NAMED DATA NETWORKING

ACM ICN Paris 2021

Securely Deploying NDN Apps:

Security Bootstrapping with DCT Identity Bundles

Tianyuan Yu (UCLA)

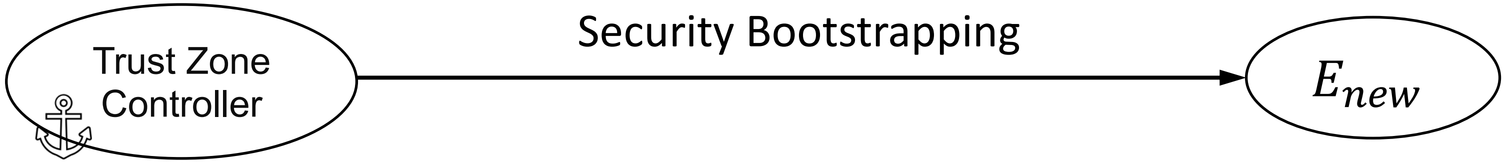
Tutorial: Power of Trust Schemas for Easy and Secure Deployment of NDN Applications

Exploring Problem Space in Security Bootstrapping

- Case-1: Bootstrapping local entities in secured environment (previous speaker)
  - Making and installing identity bundle out-of-band
  - DCT makes and installs the bundle by command line tools
  - Direct/Physical access achieves the mutual authentication by forming a secured environment
- More entities need to be bootstrapped within unsecured environment
  - **Case-2: Bootstrapping local entities in unsecured environment**
  - **Case-3: Bootstrapping remote entities**

Case-2: Bootstrapping Entities in Unsecured Local Environment

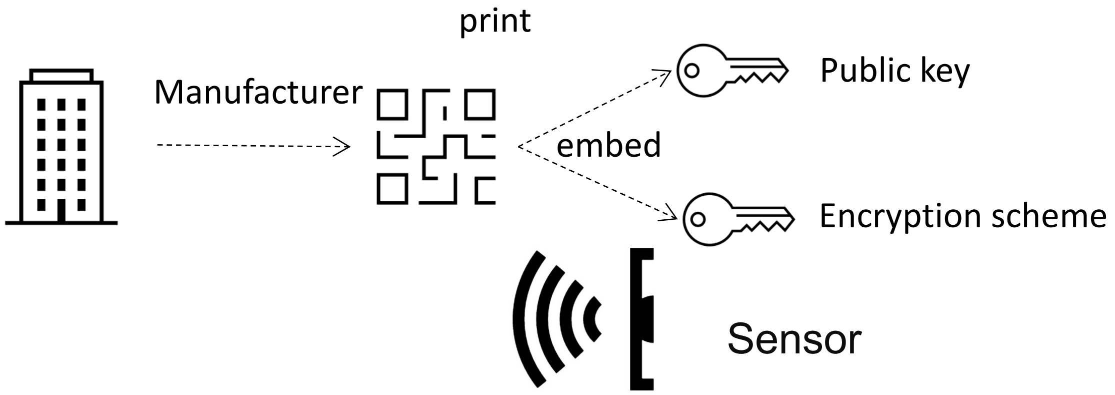
- Different from case-1: Network environment is unsecured
- Same as case-1, Trust Zone Controller and the new entity E_{new} are at local
  - e.g., one-hop wireless communication range



Unsecured Environment

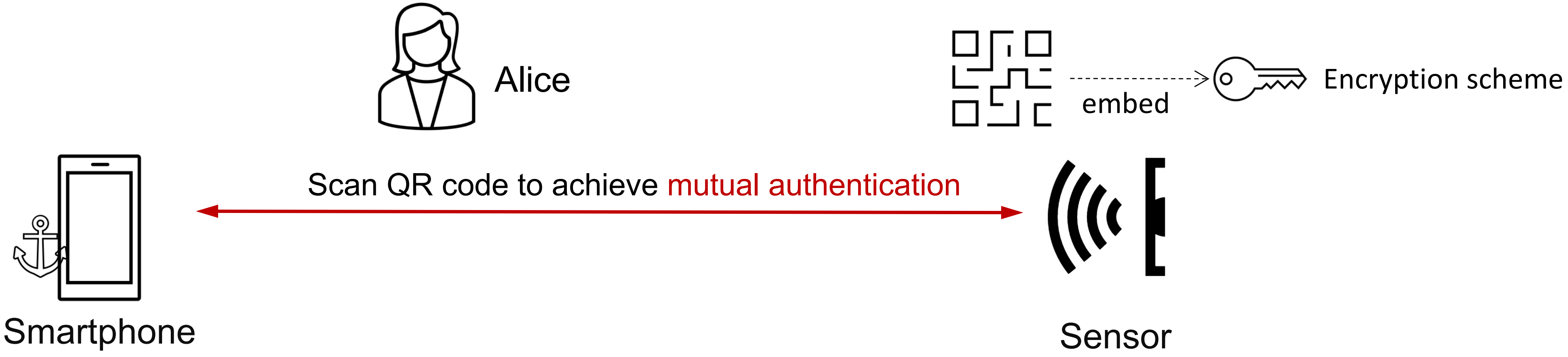
Bootstrapping E_{new} within Physical Vicinity

