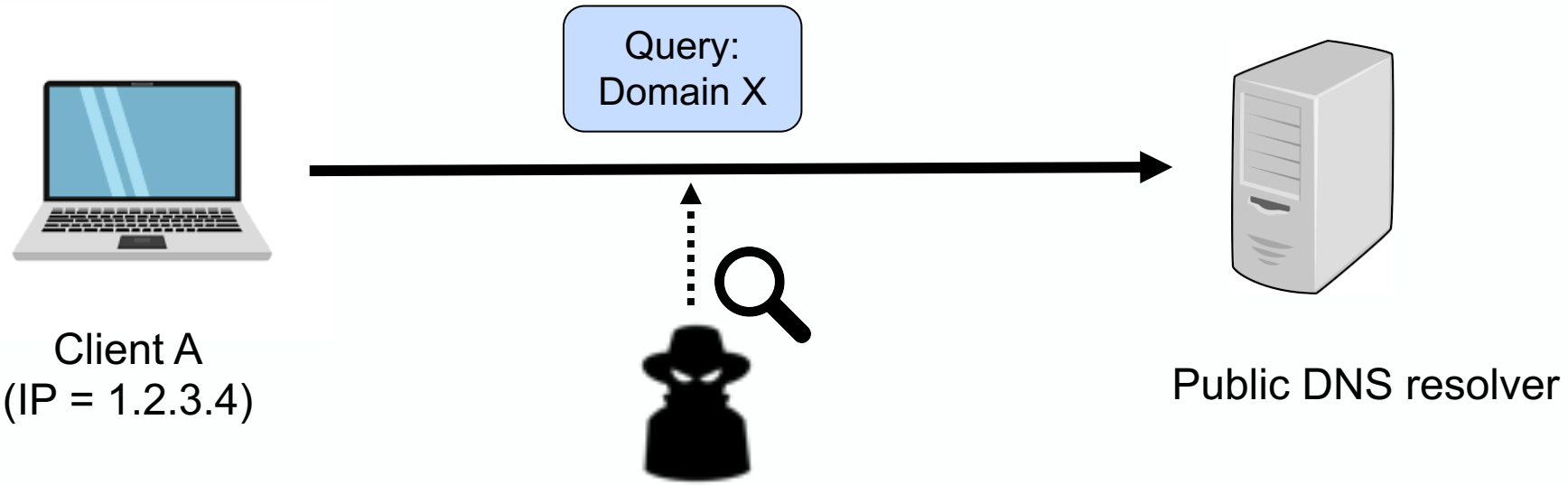
Programmable In-Network Obfuscation of DNS Traffic

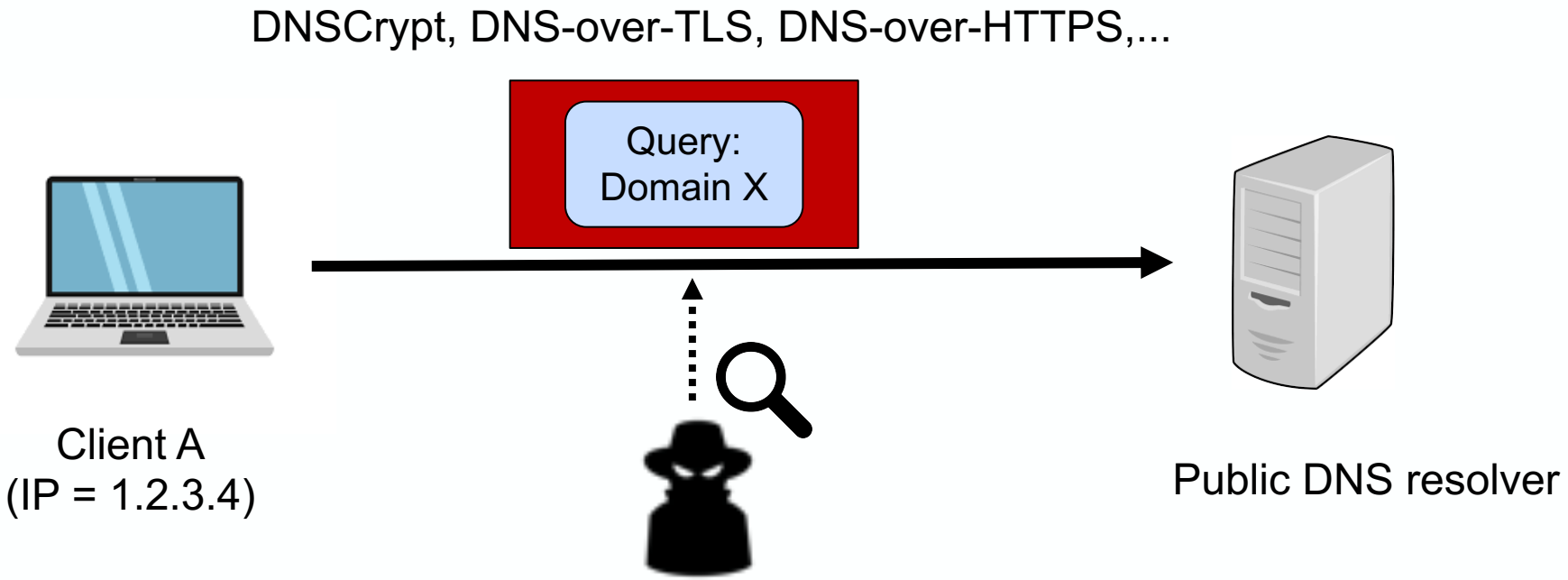
Liang Wang, Hyojoon Kim, Prateek Mittal, Jennifer Rexford

