Chained and Delegable Authorization Tokens

G. Navarro J. García J. A. Ortega-Ruiz

Dept. of Computer Science
Universitat Autònoma de Barcelona

NordSec 2004

UAB

Outline

1. Introduction
2. Example
   - Initialization
   - Token delgation
   - Chain delegation
3. Delegation in CADAT
4. Implementation and Applications
   - Implementation
   - SPKI cert without using full tag intersection
   - SPKI cert using full tag intersection
5. Conclusions

Outline

1. Introduction
2. Example
   - Initialization
   - Token delgation
   - Chain delegation
3. Delegation in CADAT
4. Implementation and Applications
   - Implementation
   - SPKI cert without using full tag intersection
   - SPKI cert using full tag intersection
5. Conclusions

UAB

Chained And Delegable Authorization Tokens

- Hash chains as chains of authorization tokens.
  - tokens represent generic authorizations (not just micropayments).
- Delegation
  - delegation of chains or subchains.
- Implemented with a **trust management** infrastructure.

CADAT

Chained **A**nd **D**elegable **A**uthorization **T**okens

Chained And Delegable Authorization Tokens

- Hash chains as chains of authorization tokens.
  - tokens represent generic authorizations (not just micropayments).
- Delegation
  - delegation of chains or subchains.
- Implemented with a **trust management** infrastructure.

CADAT

Chained **A**nd **D**elegable **A**uthorization **T**okens

Chained And Delegable Authorization Tokens

- Hash chains as chains of authorization tokens.
  - tokens represent generic authorizations (not just micropayments).
- Delegation
  - delegation of chains or subchains.
- Implemented with a **trust management** infrastructure.

CADAT

Chained **A**nd **D**elegable **A**uthorization **T**okens

Chained And Delegable Authorization Tokens

- Hash chains as chains of authorization tokens.
  - tokens represent generic authorizations (not just micropayments).
- Delegation
  - delegation of chains or subchains.
- Implemented with a **trust management** infrastructure.

CADAT

Chained **A**nd **D**elegable **A**uthorization **T**okens

UAB

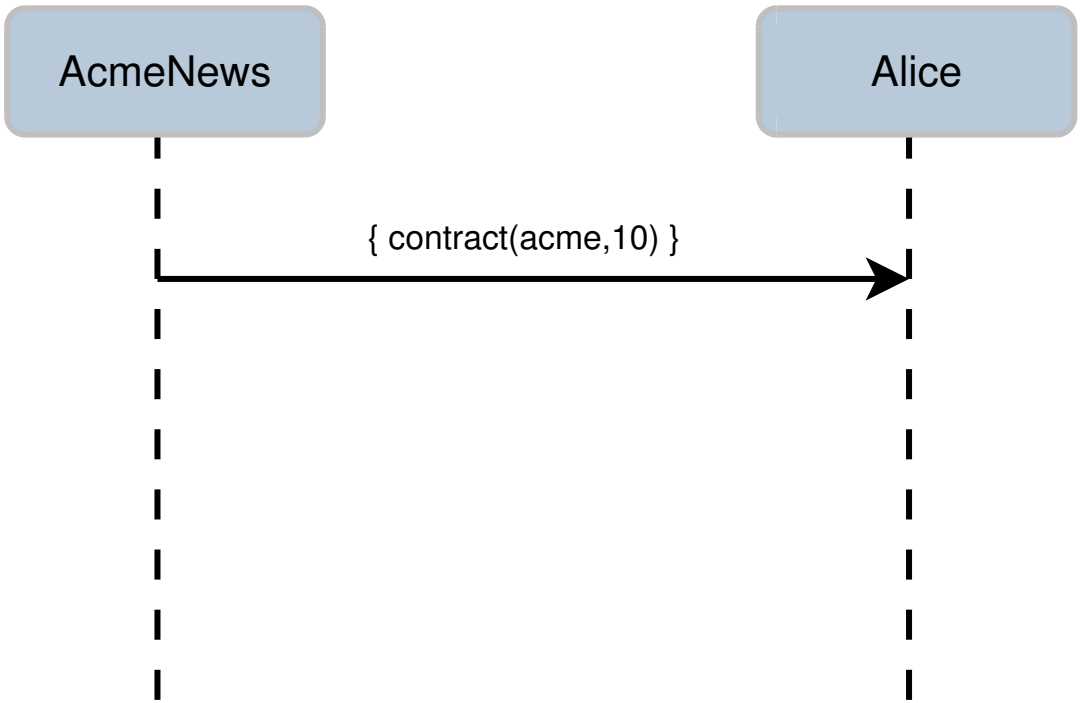
Chained And Delegable Authorization Tokens

- Hash chains as chains of authorization tokens.
  - tokens represent generic authorizations (not just micropayments).
- Delegation
  - delegation of chains or subchains.
- Implemented with a **trust management** infrastructure.

