TOO BAD… YOUR PASSWORD HAS JUST BEEN STOLEN!

DID YOU CONSIDER USING 2FA?

Florence Blanc-Renaud (flo@redhat.com)

Senior Software Engineer - Identity Management - Red Hat

A GOOD PASSWORD:

SECURITY THROUGH AUTHENTICATION

✓ must be easy to remember	✗ but not a dictionary word
✓ must contain different classes of characters	✗ but not easy to guess
✓ must be long enough to prevent brute-force attacks	✗ but not written on a sticky note
✓ must not be reused everywhere	✗ but I have tens of passwords to remember!
✓ must be changed regularly	✗ but is never safe against phishing traps...

MANY CONSTRAINTS BUT NO GUARANTEE THAT ONLY BOB KNOWS BOB'S PWD

PASSWORDS ARE NOT SECURE.
WHAT SHOULD I DO, THEN?

PASSWORDS ARE NOT SECURE. WHAT SHOULD I DO, THEN?

TWO FACTOR AUTHENTICATION

- **ONE-TIME PASSWORDS (OTP):**
  - Usual password
  - Code retrieved from an external device (smartphone, hardware token)
- **SMART CARDS:**
  - PIN
  - Smart Card (credit-card format + reader or USB token)

IDENTITY MANAGEMENT

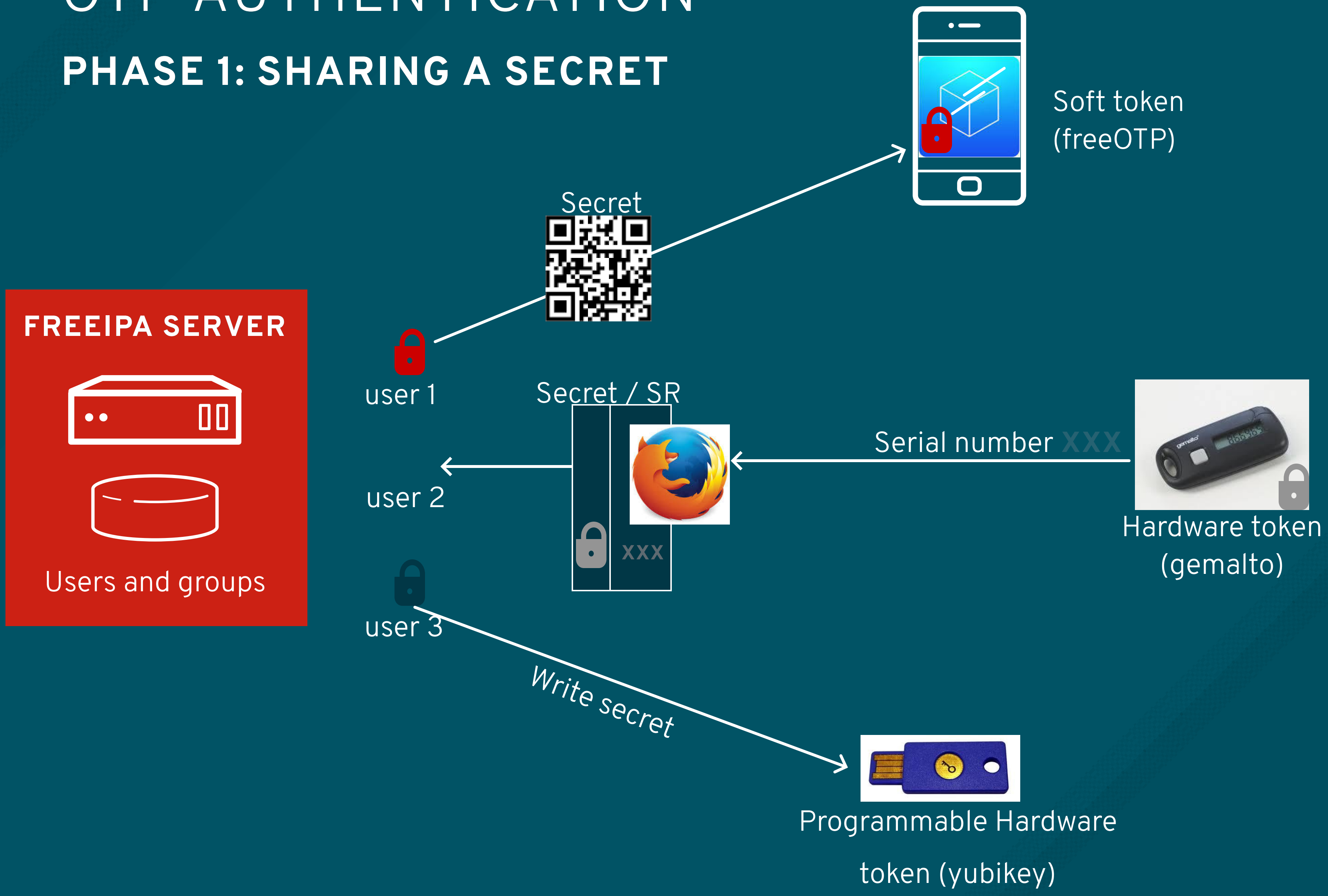
MAIN FEATURES

- **CENTRALIZED AUTHENTICATION**
  - Source: IDM or Active Directory
  - Credentials: passwords, certificates, Smart Cards, OTP tokens
  - Single Sign-On: Kerberos, SAML, OpenID
- **CENTRALIZED AUTHORIZATION**
  - Resources: systems, services, applications
  - HBAC, sudo rules, privileges
- **CENTRALIZED MANAGEMENT**
  - Policy
  - Certificates and Keys
- **DNS**

BASED ON A COLLECTION OF OPEN SOURCE COMPONENTS:
KDC, LDAP, PKI, DNS, FREEIPA

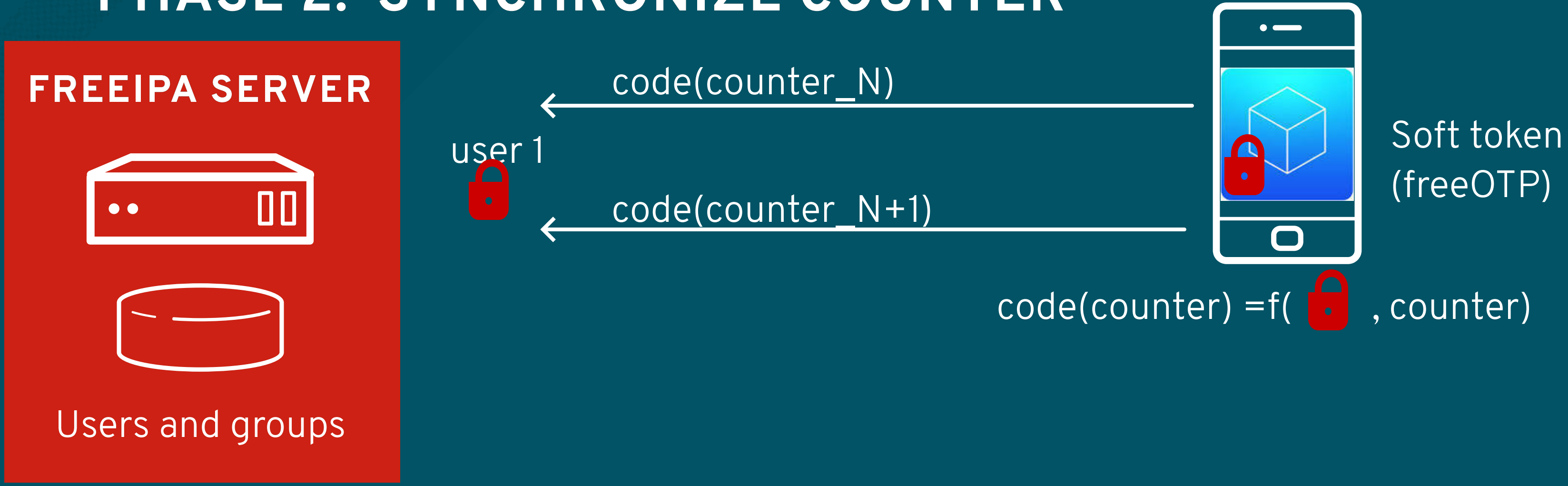
OTP AUTHENTICATION

PHASE 1: SHARING A SECRET



OTP AUTHENTICATION

PHASE 2: SYNCHRONIZE COUNTER



FREEIPA SERVER

Users and groups

code(counter_N)