Princeton University

Do53 Traffic Reveals Sensitive Information

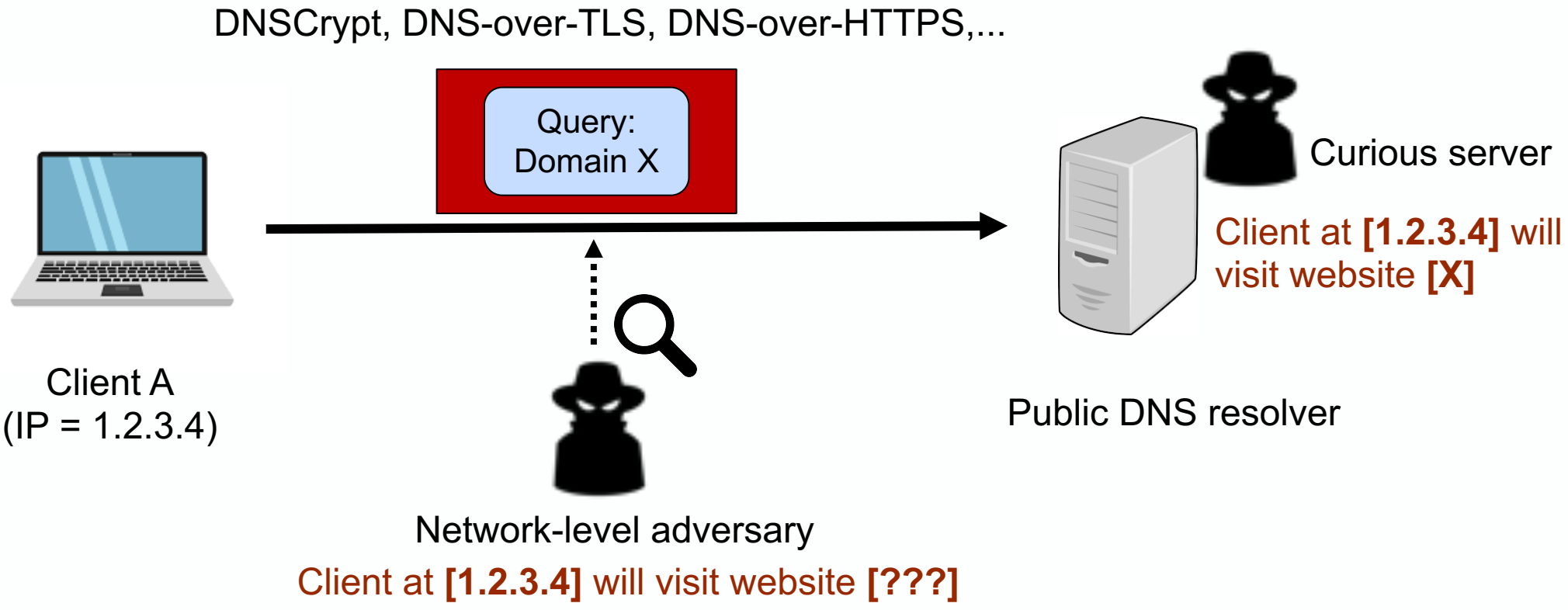


Even Encrypted DNS Communications Reveal Sensitive Information



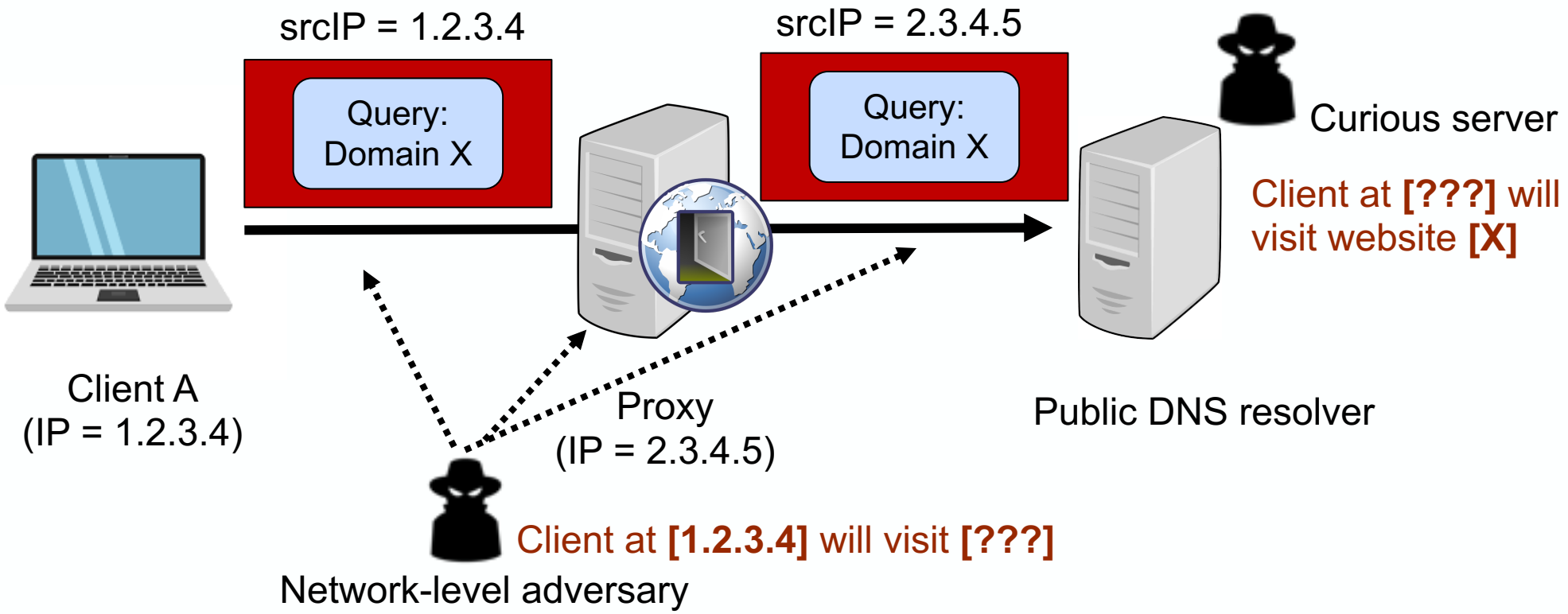
Client at **[1.2.3.4]** will visit website **[???]**

Encrypted DNS: DNS Resolver --- A Single Point of Privacy Failure



Need to hide client IP addresses from DNS resolvers

Proxy-based DNS Protects Client IP Addresses



Solutions: DNS over Tor, Anonymized DNSCrypt, Oblivious DNS, Oblivious DoH, …

Proxy-based DNS: Practical Challenges

Solutions: DNS over Tor, Anonymized DNSCrypt, Oblivious DNS, Oblivious DoH, …

- Higher latency
- Modifications to DNS client / infrastructure

➢ High deployment barriers for proxy-based solutions

Need a lightweight IP anonymization method that requires no modifications to DNS client and server

Can We Embed Proxy in Network Elements?