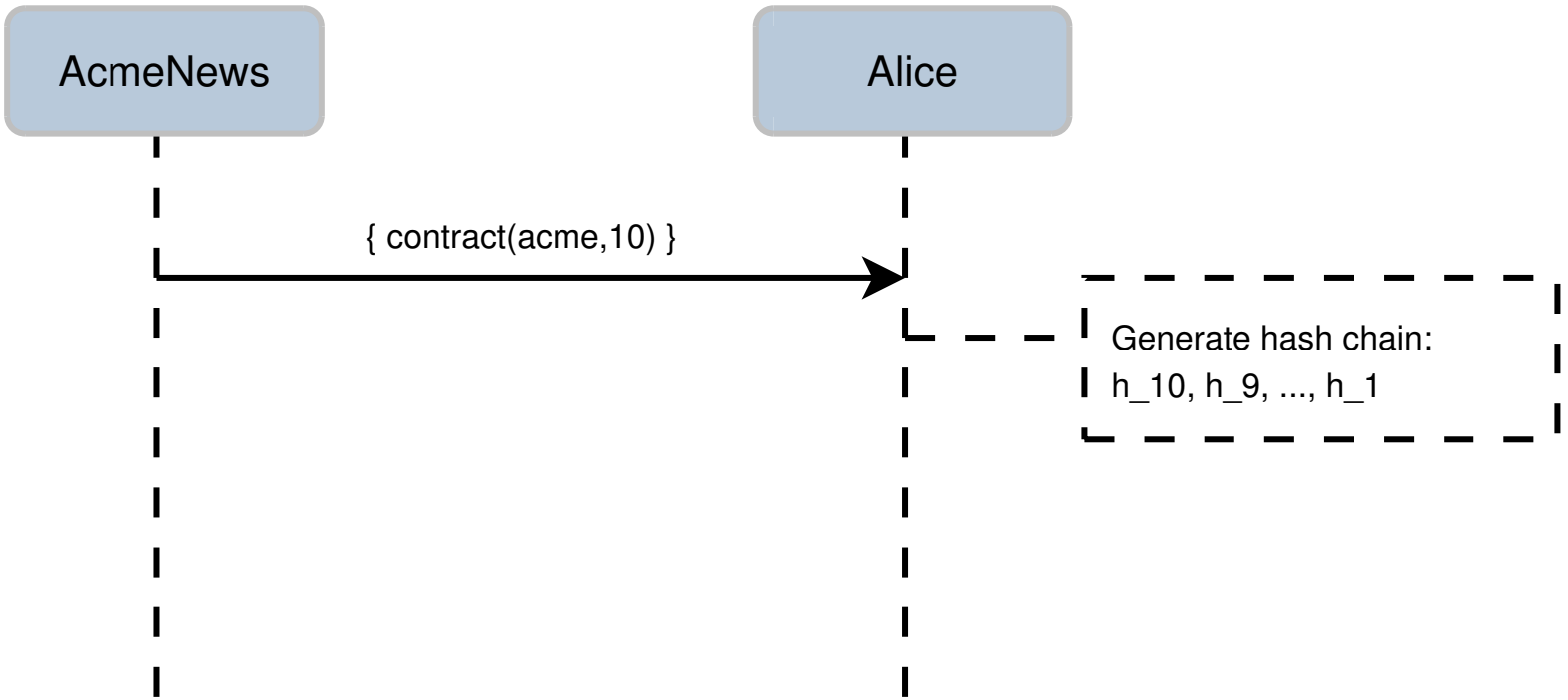
CADAT

Chained **A**nd **D**elegable **A**uthorization **T**okens

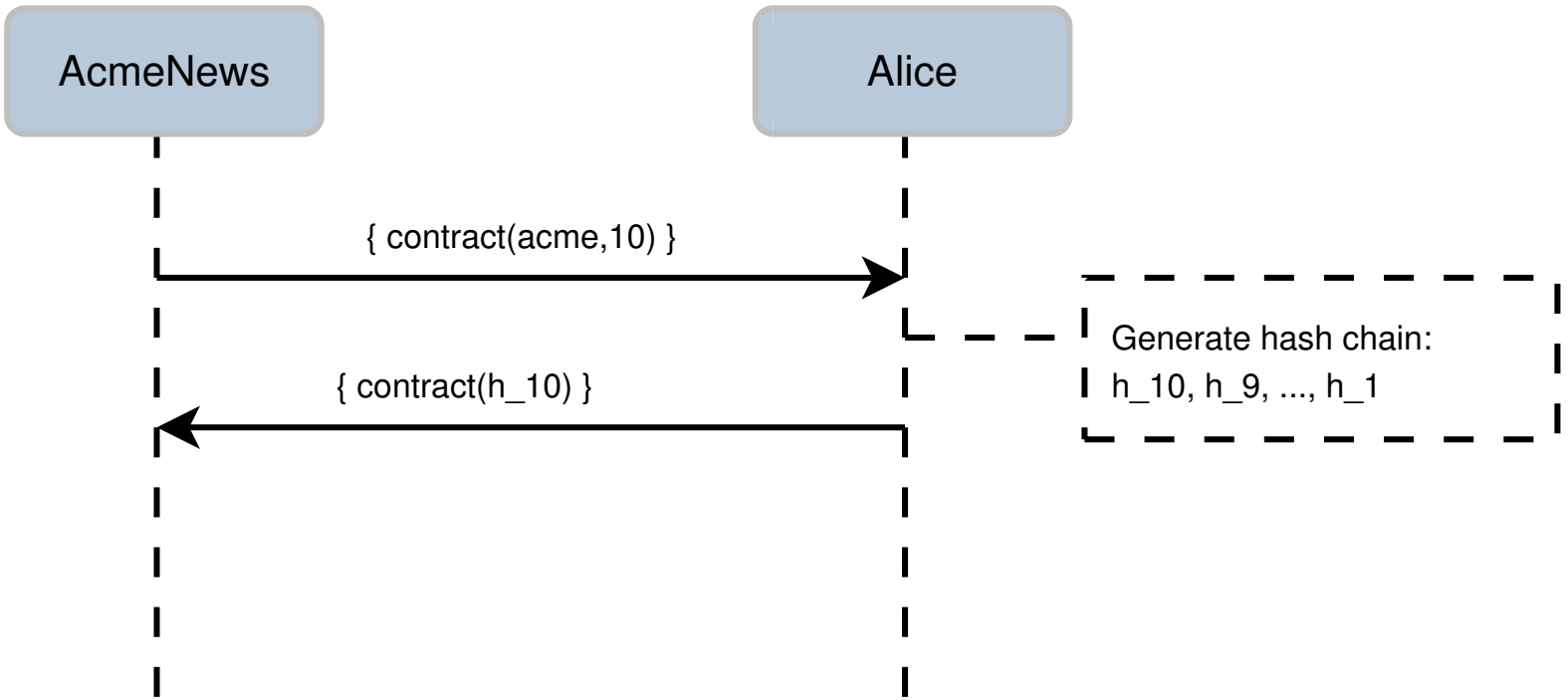
Example: first use



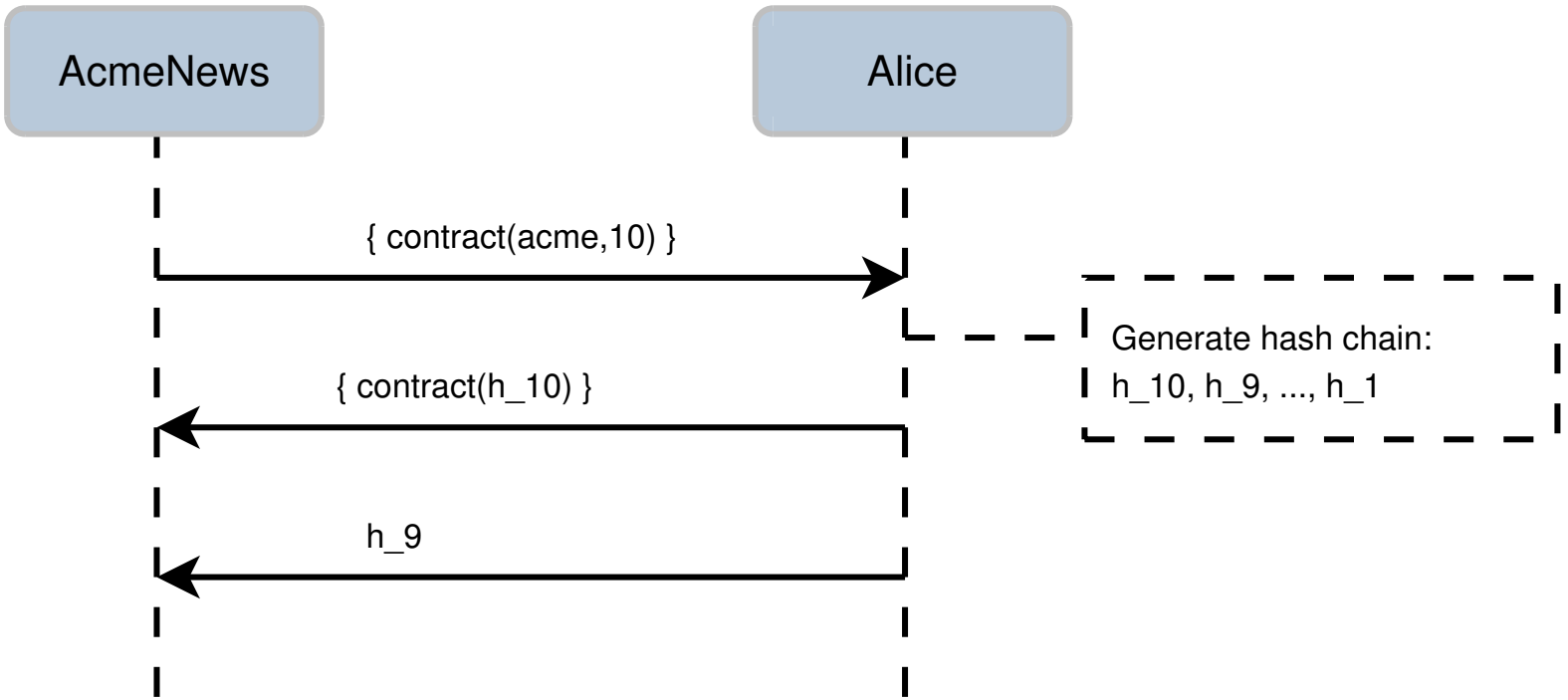
Example: first use



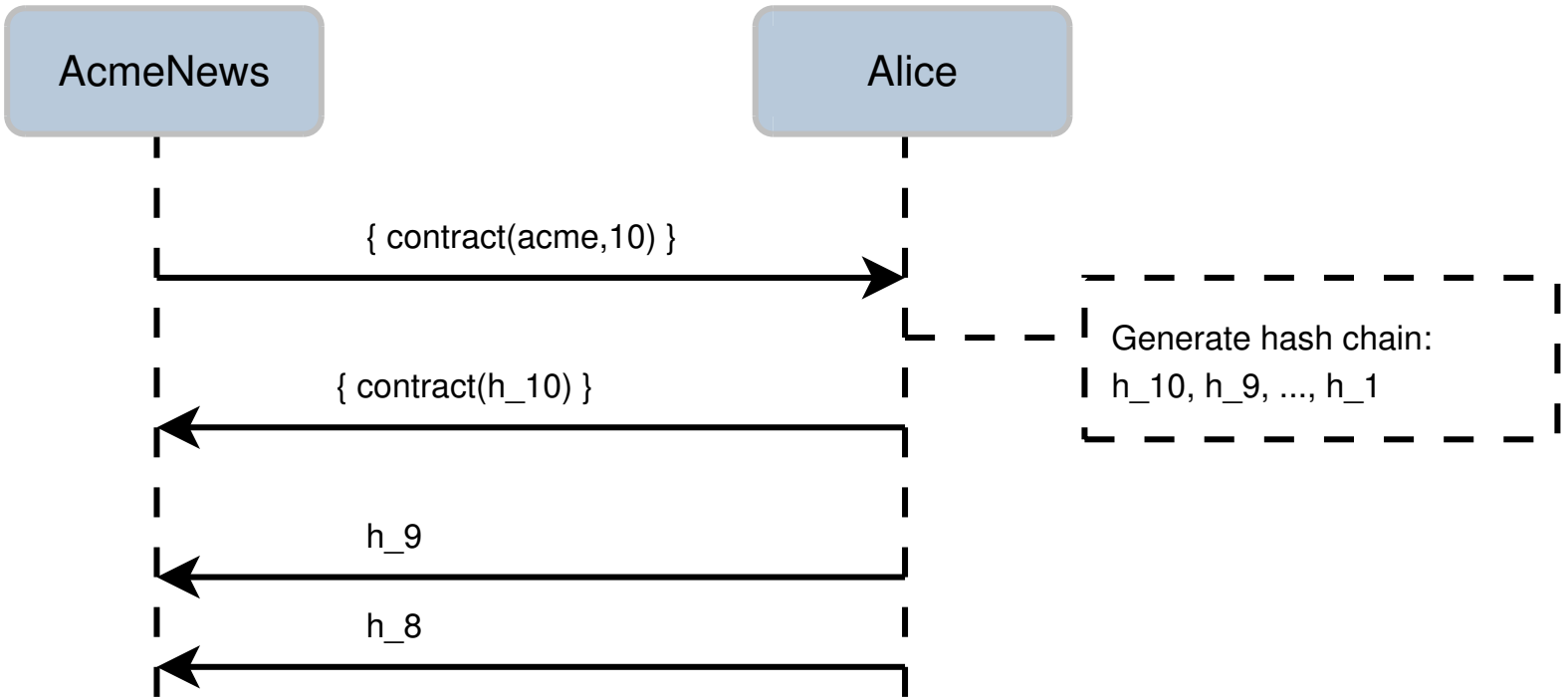
Example: first use



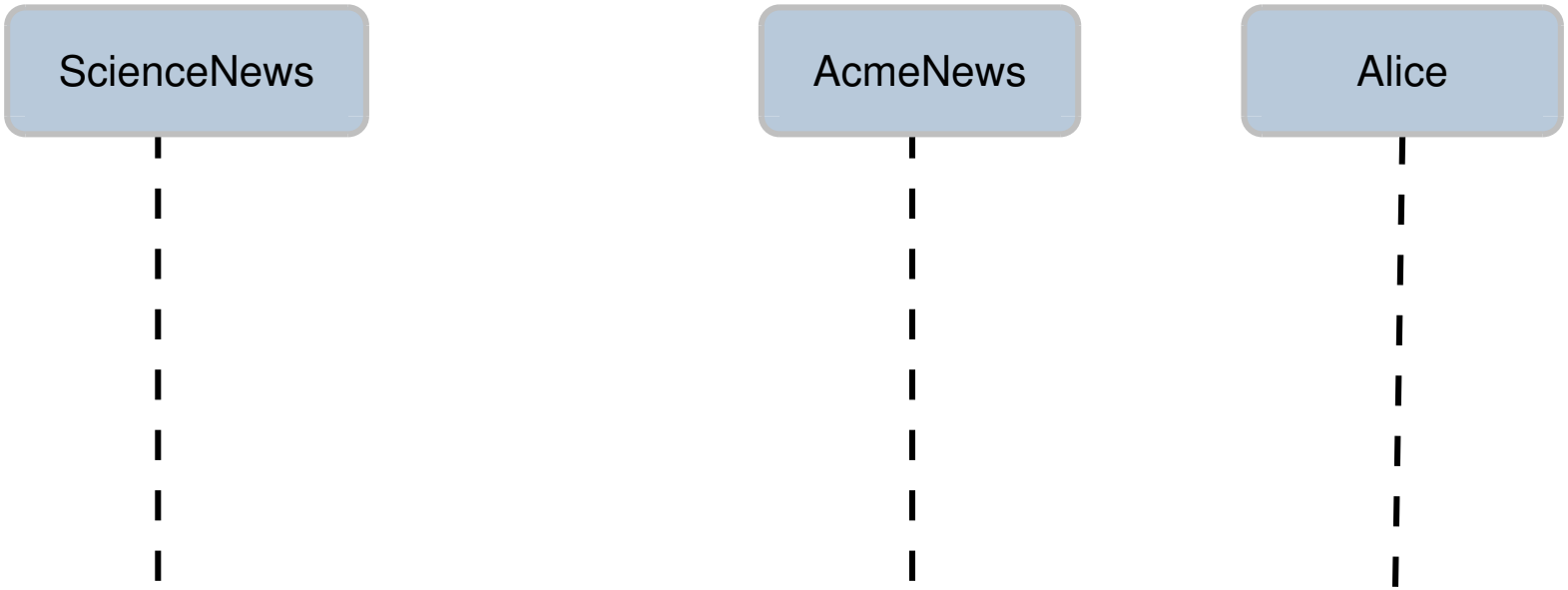
Example: first use



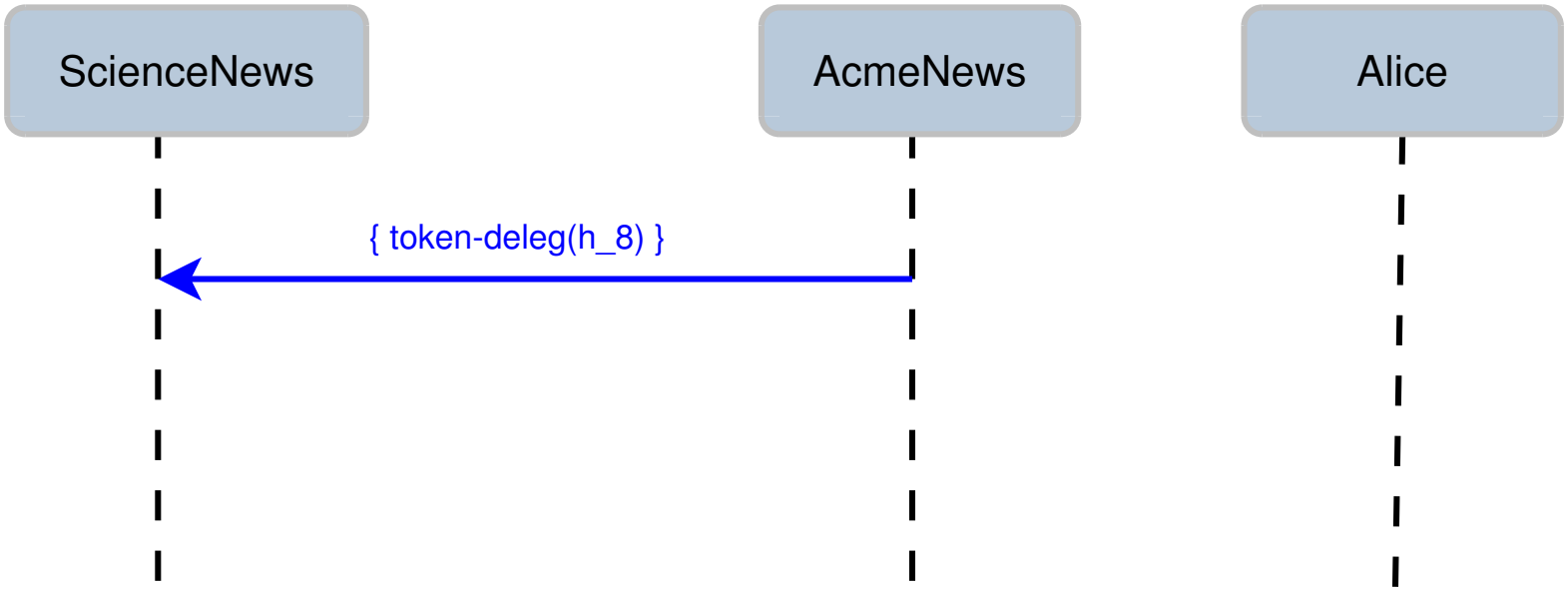