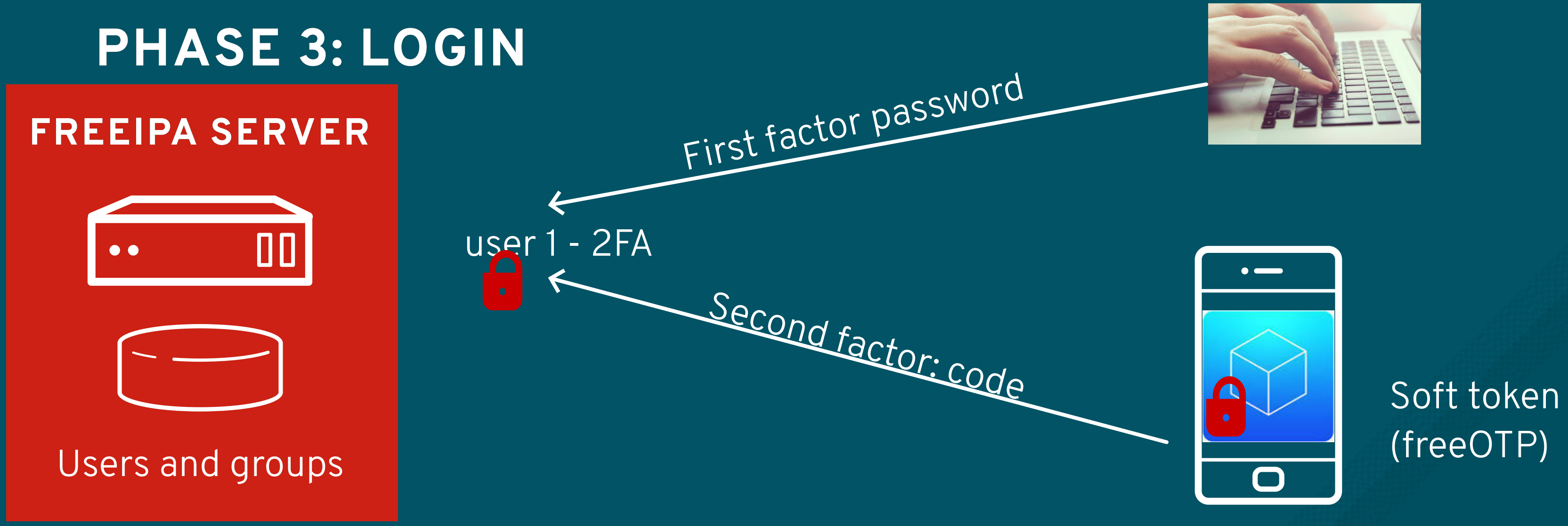
user 1

code(counter_N+1)

Soft token (freeOTP)

code(counter) =f(🔒 , counter)