- IoT devices usually come with BAR code or QR code
  - Manufacturers can encode necessary information into it to facilitate bootstrapping
- BAR/QR code may contain URL to the manufacturer, device public key, temporary encryption scheme, …
- Device owner scans BAR/QR code to initiate bootstrapping



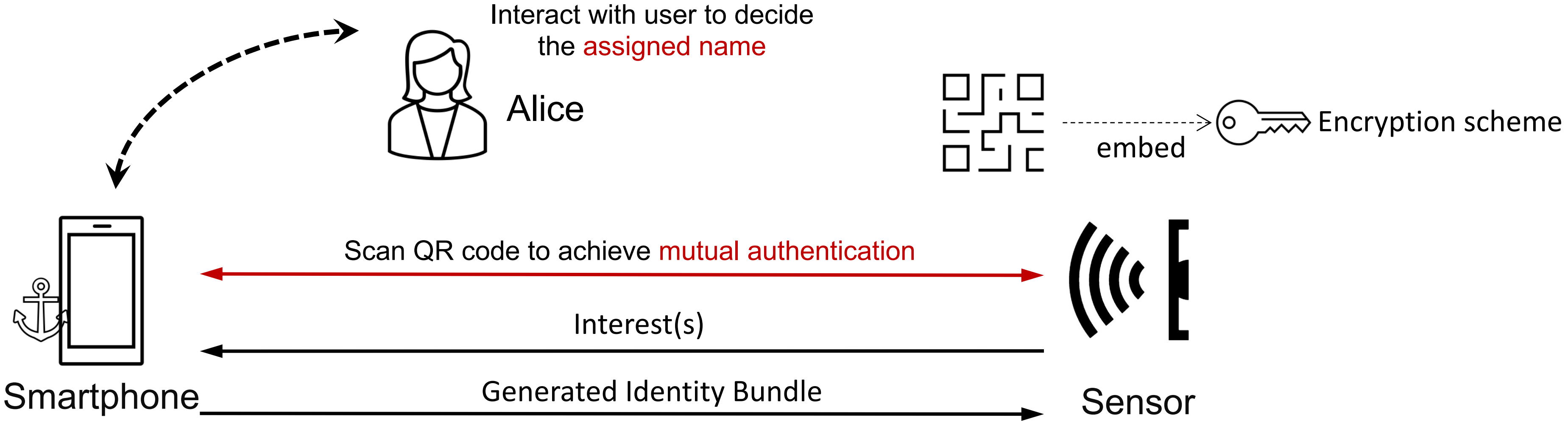
Mutual Authentication by Secured Channel

- Simple case: QR code contains a temporary encryption key
- Sensor authenticates smartphone for it communicating with the encryption key
  - The physical vicinity (e.g., < 1m) limits only the Smartphone can obtain this key
- Smartphone authenticates sensor for it communicating with the encryption key
- Other cases achieve the same goal of mutual authentication



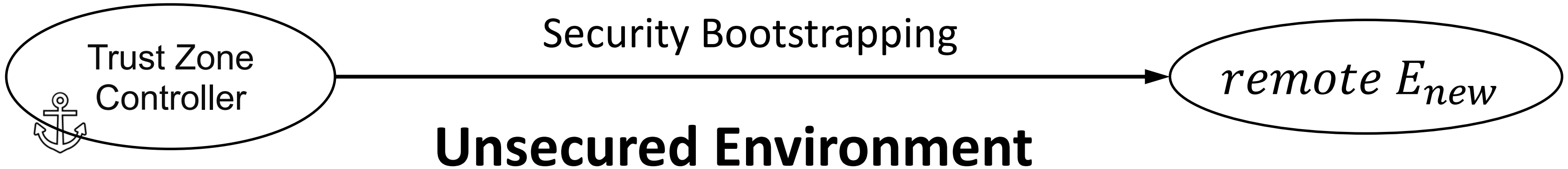
Obtain Identity Bundle in Secured Channel

- Then Alice’s smartphone can bootstrap sensor app in secured channel
  - Smartphone generates Identity Bundle for the sensor app
  - End-to-end encryption provides the communication security of the bundle


Interact with user to decide
the assigned name
Alice
embed
Encryption scheme
Scan QR code to achieve mutual authentication
Interest(s)
Generated Identity Bundle
Smartphone
Sensor

Case-3: Bootstrapping remote E_{new}

- Different from case-1 and case-2: E_{new} is remote
  - e.g., a remote application instance
  - Communication channel between the two is unsecured