Example: first use



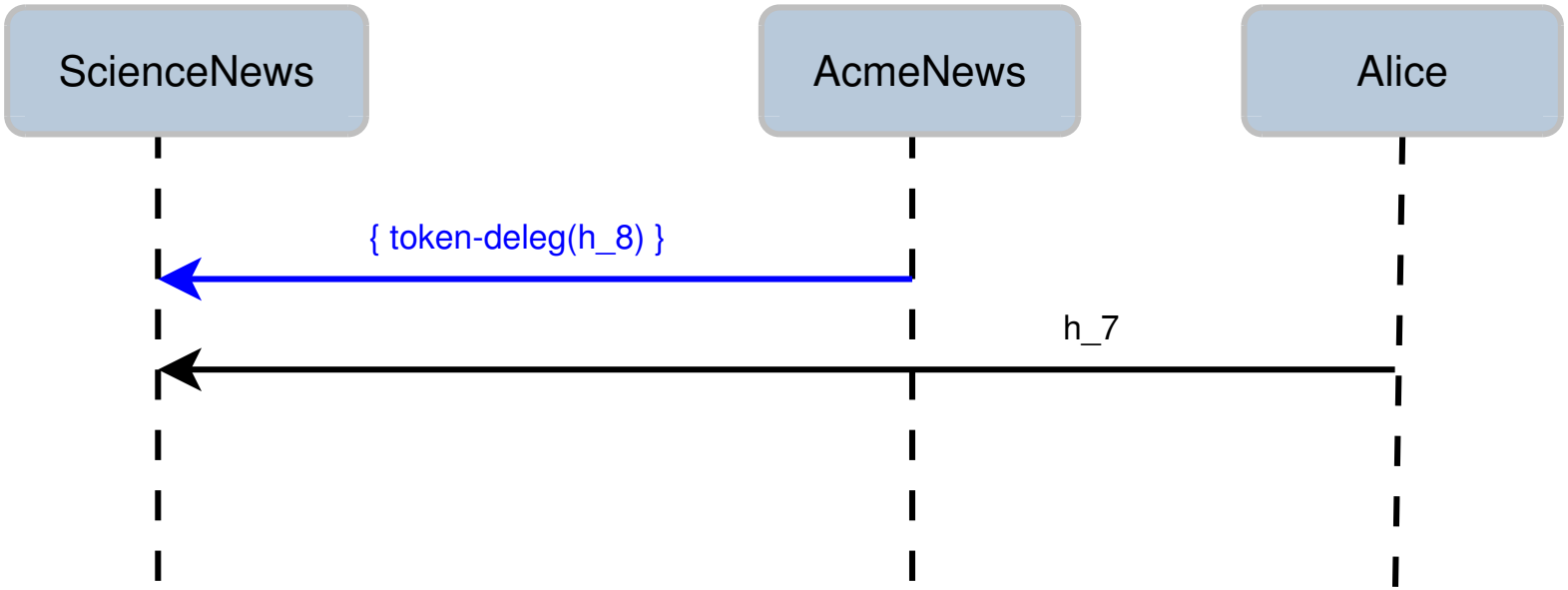
Example: token delegation



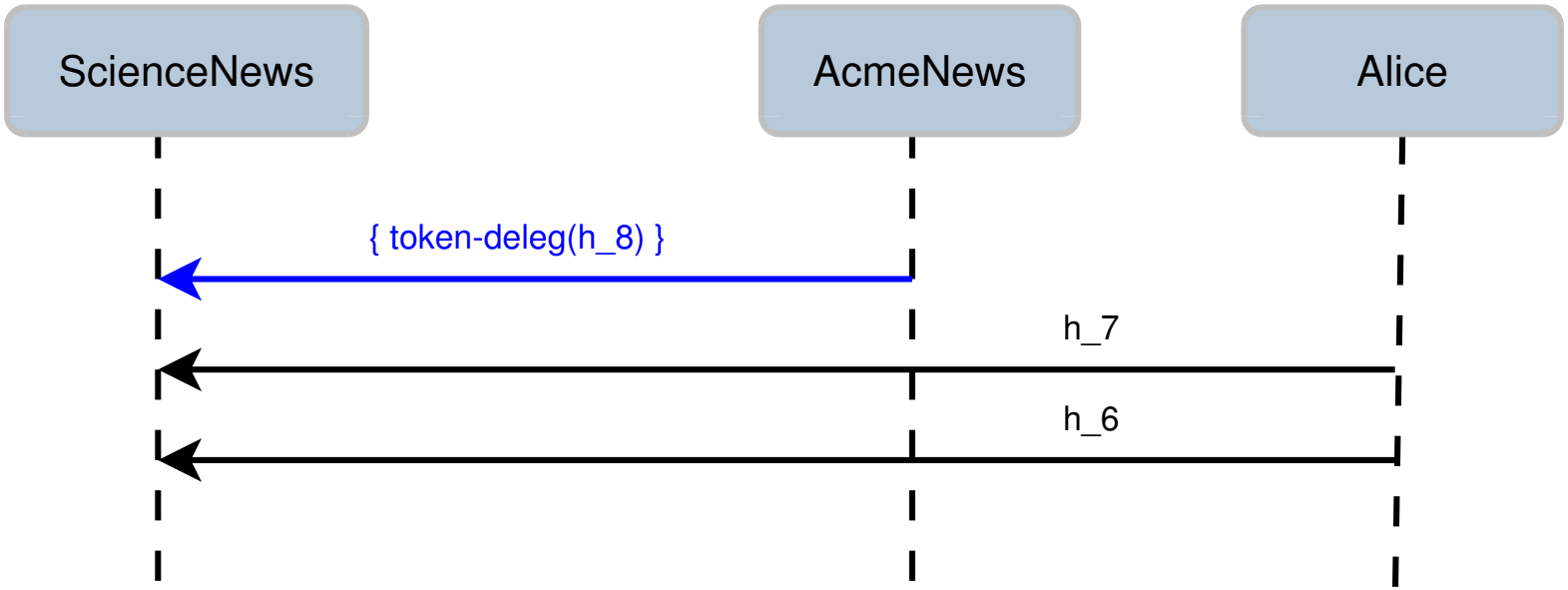
Example: token delegation



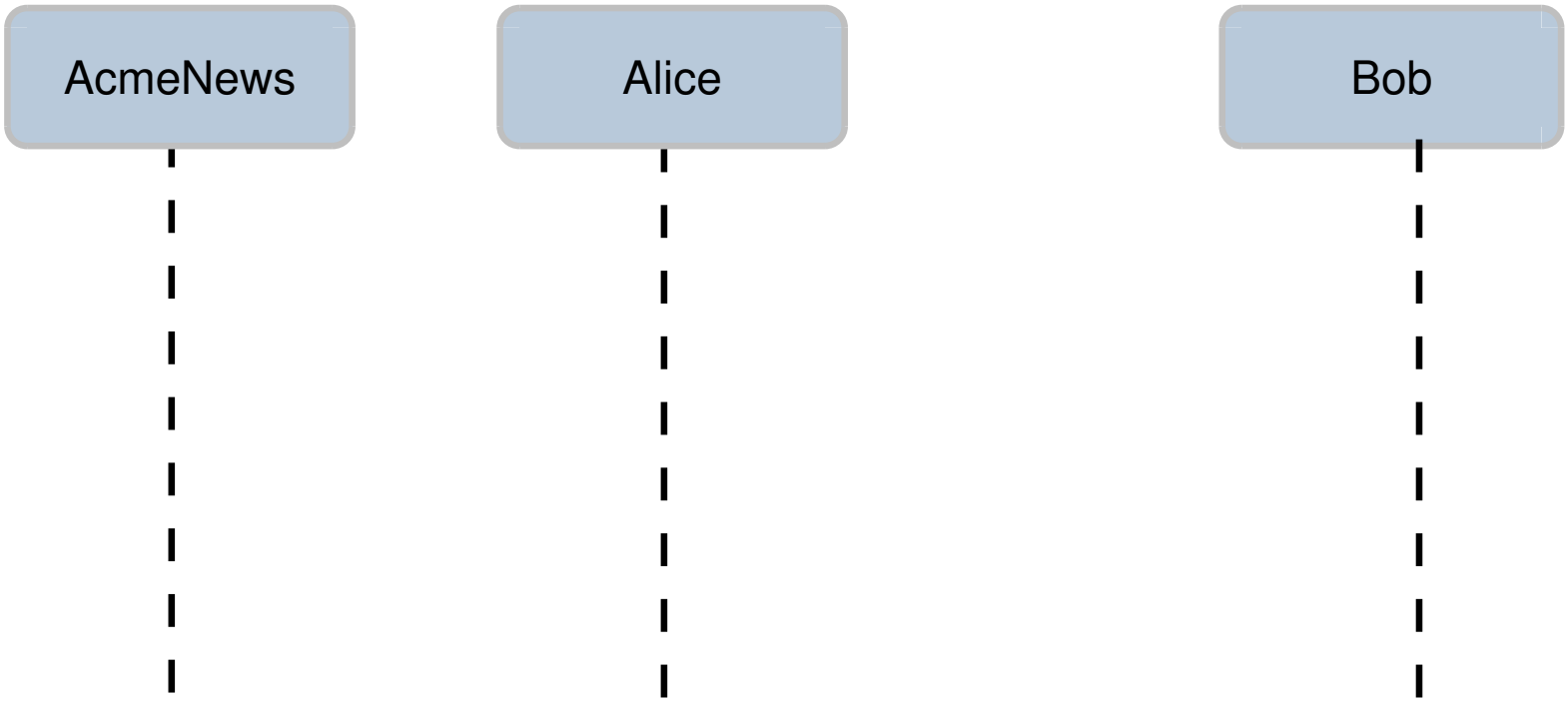
Example: token delegation



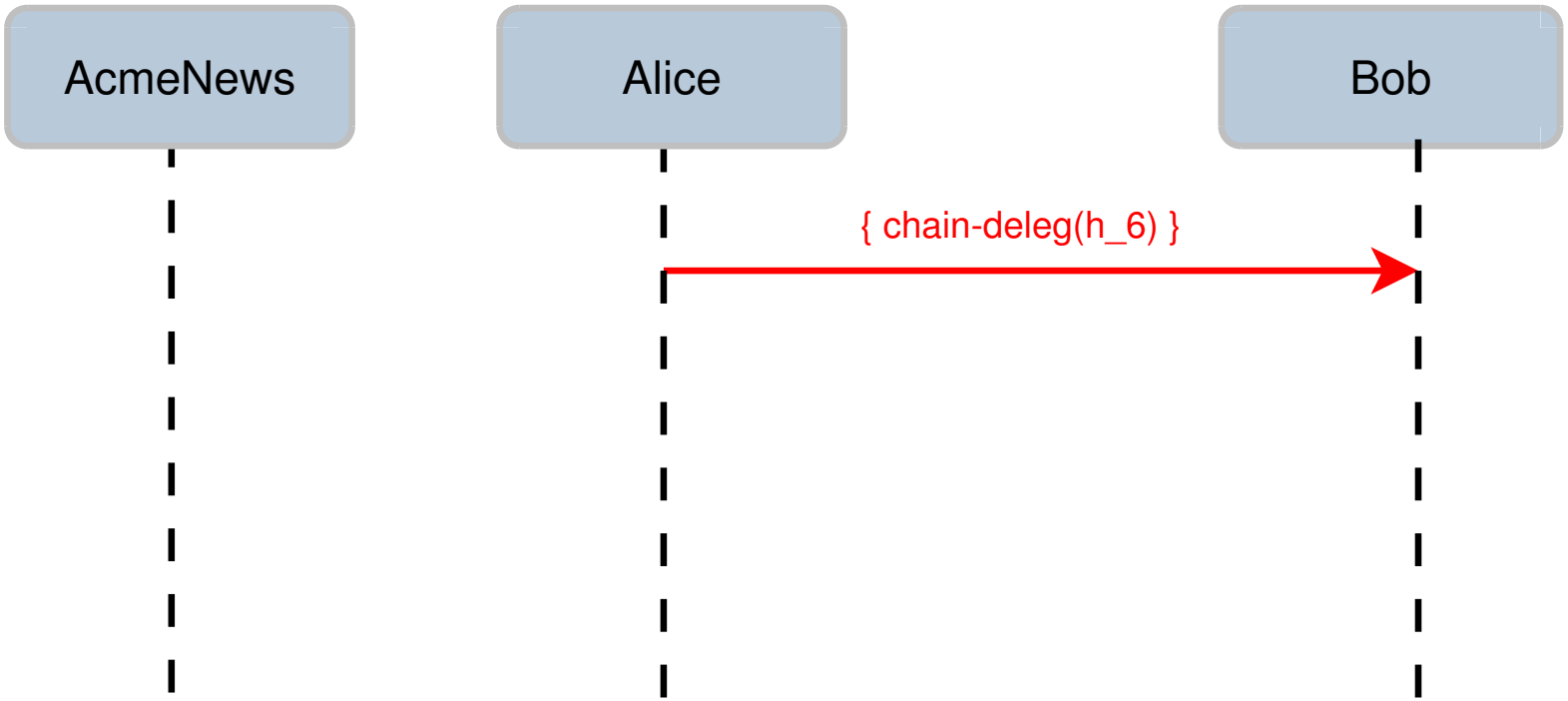
Example: token delegation



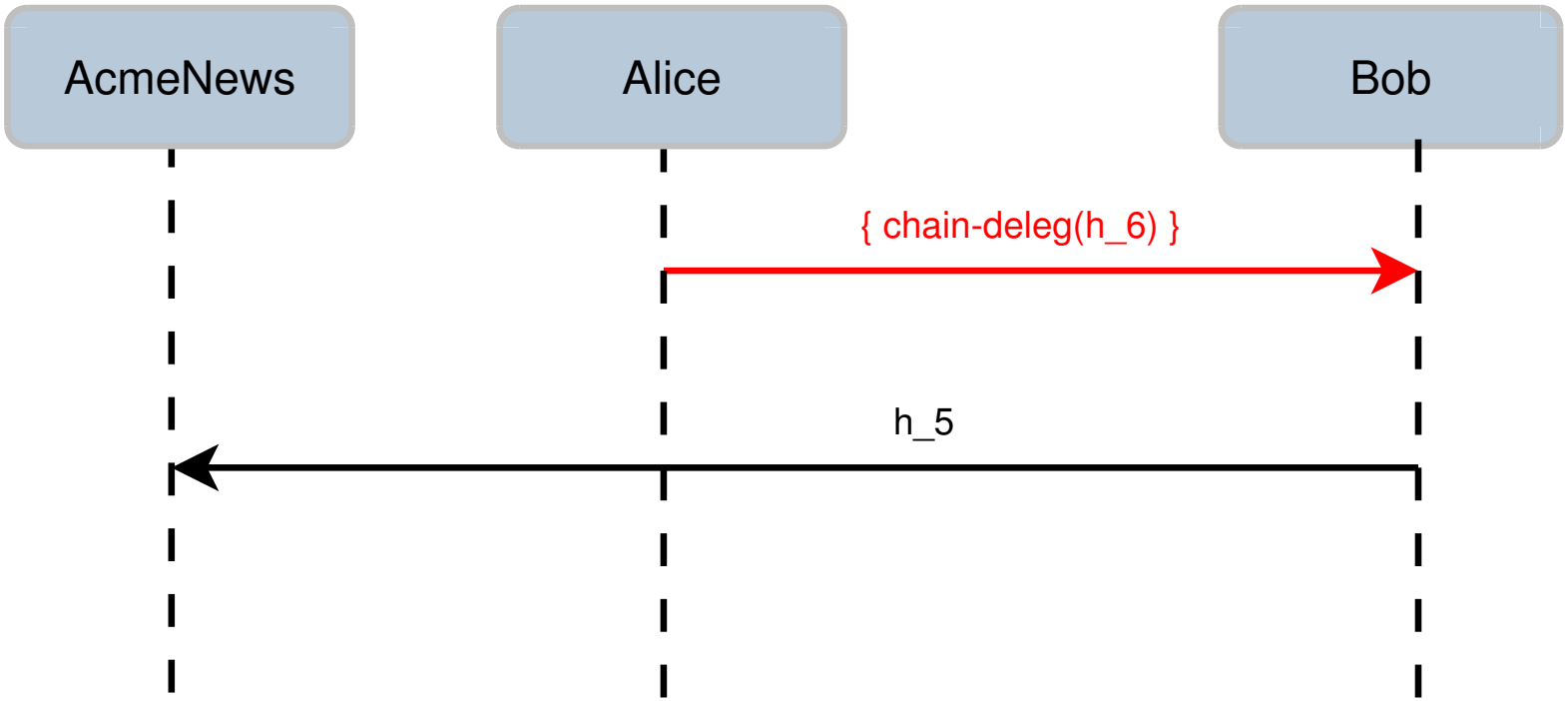
Example: chain delegation