PHASE 3: LOGIN



FREEIPA SERVER

Users and groups

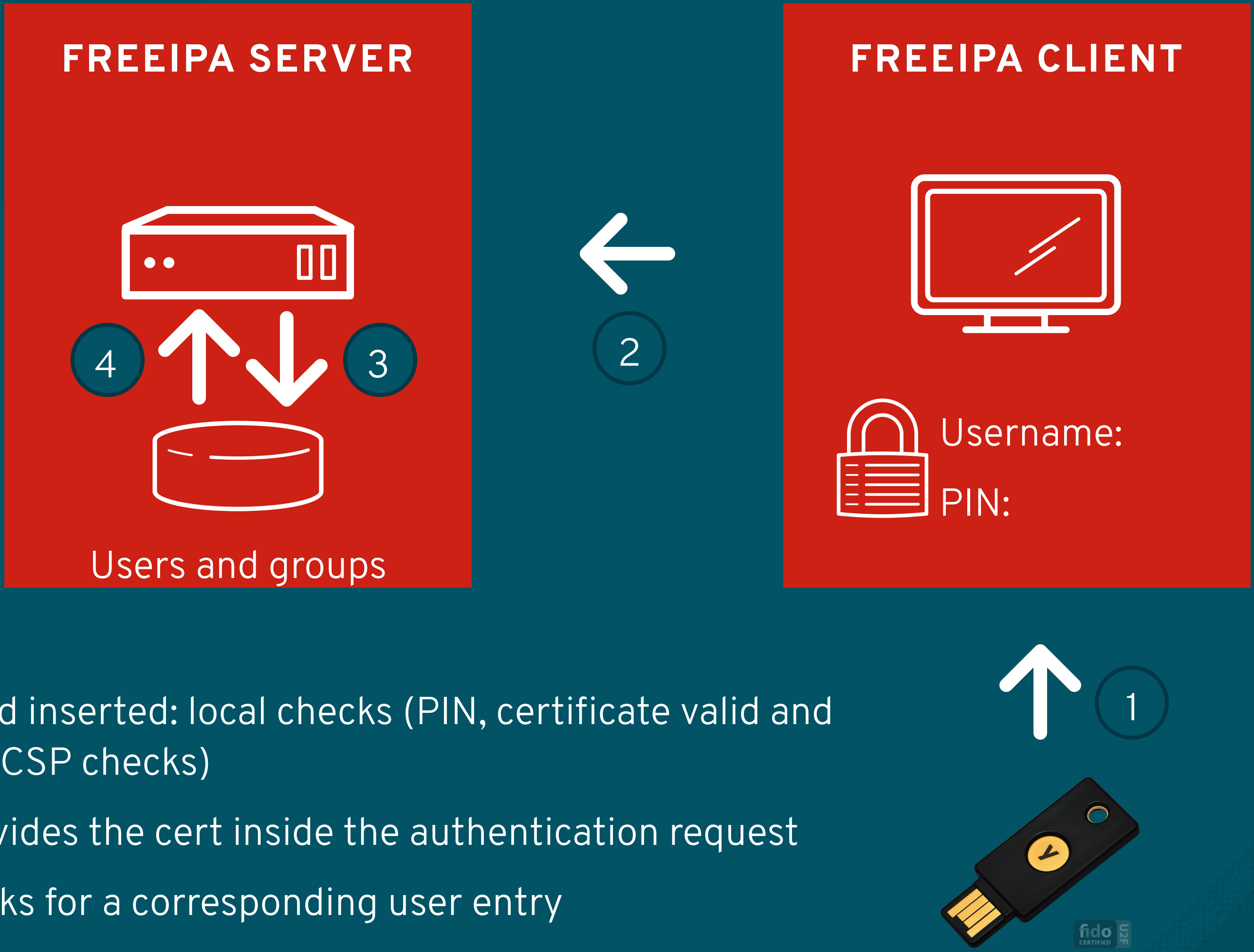
First factor password

user 1 - 2FA

Second factor: code

Soft token (freeOTP)

SMART CARD AUTHENTICATION



1. Smart card inserted: local checks (PIN, certificate valid and trusted, OCSP checks)
2. client provides the cert inside the authentication request
3. server looks for a corresponding user entry
4. user found: user is authenticated and gets a TGT

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)

FREEIPA SERVER

Users and groups

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)