Unsecured Environment

Bootstrapping Remote E_{new} via Existing Authentications

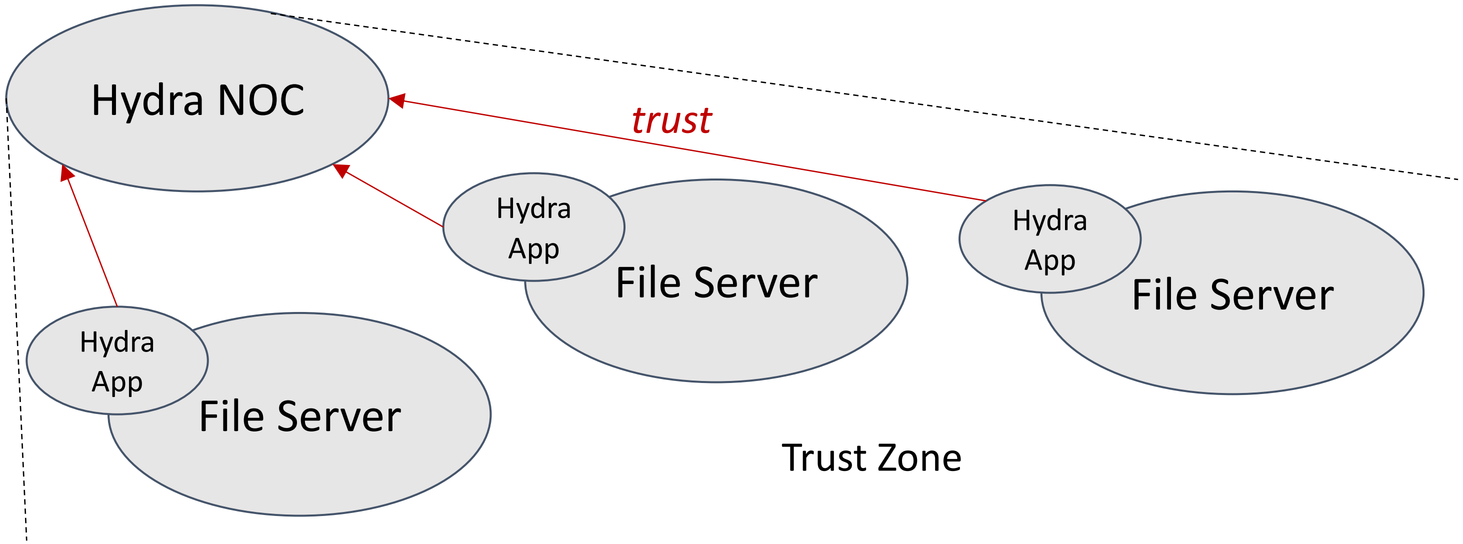
- Trust Zone Controller can only reach *remote* E_{new} over TCP/IP connectivity
- To achieve mutual authentication between Trust Zone Controller and E_{new},
  - We look into leveraging existing trust relations and authentications solutions
- Multiple such solutions exist in today's Internet
  - Certificate Authority system (CAs), DNSSEC, Single Sign-On (SSO), …

Bootstrapping Remote E_{new}

- Assuming E_{new} is an NDN app running on user's computer
- Trust Zone Controller authenticates E_{new}
  - If the current app user is authenticated
- E_{new} authenticates Trust Zone Controller
  - Built-in during software distribution
  - App package for installation can contain the trust anchor and initial trust schema
    - Initial trust schema enforces the Identity Bundle must be signed by the trust anchor
    - Therefore, E_{new} can validate the Identity Bundle received later
  - Trust Zone Controller's authenticity is assured by today's web security support
    - For example, if Alice fetches her app package from a Github URL
    - Github's CA DigiCert assures the authenticity