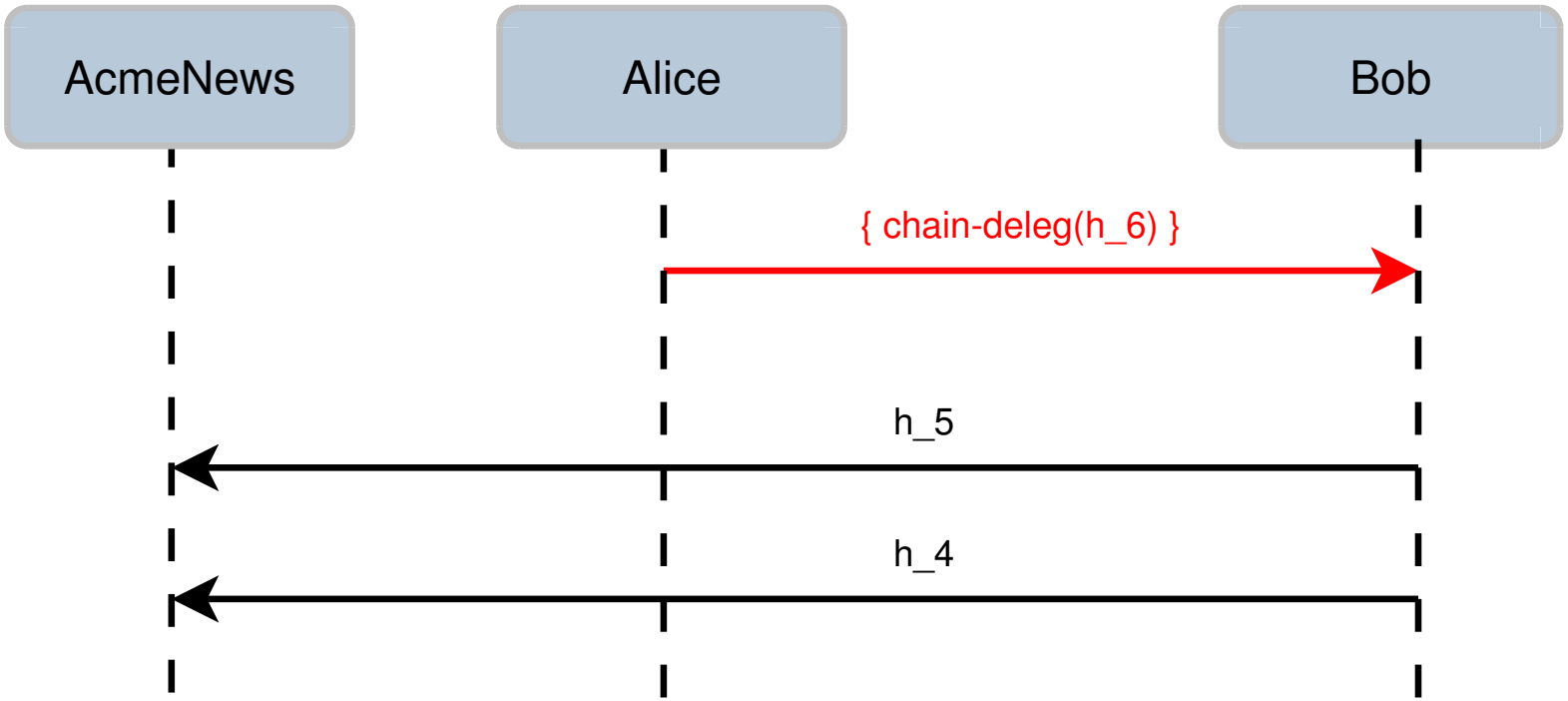
Example: chain delegation



Example: chain delegation



Example: chain delegation



CADAT & Delegation

- token-delegation: delegatee is the consumer of tokens, who offers the service (aka *server-side* delegation).
- chain-delegation: delegatee is the user of the tokens, who access the service (aka *client-side* delegation).

CADAT & Delegation

- token-delegation: delegatee is the consumer of tokens, who offers the service (aka *server-side* delegation).
- chain-delegation: delegatee is the user of the tokens, who access the service (aka *client-side* delegation).

Implementation

- CADAT is implemented in Java.
- Contracts and delegations encoded as **SPKI/SDSI authorization certificates**.
- Basic functionality provided by **JSDSI**;
  - Chain discovery algorithm $\Longrightarrow$ all computations needed by CATAD.
  - Extended to support hash chain verification in the algorithm.

Implementation

- CADAT is implemented in Java.
- Contracts and delegations encoded as **SPKI/SDSI authorization certificates**.
- Basic functionality provided by **JSDSI**;
  - Chain discovery algorithm $\Longrightarrow$ all computations needed by CATAD.
  - Extended to support hash chain verification in the algorithm.

Implementation

- CADAT is implemented in Java.
- Contracts and delegations encoded as **SPKI/SDSI authorization certificates**.
- Basic functionality provided by **JSDSI**;
  - Chain discovery algorithm $\Longrightarrow$ all computations needed by CATAD.
  - Extended to support hash chain verification in the algorithm.

