UCI University of California, Irvine

Mobile Sessions in Content-Centric Networks

Marc Mosko
Xerox PARC
marc.mosko@parc.com

Ersin Uzun
Xerox PARC
ersin.uzun@parc.com

Christopher A. Wood
University of California Irvine
woodc1@uci.edu

IFIP Networking 2017 — June 12, 2017

Agenda

- CCN recap
- CCNxKE design & features
- Experimental results
- Conclusion

CCN Highlights

- Architecture for transferring **named data** from producer to consumer upon request
- Names are **cryptographically bound** to data
- Requests (datagrams) are routed based on **names** rather than endpoint addresses
- Content can be **opportunistically cached** in the network

Benefits

- Simplified protocol stack
- Native content dissemination
- Better opportunities for transport
- …

(Open) Problems

- How is sensitive long-term keying material transmitted from producer to consumer?
- How can content be encrypted end-to-end from producer to consumer?
- What about forward secrecy?

Our Approach

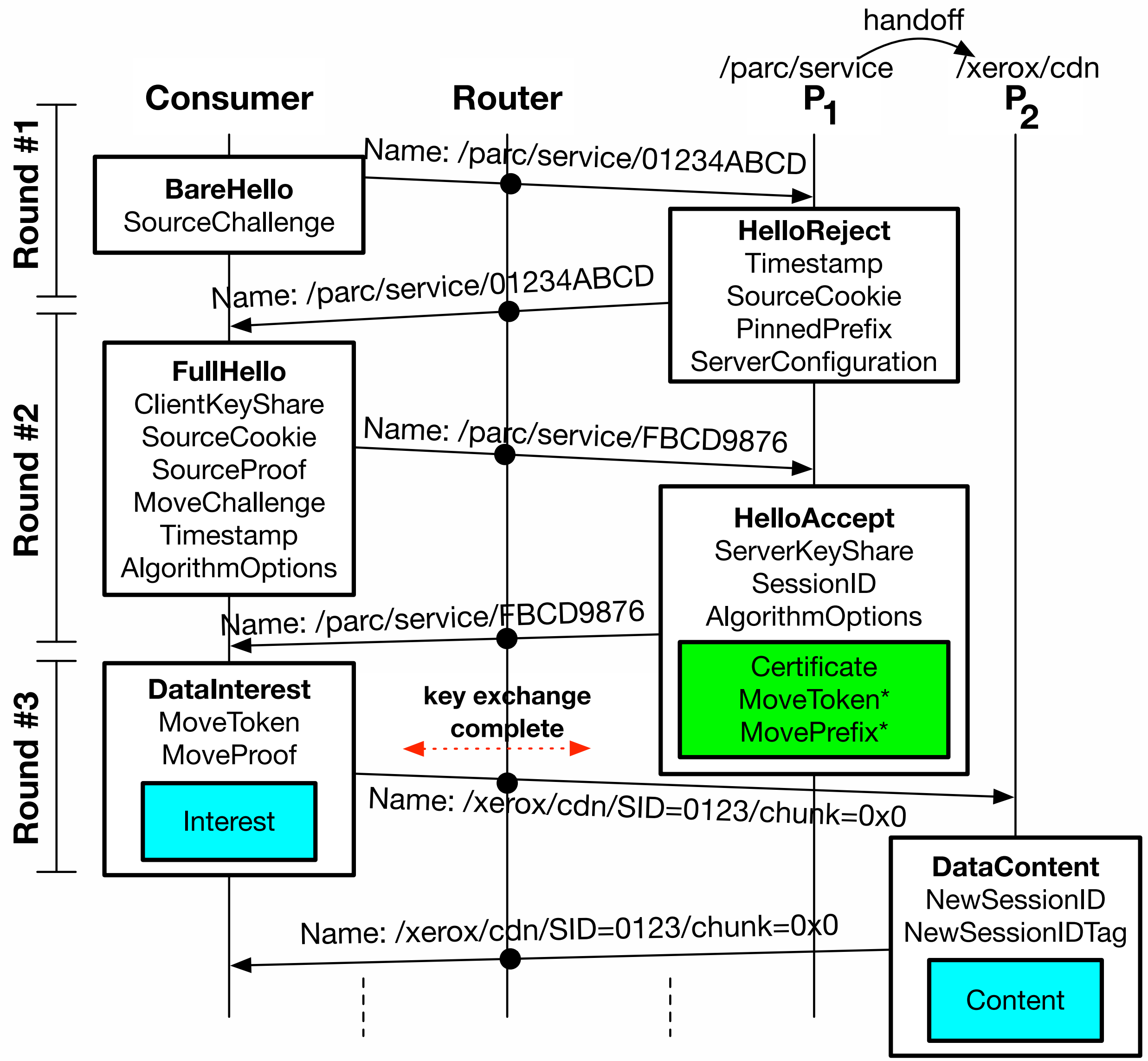
- Build a TLS-like protocol for CCN
- Key challenges:
  - How to identify sessions and ensure traffic goes end-to-end?
  - How to mitigate against volumetric DoS attacks on the producer?
  - How to apply TLS semantics to the CCN communication model (request/response)