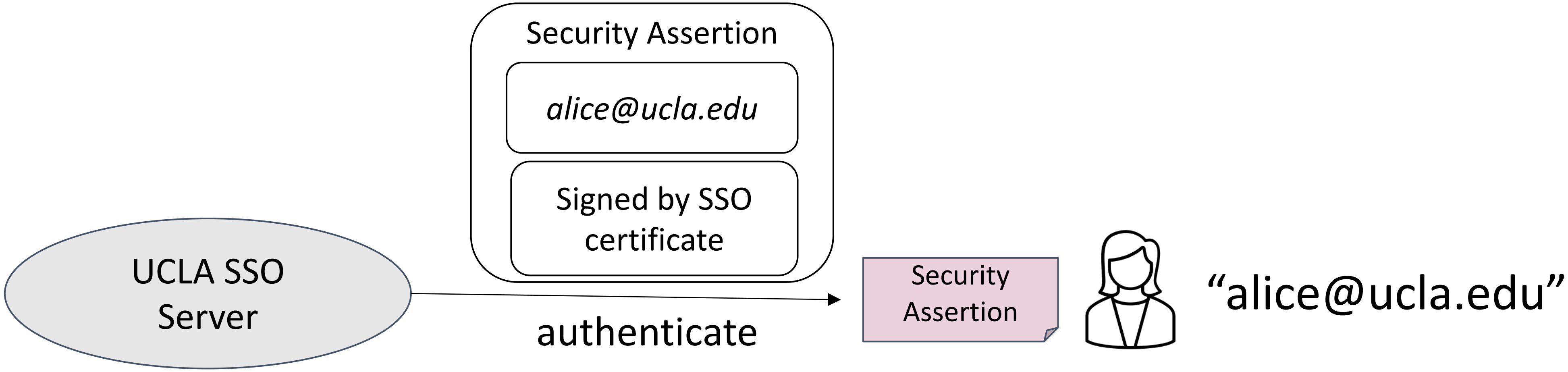
Remote Security Bootstrapping: An Example

- Hydra is an ongoing federated storage project
- Different organizations contribute file servers and share data
  - Users installs Hydra app on contributed file servers
  - Hydra Networking Operating Center (NOC) serves as Trust Zone Controller for “/hydra”
- The remote Hydra apps need authentication



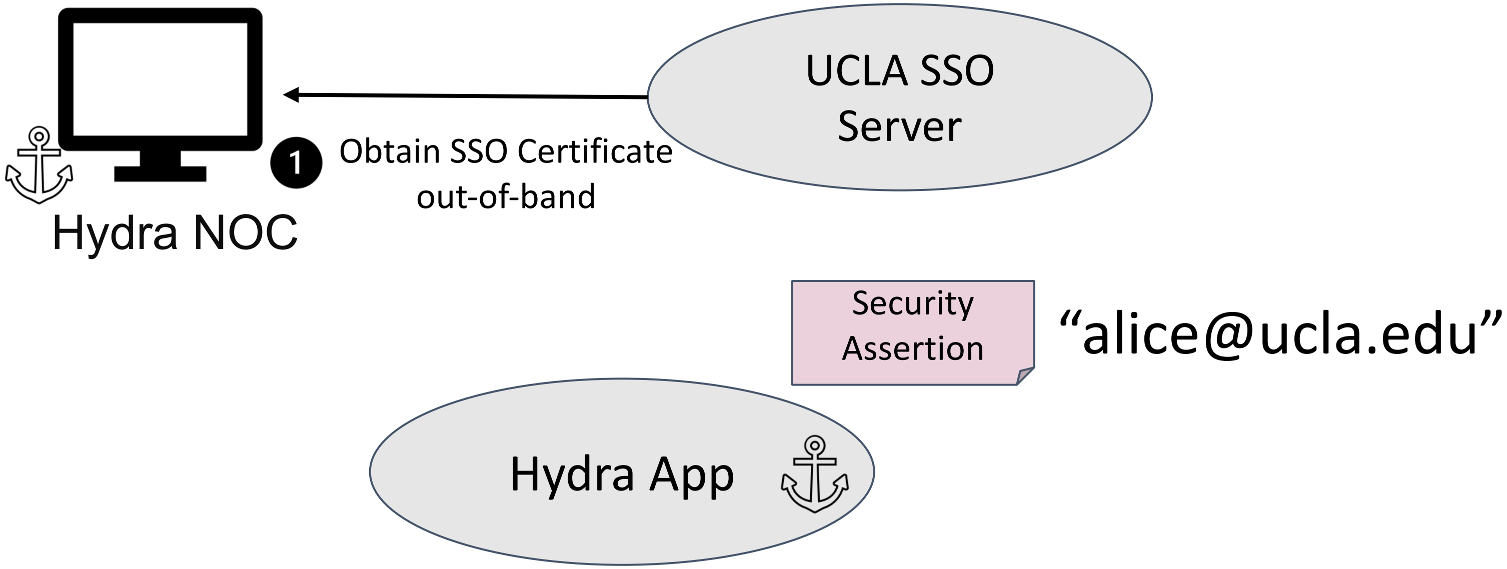
Authenticating Remote Hydra App via Campus SSO

- The user Alice who runs the remote Hydra app already has an assigned identifier
  - For example, identifier under UCLA campus *"alice@ucla.edu"*
- Alice can be authenticated by campus SSO
- Campus SSO generates a *security assertion* for the Alice by
  - cryptographically signing the identifier *"alice@ucla.edu"* with SSO certificate
- The security assertion is Alice's "existing" authentication