Token as SPKI authorization certificate

Partial tag intersection

Authorization token: $p = (cid, i, h^i(m))$

Token-cert without hash verification

```
(cert
  (issuer ...)
  (subject ...)
  (tag
    (h-chain-id |123456789|)
    (h-chain-index (* range numeric ge 7)))
  (comment
    (h-val
      (hash
       md5 |899b786bf7dfad58aa3844f2489aa5bf|))))
```

Token as SPKI authorization certificate

Partial tag intersection

Authorization token: $p = (cid, i, h^i(m))$

Token-cert without hash verification

```
(cert
  (issuer ...)
  (subject ...)
  (tag
    (h-chain-id |123456789|)
    (h-chain-index (* range numeric ge 7)))
  (comment
    (h-val
      (hash
       md5 |899b786bf7dfad58aa3844f2489aa5bf|))))
```

Token as SPKI authorization certificate

Full tag intersection

Authorization token: $p = (cid, i, h^i(m))$

Token-cert with hash verification

```
(cert
  (issuer ...)
  (subject ...)
  (tag
    (hash-auth
      (hchain-id |lksjfSDFIsdfkj0sndKIShfoMSKJSD|)
      (hchain-index 15)
      (hash md5 |d52885e0c4bc097f6ba3b4622e147c30|))))
```

Token as SPKI authorization certificate

Full tag intersection

Authorization token: $p = (cid, i, h^i(m))$

Token-cert with hash verification

```
(cert
  (issuer ...)
  (subject ...)
  (tag
    (hash-auth
      (hchain-id |lksjfSDFIsdfkj0sndKIShfoMSKJSD|)
      (hchain-index 15)
      (hash md5 |d52885e0c4bc097f6ba3b4622e147c30|))))
```

Applications

- Generic token-based access control system.
- Micropayment schemes.
- Current application:
  - **Token-based access control for mobile agents**.

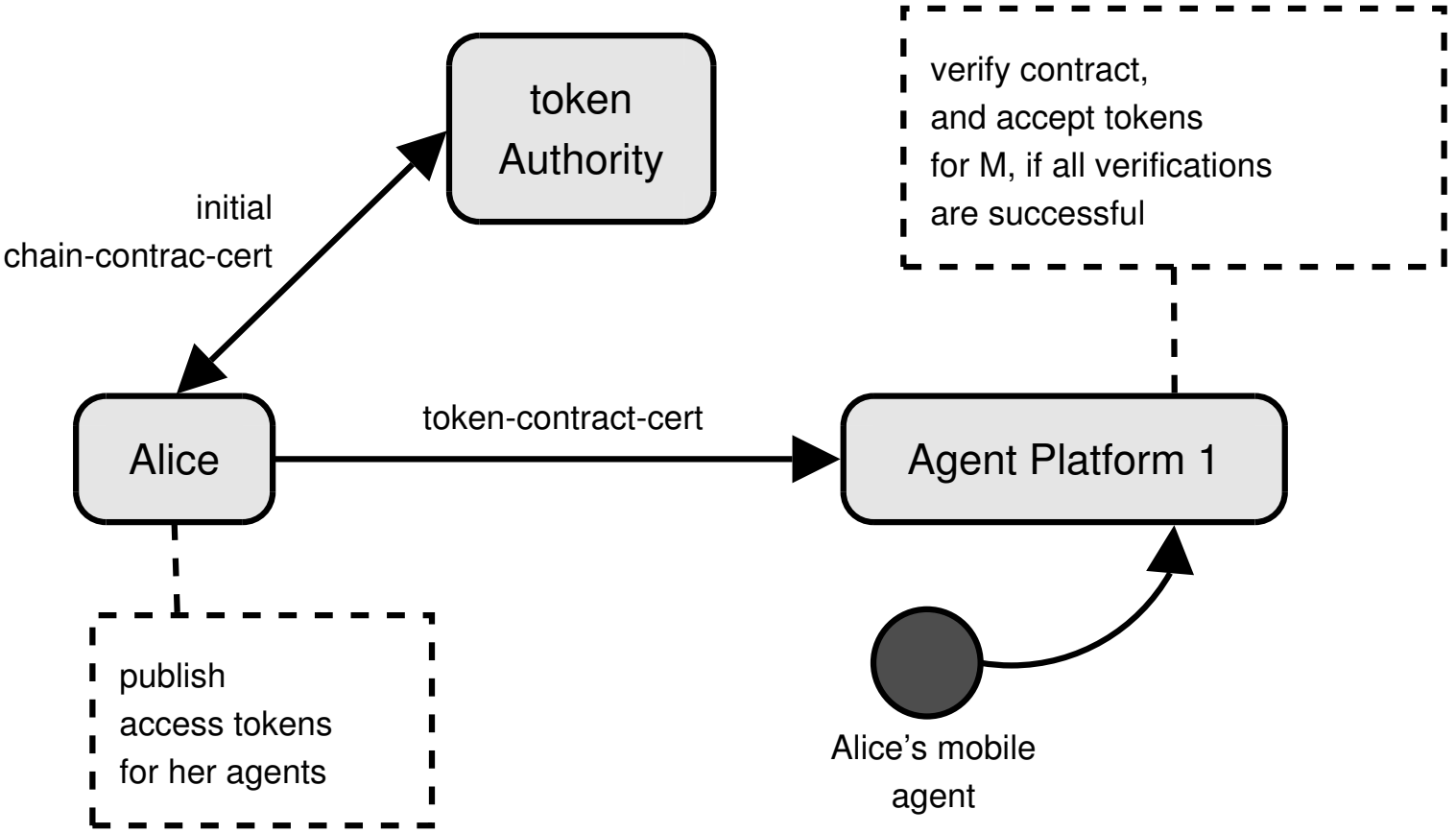
Applications

- Generic token-based access control system.
- Micropayment schemes.
- Current application:
  - **Token-based access control for mobile agents**.

Applications

- Generic token-based access control system.
- Micropayment schemes.
- Current application:
  - **Token-based access control for mobile agents**.

CADAT and mobile agent access control



Conclusions

- A system for token-based access control and micropayment systems.
  - hash chains,
  - delegation.
- Implemented with SPKI/SDSI.
- Current application: access control in mobile agent systems.

Conclusions

- A system for token-based access control and micropayment systems.
  - hash chains,
  - delegation.
- Implemented with SPKI/SDSI.
- Current application: access control in mobile agent systems.

Conclusions

- A system for token-based access control and micropayment systems.
  - hash chains,
  - delegation.
- Implemented with SPKI/SDSI.
- Current application: access control in mobile agent systems.

[C-x C-c]

Thank you! questions?