Opportunities:

- Programmable data-plane hardware
  - Offload privacy functionality to the network
    - High speed
    - Avoid end user involvement

- Growing ubiquity of IPv6 in the Internet core
  - Use IPv6 address to embed information

Our Solution: PINOT

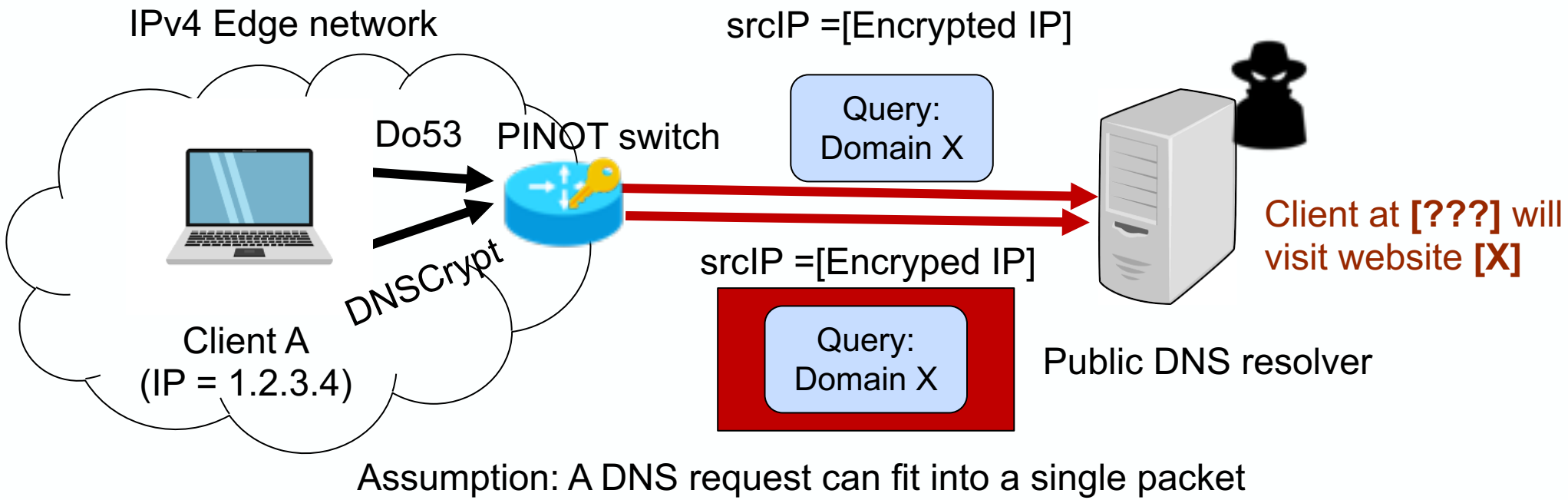
PINOT: A lightweight in-network IP address obfuscation system

- **Goal**: Prevent public DNS services from associating client IP addresses to queries
- Use programmable switch to encrypt IP addresses at a high speed (12.8 Tbps)
- No modification to DNS protocols; No additional client software installation
- Complementary to encrypted DNS