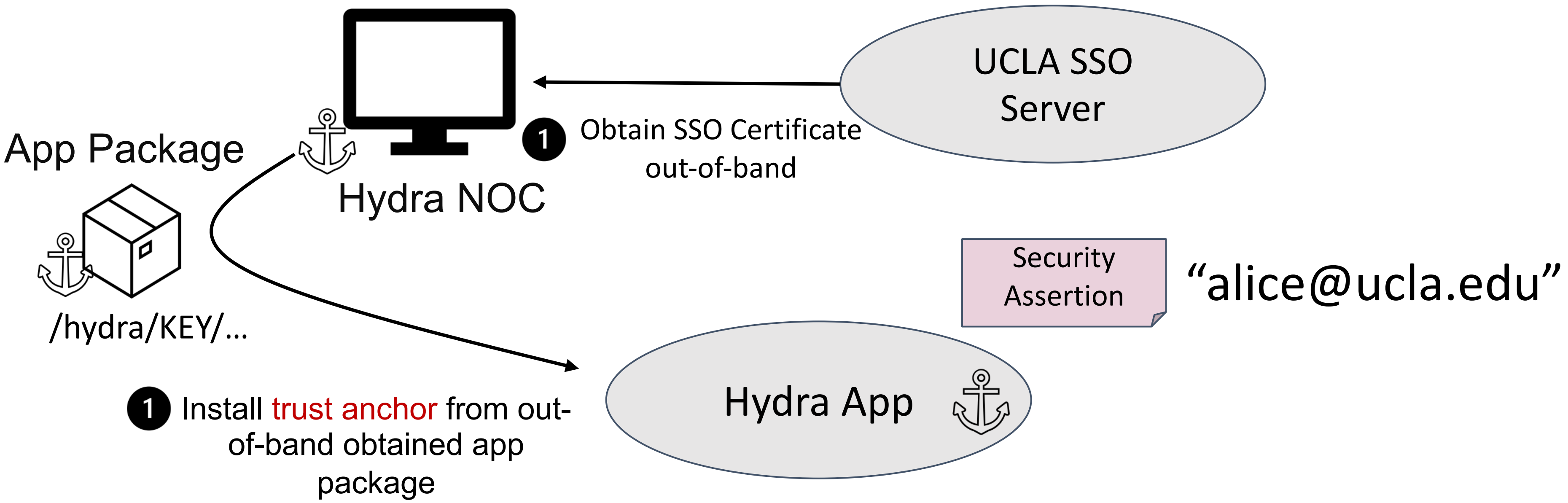
Authenticating E_{new}

- Hydra NOC obtains campus SSO certificates out-of-band
  - *e.g.*, contact campus SSO operators via emails
- Hydra NOC can authenticate all campus SSO authenticated users (e.g., Alice)
  - Thereby can authenticate Hydra app instances run by them