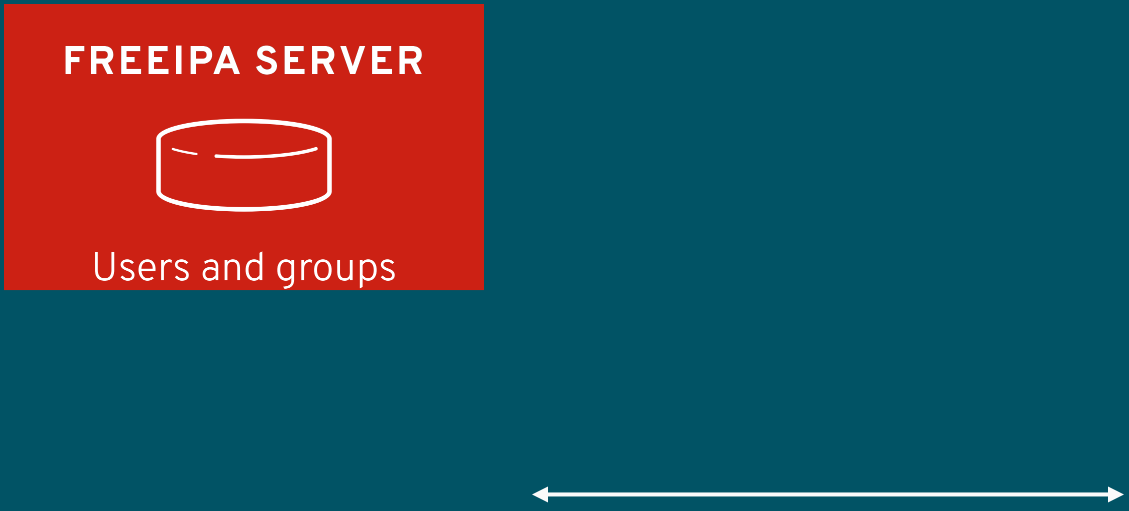
FREEIPA SERVER

Users and groups

```
-----BEGIN CERTIFICATE-----
MIIC3jCCAcagAwIBAgICfhUw
[...]
RTuD/xbee2FwuOpTLsfUo0ef
-----END CERTIFICATE-----
```

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)



FREEIPA SERVER

Users and groups

```
-----BEGIN CERTIFICATE-----
MIIC3jCCAcagAwIBAgICfhUw
[...]
RTuD/xbee2FwuOpTLsfUo0ef
-----END CERTIFICATE-----
```

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)

FREEIPA SERVER

Users and groups

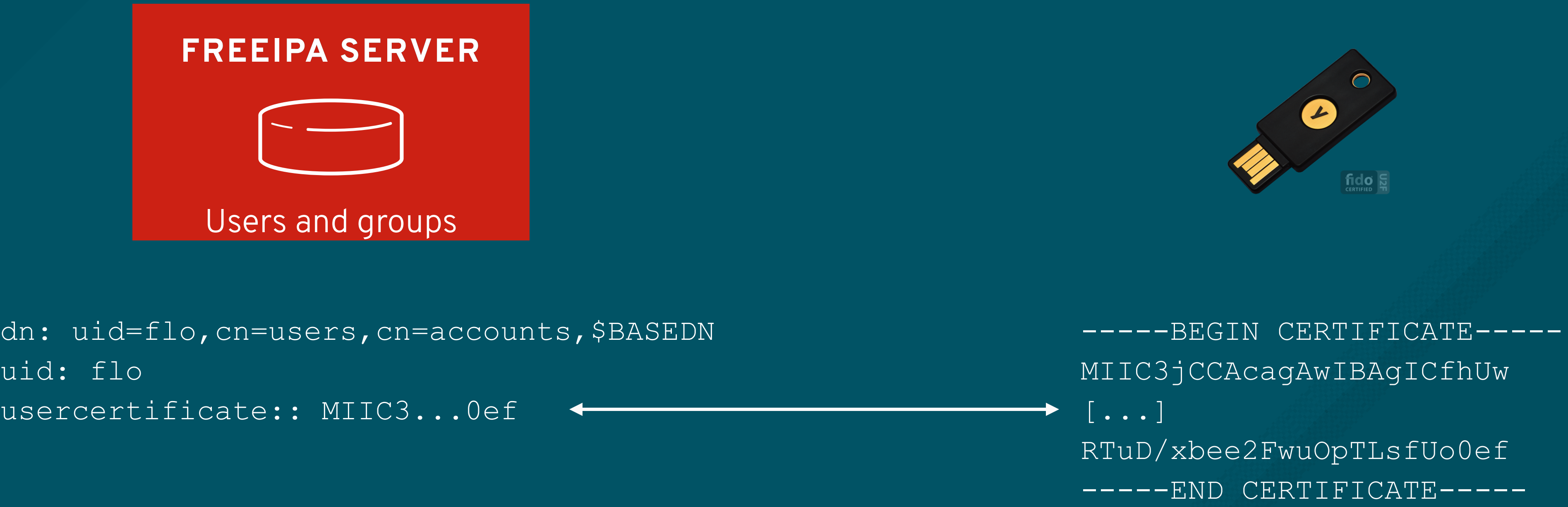
```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
usercertificate:: MIIC3...0ef
```

```
-----BEGIN CERTIFICATE-----
MIIC3jCCAcagAwIBAgICfhUw
[...]
RTuD/xbee2FwuOpTLsfUo0ef
-----END CERTIFICATE-----
```

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)