PINOT in An Edge Network

Run PINOT at the network border

- DNS requests: Encrypt the source IP address in each packet

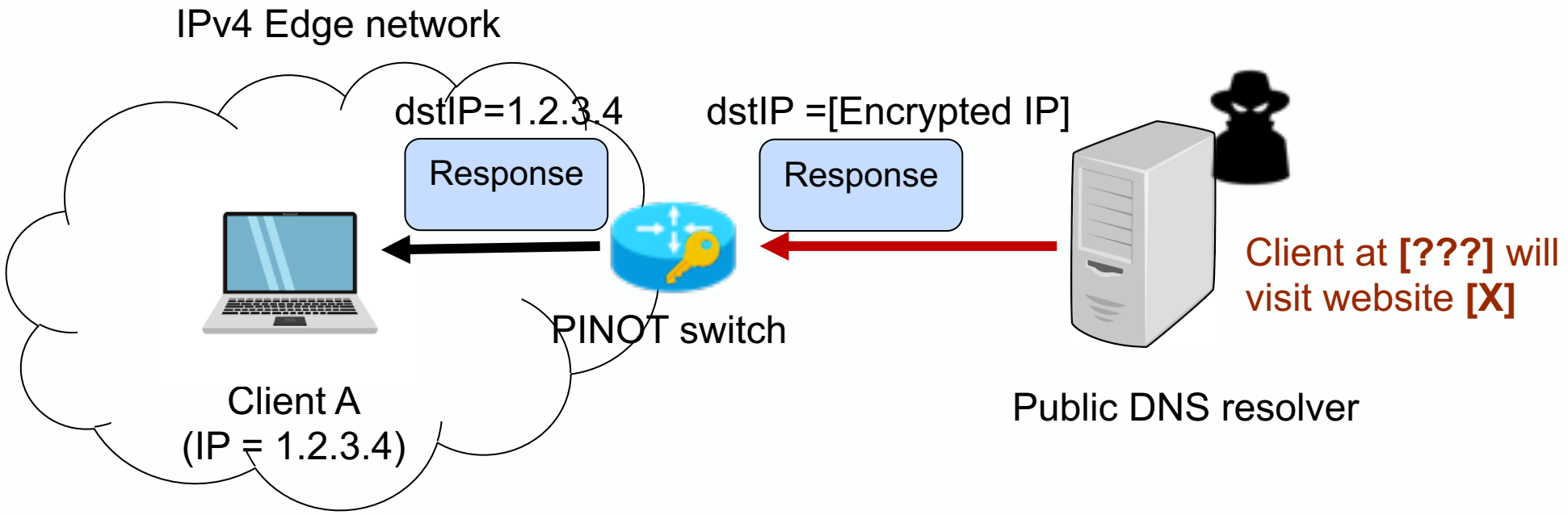


Assumption: A DNS request can fit into a single packet

PINOT in An Edge Network

Run PINOT at the network border

- DNS requests: Encrypt the source IP address in each packet
- DNS responses: Decrypt the destination IP address and forward packets

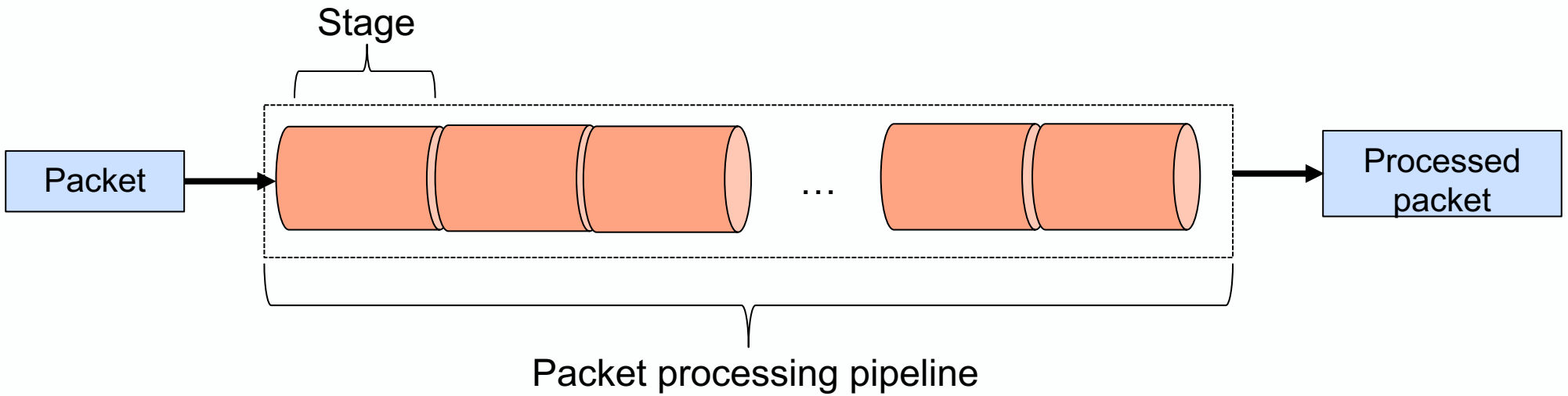


Challenges

- Perform encryption on resource-constrained programmable switch
- Receive return traffic without cooperation
- Work with asymmetric routing