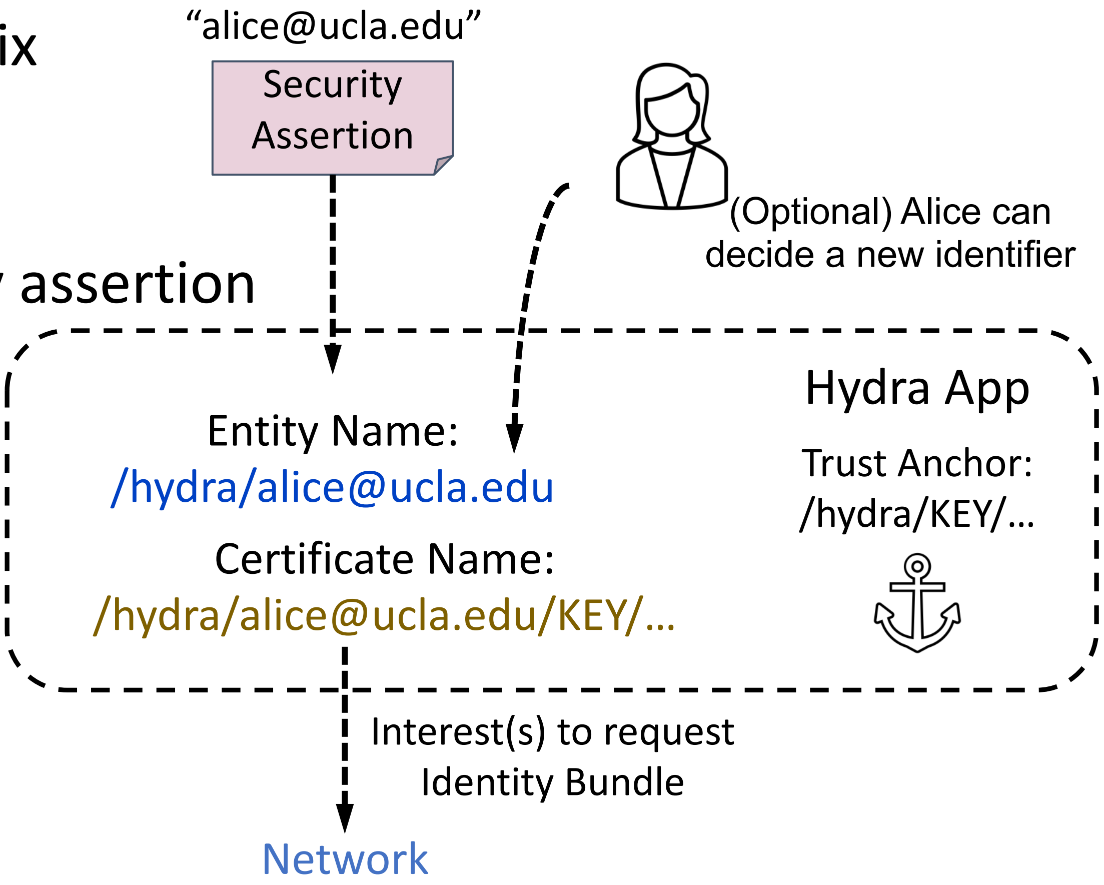
Authenticating Trust Zone Controller

- Hydra app authenticates Hydra NOC at the application installation time
- Hydra trust anchor and initial trust schema are embedded in the application package that implements the Hydra app
- Application package is authenticated out-of-band (e.g., GitHub)


UCLA SSO Server
1 Obtain SSO Certificate out-of-band
App Package
Hydra NOC
/hydra/KEY/…
Security Assertion
"alice@ucla.edu"
Hydra App
1 Install trust anchor from out-of-band obtained app package