CCNxKE

CCN-compliant key exchange (and secure session) protocol with the following features:

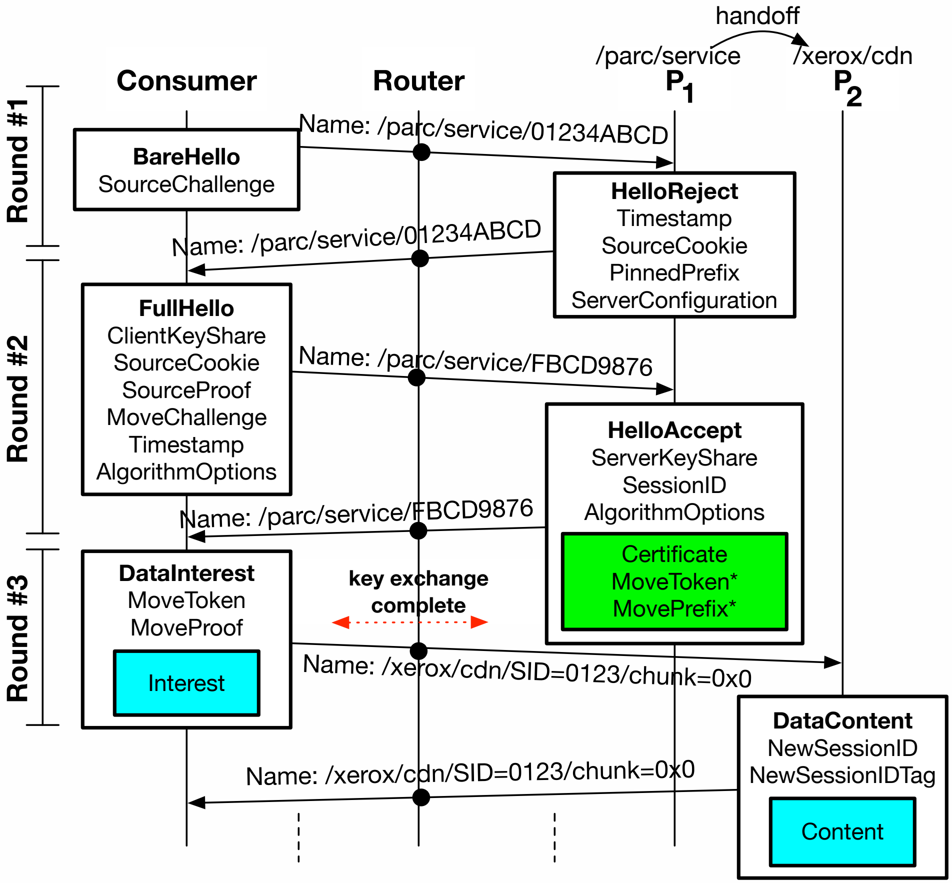
- Forward-secure key derivation
- Name-based session identifiers
- Cross-namespace session migration