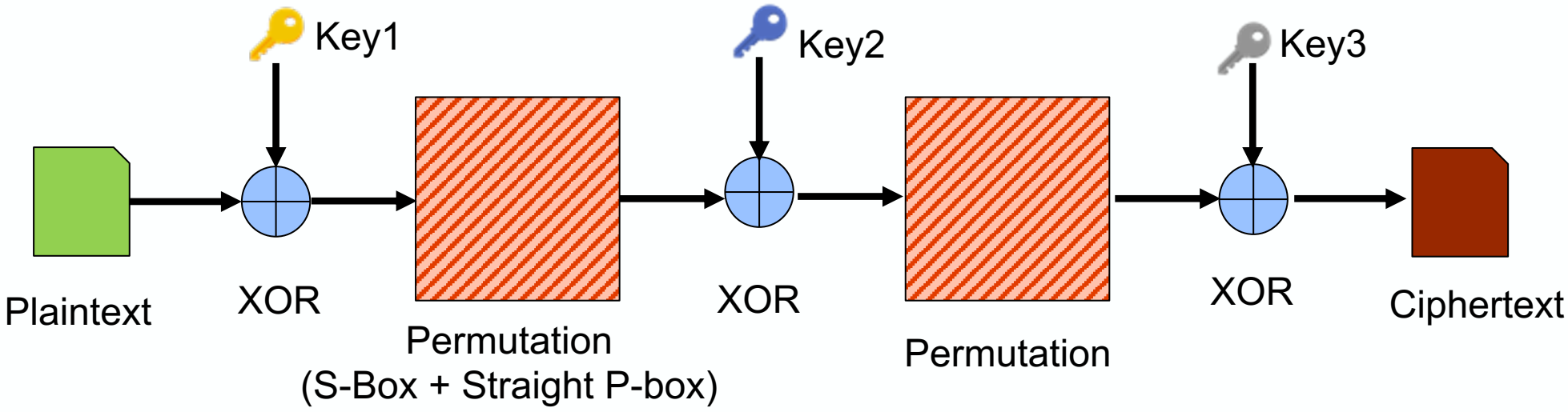
Programmable Switch Resource Constraints

- Limited memory
- Limited operations
  - 1 pipeline: a small number of stages
  - 1 stage: a limited number of table lookups, and math/logical ops



Efficient Encryption of IP in Data Plane

- AES is too expensive on data plane
- Solution: Two-round Even-Mansour encryption (2EM)



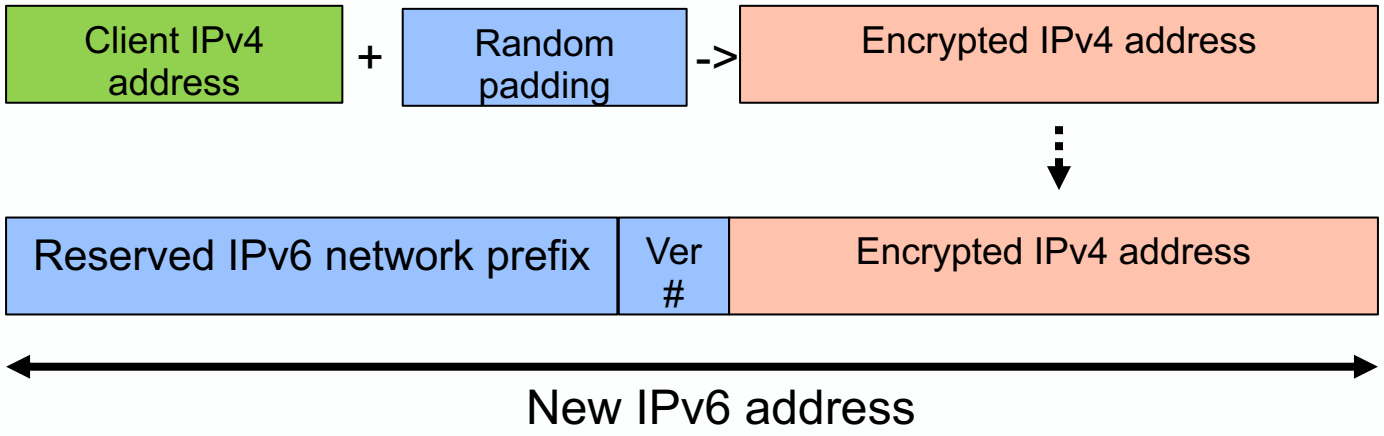
- 2EM can be implemented using table lookups and XORs