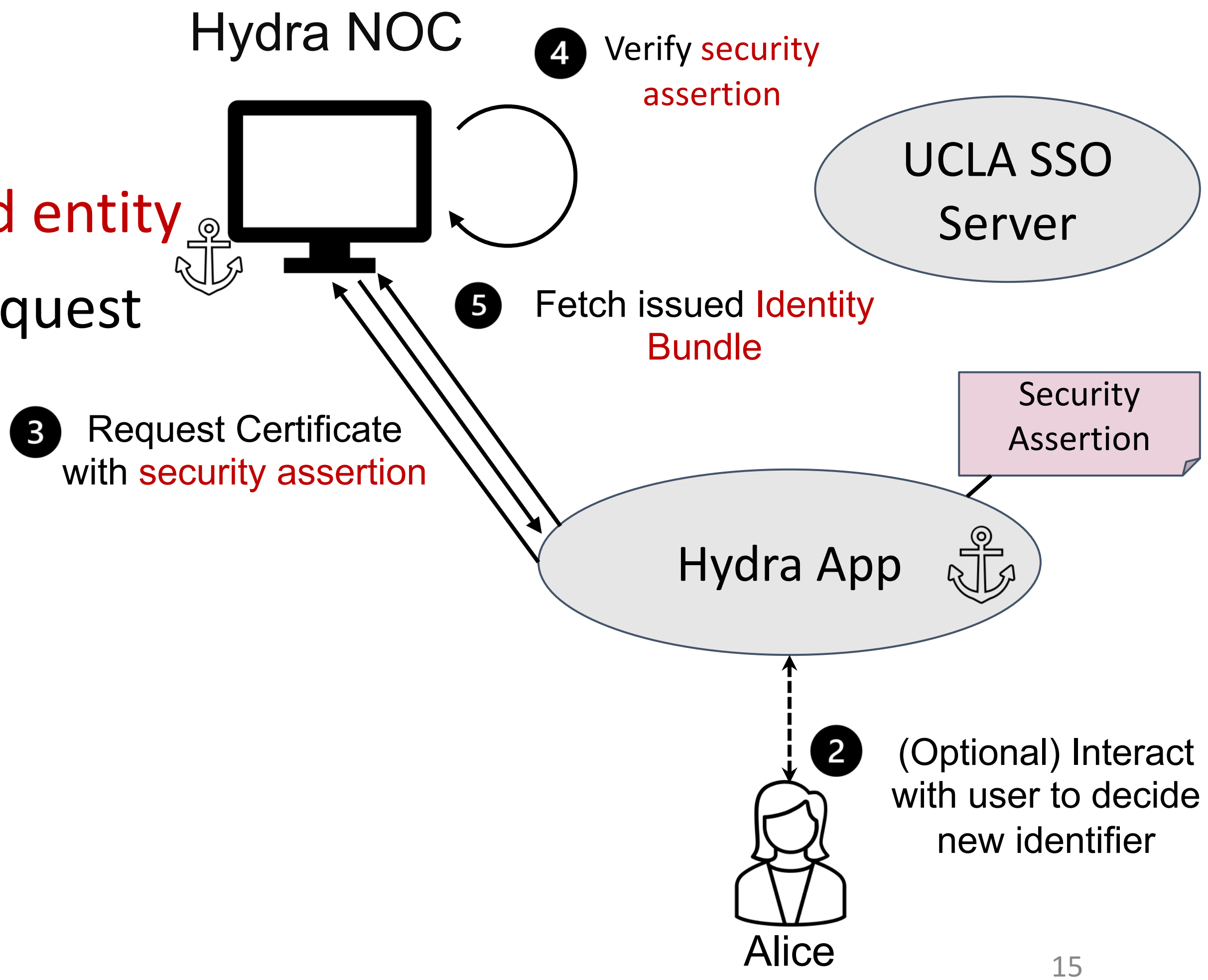
Naming Remote E_{new}

- Hydra NOC needs E_{new} name as input to generate Identity Bundle
- E_{new} name has an app prefix and unique suffix
  - Application prefix comes from trust anchor
  - Unique suffix needs assignment
- Hydra app can self-obtain name from security assertion
  - e.g., reuse the identifier "alice@ucla.edu"
  - Optionally, Alice can decide a new identifier
- Hydra app requests Identity Bundle for the newly obtained entity name



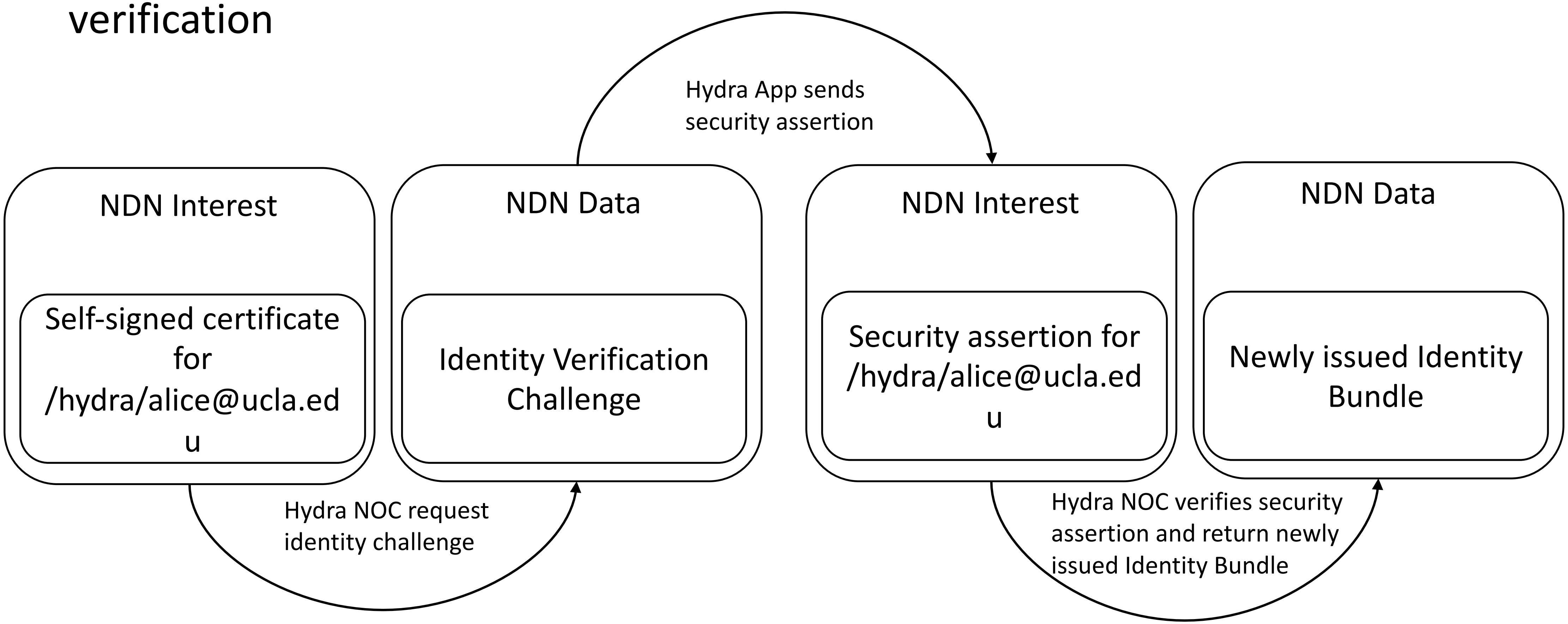
Obtaining Identity Bundle

- *Mutual authentication is achieved*
- E_{new} *Name is self-obtained*
- Identity Bundle is still needed for named entity
- Hydra app uses NDNCERT protocol to request Identity Bundle from Hydra NOC with security assertion


Hydra NOC
4 Verify security assertion
UCLA SSO Server
5 Fetch issued Identity Bundle
Security Assertion
3 Request Certificate with security assertion
Hydra App
2 (Optional) Interact with user to decide new identifier
Alice

Requesting Identity Bundle following NDNCERT

- Hydra app request Identity Bundle and provide security assertion as identity verification