CCNxKE in a Nutshell



Three Rounds

1. Origin authentication
2. Session creation
3. Session migration and data exchange



Origin Authentication

1. Generate random SourceProof and hash image

$$x \leftarrow \{0,1\}^\lambda$$

$$y := H(x)$$

2. Consumer sends y to the producer in Round 1
3. Producer computes and returns a SourceCookie

$$c = F_k(y)$$

4. Consumer sends (x, c) in Round 2
5. Producer verifies that the SourceProof matches the cookie:

$$c = F_k(H(x))$$

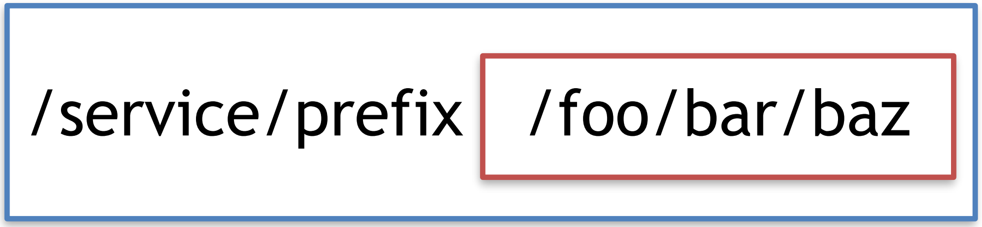
Session Migration

- Session can be migrated from producer to trusted service
- Mechanism similar to origin authentication
- MoveToken (a la SourceCookie) is an **encryption** of a traffic secret and hash of consumer-generate nonce
    - Symmetric or public key based on the trust relationship

