+

Name Privacy

(Routing on encrypted names)

Dagstuhl 16251

Assumptions

- No discovery, e.g., a search engine
- Content can be requested by an identifier, e.g., it's cryptographic hash digest
- Consumers know public key of producers
- Names are composed of a routable prefix, application-specific suffix, and possibly other identifiers
  - Key ID
  - Content ID

+ Terms

- Name privacy:
  - Goal: routable prefixes reveal no more than IP address and port
  - Application-specific suffix reveals nothing about the content
- Assume that revealing the content ID is not a problem

Requests with Content ID

- Requests have a locator and a content ID
- Locator can be uncorrelated to data

Locator: /akamai/

Content ID: 0x1231…

Requests without a Content ID

- Routable prefix must meet our definition
- Application-specific suffix must not reveal any information about the content. How?
  - Encrypt it.

Name Encryption

- What is the routable prefix? How does a consumer learn this?
  - Assume it does for now.
- What key is used to encrypt the suffix?
  - Producer public key (forget DoS attacks now)
- What if the result is not encrypted?
  - Possible to infer name from the data
- Outcome: result must be encrypted
  - How?

Response Encryption

- Use consumer-supplied key:
  - Eavesdroppers can use the same technique to replay the name
    - Not true if we use CCA-secure encryption
      - … but we still must learn the routable prefix.
- Anything else: eavesdroppers can use the same thing. (?)

+ Outcome

1. We need a way to discover the routable prefix.
   - Upper-layer service…
2. Response needs some form of access control
   - Upper-layer service…
3. Locator and Content ID are obtained via some other means
   - Upper-layer service…
     - Search engine
     - Session