Hydra App sends security assertion

NDN Interest

Self-signed certificate for /hydra/alice@ucla.edu

NDN Data

Identity Verification Challenge

NDN Interest

Security assertion for /hydra/alice@ucla.edu

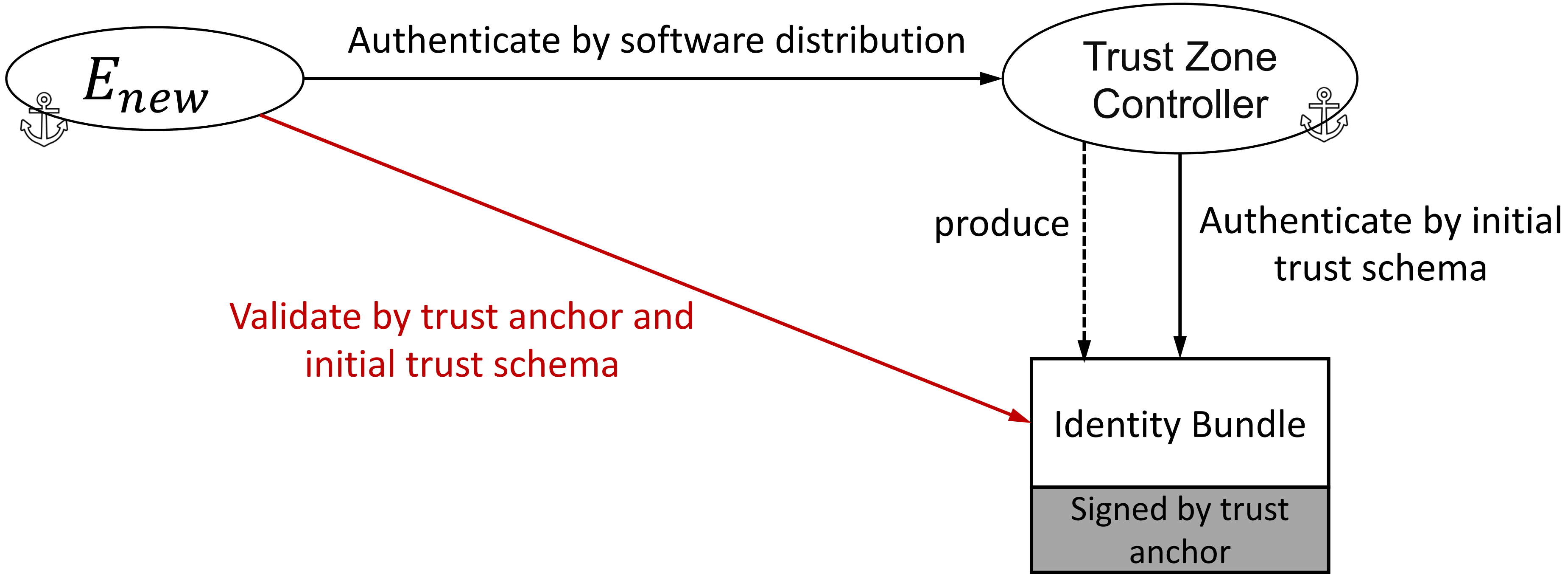
NDN Data

Newly issued Identity Bundle

Hydra NOC request identity challenge

Hydra NOC verifies security assertion and return newly issued Identity Bundle

Validating Received Identity Bundle


E_{new}
Authenticate by software distribution
Trust Zone Controller
produce
Authenticate by initial trust schema
Validate by trust anchor and initial trust schema
Identity Bundle
Signed by trust anchor

Bootstrapping NDN by Existing Trust Relations

- Before the bootstrapping can start, Trust Zone Controller and E_{new} need to authenticate each other
- Authentications are based on existing trust relations
  - Case-1: network environment is secured
    - Mutual authentication is directly achieved
  - Case-2: network environment is unsecured, E_{new} is at local
    - Physical vicinity facilitates the mutual authentication
  - Case-3: network environment is unsecured, E_{new} is only reachable via TCP/IP
    - E_{new} is authenticated by existing authentication systems
    - Trust Zone Controller is authenticated by *authenticating the software source/provider*
- Identity Bundle offers security credentials and initial trust relations after mutual authentication accomplished

Future Work: Minimize Manual Operations

- Users should have the option to manually assign an E_{new} name
- We need to offer the default option to automatically assign names
- The context of security bootstrapping may help
  - Internet hostnames (DNS names)
    - *e.g.,* “bruins.cs.ucla.edu” → “/hydra/bruins.cs.ucla.edu”
  - Information from hardware profile (for IoT cases)
    - *e.g.,* “/ndnfit/alice/locator/device-5e3f9”
- Other types of existing authentication for Hydra app
  - What if everything is certificate-based
  - InCommon now can directly issue personal certificates