Encapsulated Requests

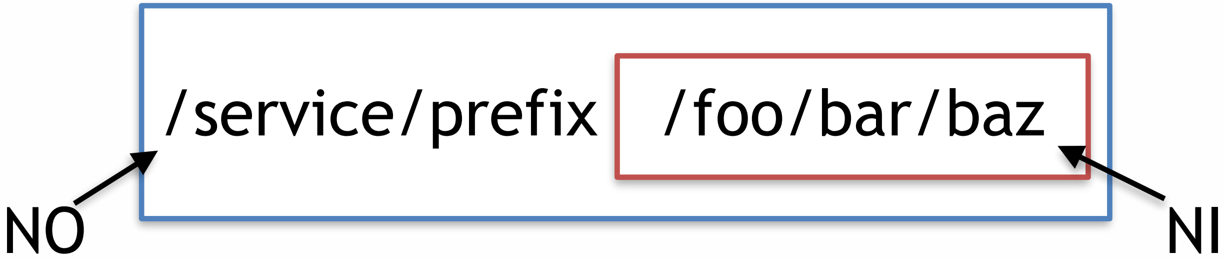
```
/foo/bar/baz
```

Encapsulated Requests

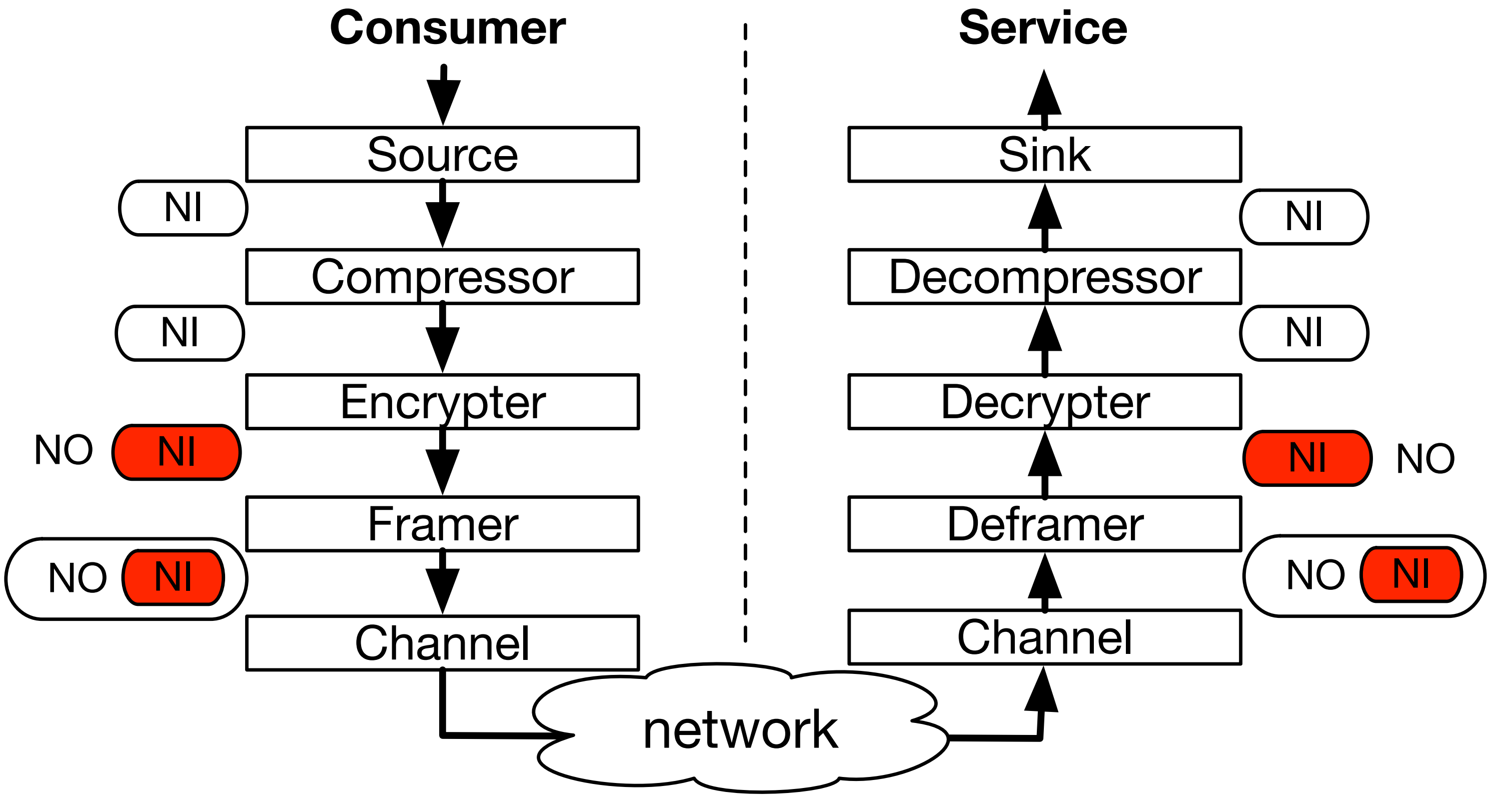


/service/prefix /foo/bar/baz

Encapsulated Requests



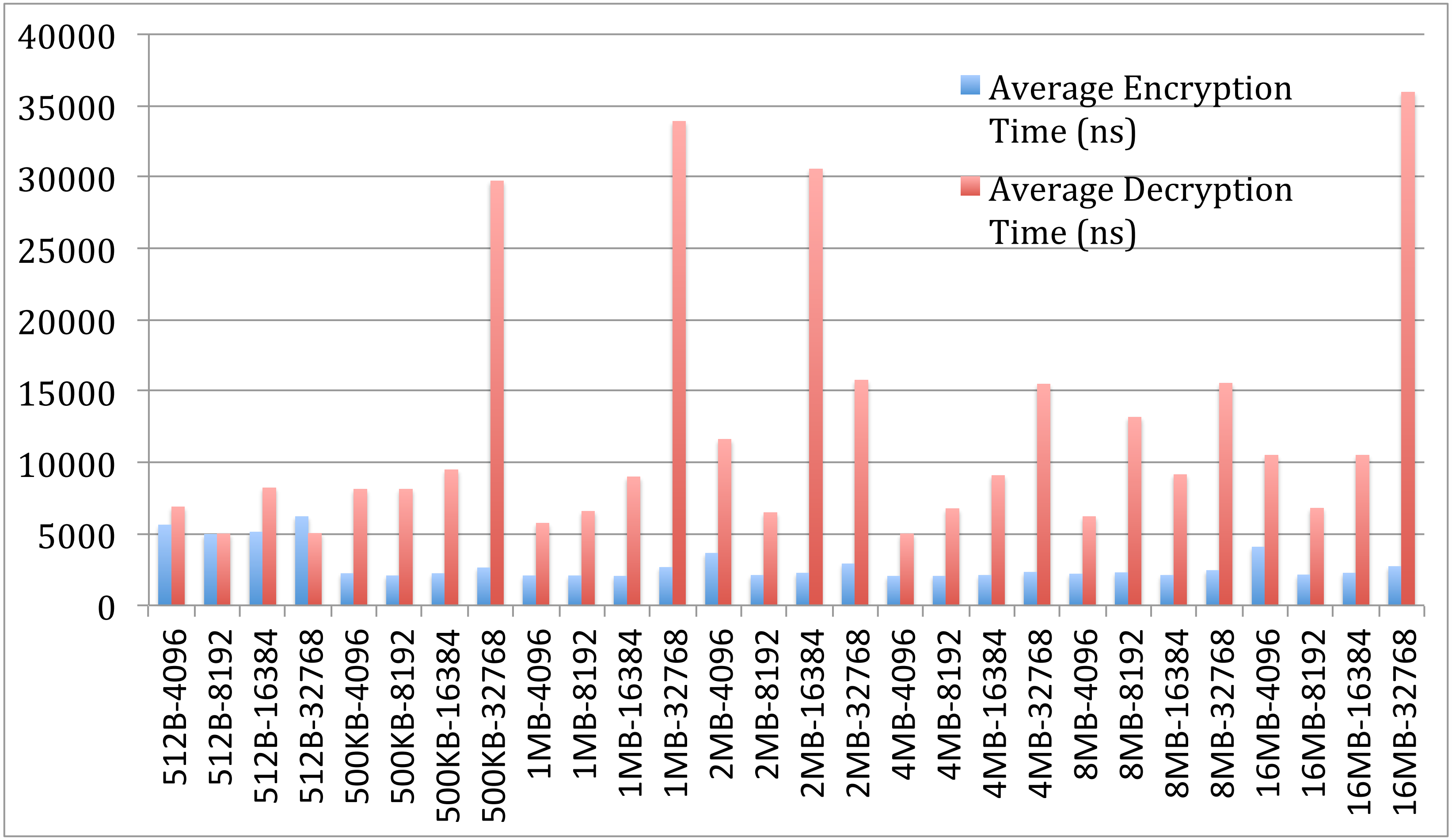
Driving the Session



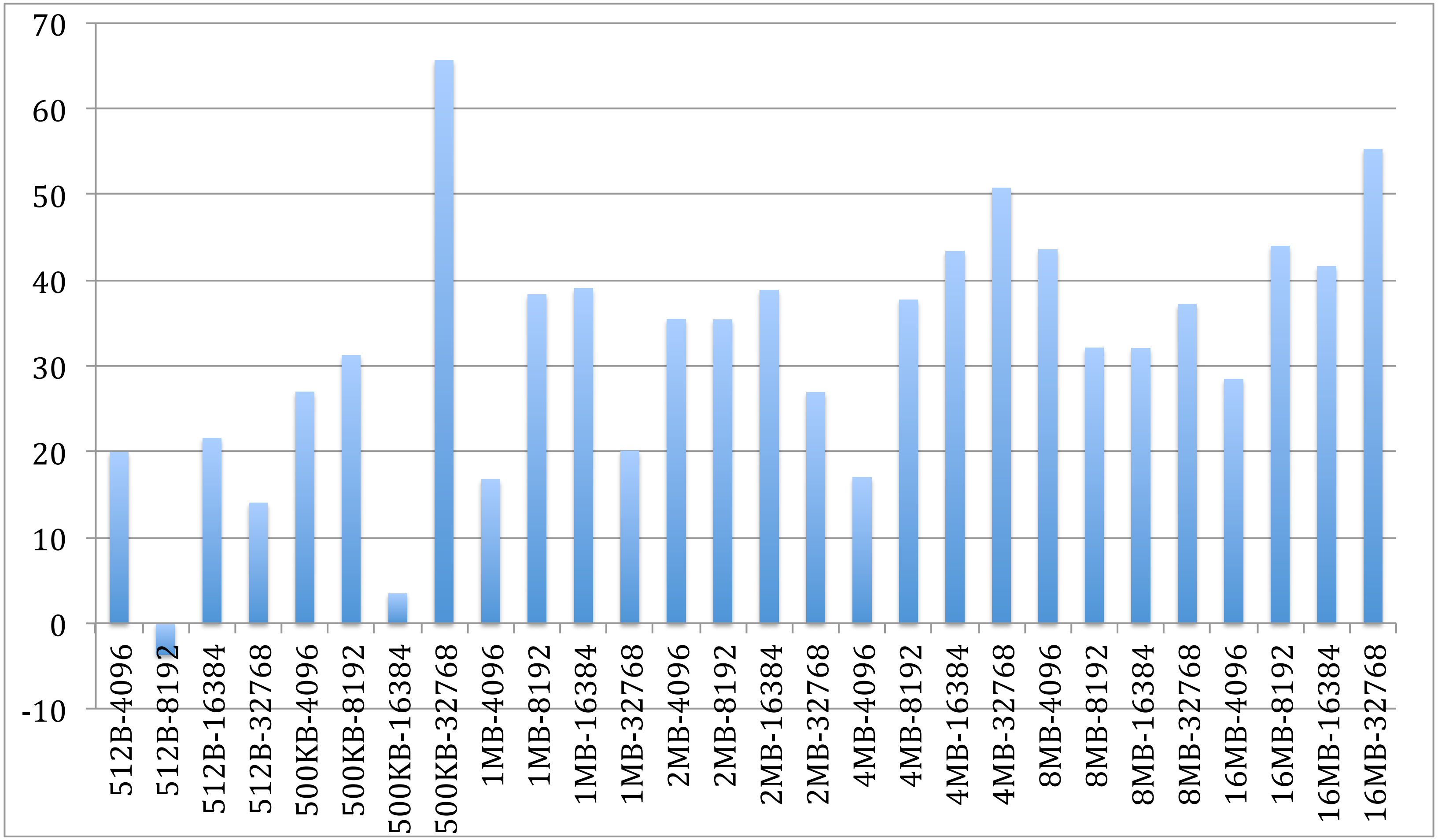
Experimental Results

- Setup: Single forwarder topology to isolate cryptographic and protocol codec overhead
- Application: transfer a large file from the producer to consumer upon request
- Transport: stop-and-wait transport protocol

Cryptographic Overhead



Data Transfer Latency (Percentage Increase)



Conclusion

- CCNxKE is a viable secure session protocol for CCN and related architectures
- CCNxKE can be used to bootstrap a shared secret for a variety of purposes:
  - Transferring sensitive keying material
  - Tunneling data from producer to consumer
- Experimental results show CCNxKE introduces only modest overhead

Future Work

- Experiment with session migration at scale
- Deploy CCNxKE and experiment with different applications:
  - Payroll, media streaming, dynamic API requests

Questions?

Fire away!