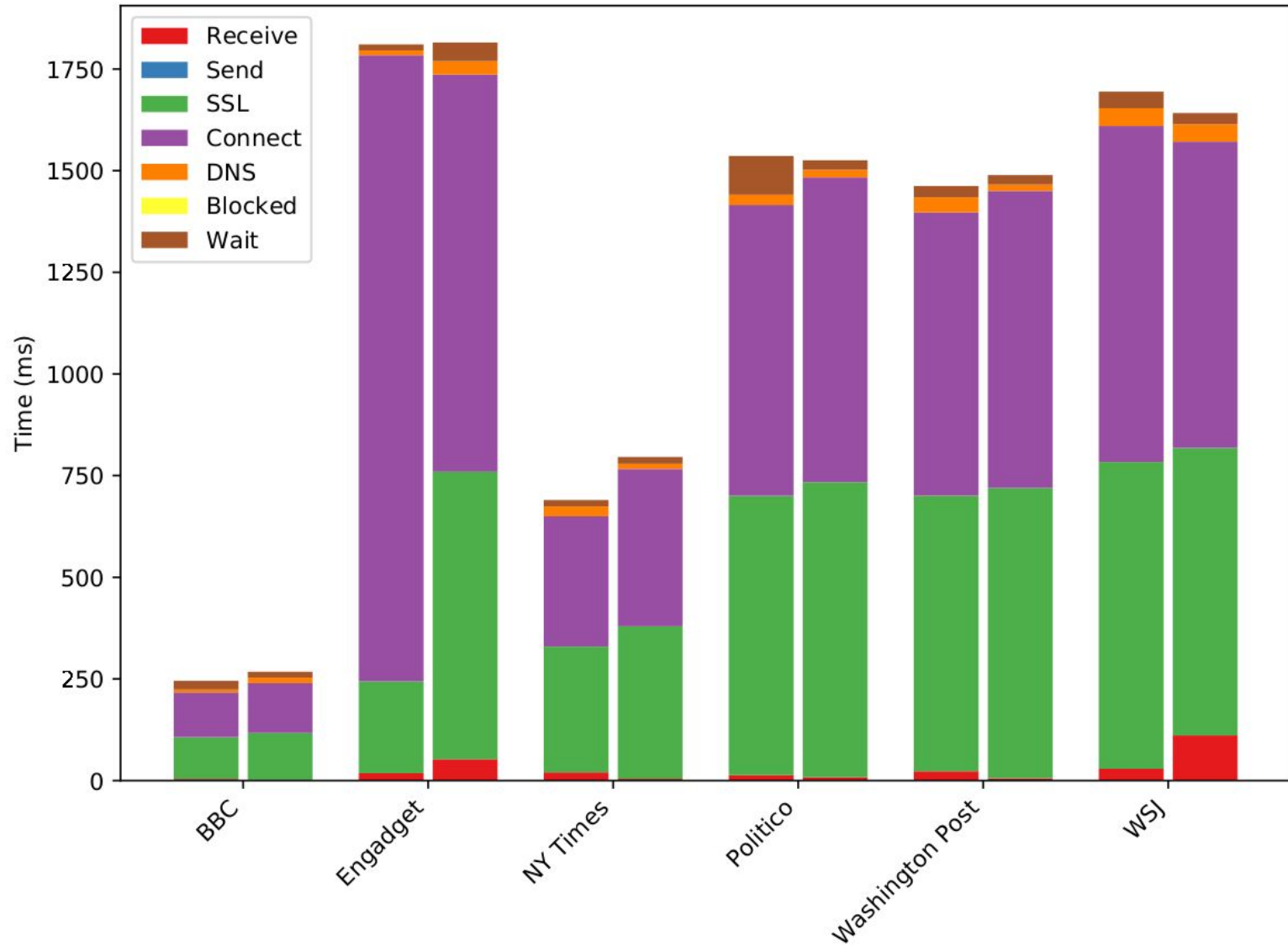


# ODNS Overhead (2)





# ODNS Overhead (3)



# Conclusion

- Designed ODNS to protect privacy by decoupling clients' identities with their requests
- Implemented and evaluated a prototype of the design
- Future
  - Implementation based on Knot
  - Explore authoritative replication
  - Examine impact on caching

# Questions?

[pschmitt@cs.princeton.edu](mailto:pschmitt@cs.princeton.edu)



# ODNS Function Overhead