Efficient Encryption of IP in Data Plane

- AES is too expensive on data plane
- Solution: Two-round Even-Mansour encryption (2EM)
  - Encrypt IP using a single pass through packet processing pipeline
  - ➢ **Encrypt packets at 3.2 Tbps on our Intel Tofino switch!**
  - Pad IPv4 address with random bits for stronger security and privacy
  - ➢ **Consecutive requests from the same client have distinct client source IP addresses**
- See paper for more details

IPv6 Encoding for Stateless Encryption and Routing

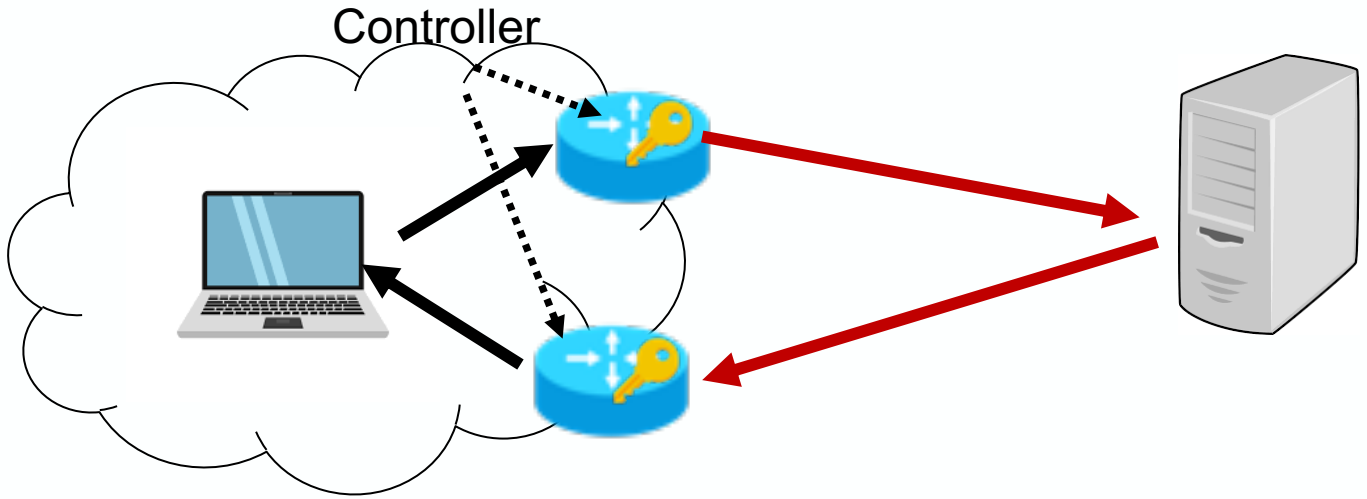
- Challenge:
  - Store ciphertext (> 32 bits) and encryption metadata (key version #)
  - Ensure successful routing of return traffic
- Solution: Convert IPv4 packets to IPv6 packets
  - Information required for decryption are stored in IPv6 address
  - **PINOT only stores encryption keys**



PINOT is stateless

IPv6 Encoding for Stateless Encryption and Routing

- Challenge: Return traffic can go to any ingress point
- Solution: A centralized controller for distributing the per-AS secret keys



PINOT can handle asymmetric routing easily

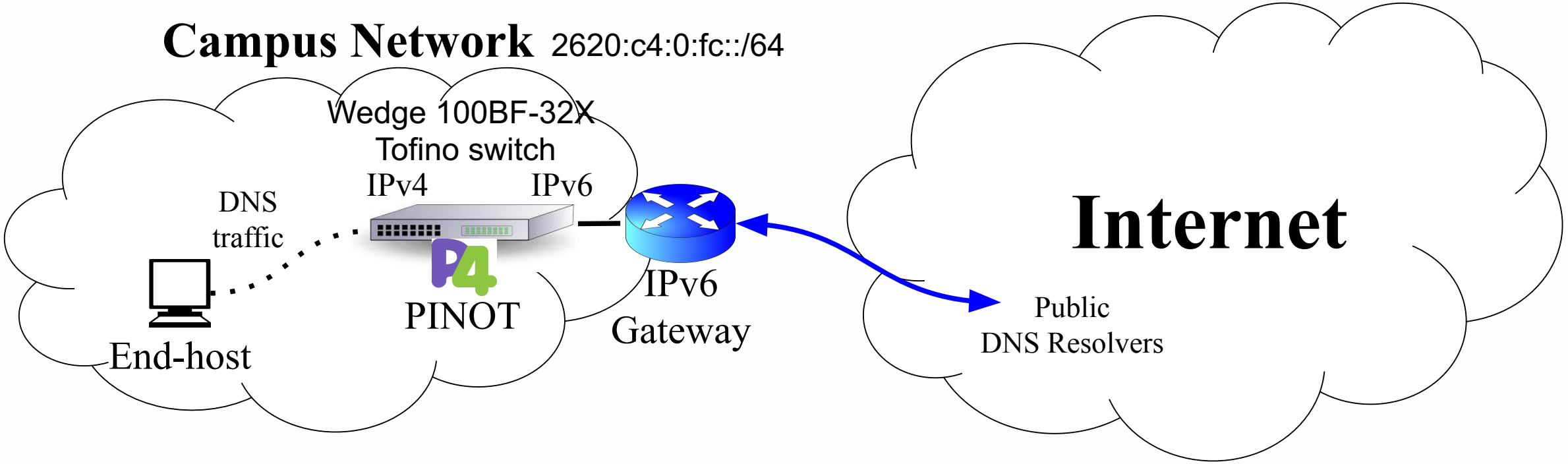
PINOT is Complementary to Encrypted DNS

	Query encryption	IP hiding	Modification to client	Proxy overhead
Do53	No	No	-	-
Encrypted	Yes	No	Yes	-
Proxy	Yes	Yes	Yes	High
PINOT + Do53	No	**Yes**	**No**	**Low**
PINOT + Encrypted	Yes	**Yes**	**No***	**Low**

**Assuming an encrypted DNS solution has already been deployed, using PINOT to achieve IP obfuscation does not require modifying the existing DNS client/server software*

Using PINOT with encrypted DNS protocols offers better privacy with little performance overhead

Real-World Deployment of PINOT at Princeton



PINOT source Code: https://github.com/liangw89/p4privacy/tree/master/pinot

Evaluation of PINOT for Do53

- Target resolver: 350+ public resolvers with both IPv4 and IPV6 support
- Query: 10 queries for random domains from Top 1M to each resolver
- Setting: IPv6 network, IPv4 network, and IPv4 + PINOT

➢ **PINOT is feasible**
  - DNS responses are consistent across settings

➢ **PINOT introduces low latency**
  - Potential overhead: IPv6 and IPv4 packets take different routing paths
  - **PINOT does not add extra latency in 97% of the cases**

PINOT for Other Connectionless Protocols

- **NTP:** IPv6 host discovery and scanning
  - Single-packet protocol like DNS

- **WireGuard VPN**: Client IP address collection
  - Crypto-key routing allows **per-packet** encryption without disrupting connectivity

PINOT prevents the public NTP/WireGuard VPN servers from learning the real client IP addresses

Conclusion

- PINOT, an in-network proxy service
- Low performance overhead
- Low deployment barriers
  - Single network
  - No modification to DNS
  - No cooperation from end-users
- A network deploy can PINOT to provide extra privacy for users as a value-added service
- A useful building block for bootstrapping more privacy applications!

PINOT Variants

- PINOT for IPv6 network
  - Use the lowest 64 bits of IPv6 address for encryption

- PINOT for connection-oriented protocols
  - Need to maintain per-connection state
  - Work with DoH and DoT