- Admin has to extract the certificate from the Smart Card in order to provision the user entry



FREEIPA SERVER

Users and groups

```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
usercertificate:: MIIC3...0ef
```

```
-----BEGIN CERTIFICATE-----
MIIC3jCCAcagAwIBAgICfhUw
[...]
RTuD/xbee2FwuOpTLsfUo0ef
-----END CERTIFICATE-----
```

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)

- Admin has to extract the certificate from the Smart Card in order to provision the user entry
- Painful process when smart card is lost: issue new certificate, update LDAP entry

FREEIPA SERVER

Users and groups

```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
usercertificate:: MIIC3...0ef
```

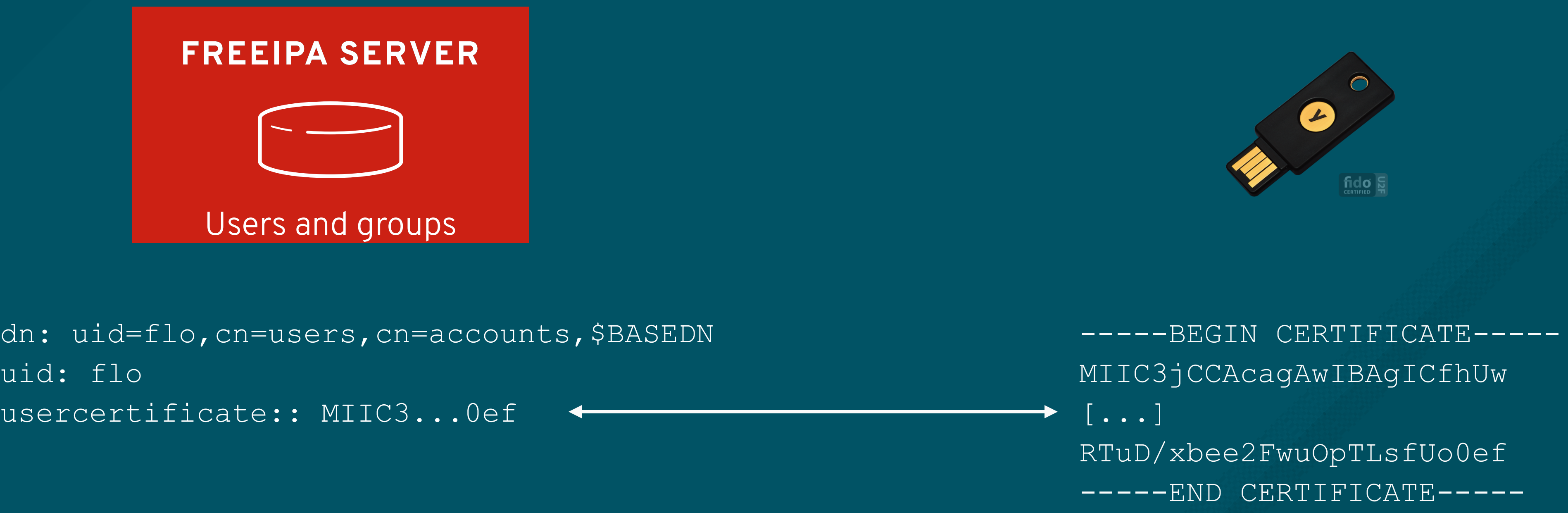
↔

```
-----BEGIN CERTIFICATE-----
MIIC3jCCAcagAwIBAgICfhUw
[...]
RTuD/xbee2FwuOpTLsfUo0ef
-----END CERTIFICATE-----
```

MAPPING A CERTIFICATE AND A USER

WITH FULL CERTIFICATE BLOB (RHEL < 7.4 / FREEIPA < 4.5)

- Admin has to extract the certificate from the Smart Card in order to provision the user entry
- Painful process when smart card is lost: issue new certificate, update LDAP entry
- A certificate can be mapped to only one user entry



FREEIPA SERVER

Users and groups

```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
usercertificate:: MIIC3...0ef
```

```
-----BEGIN CERTIFICATE-----
MIIC3jCCAcagAwIBAgICfhUw
[...]
RTuD/xbee2FwuOpTLsfUo0ef
-----END CERTIFICATE-----
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

FREEIPA SERVER

Users and groups

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

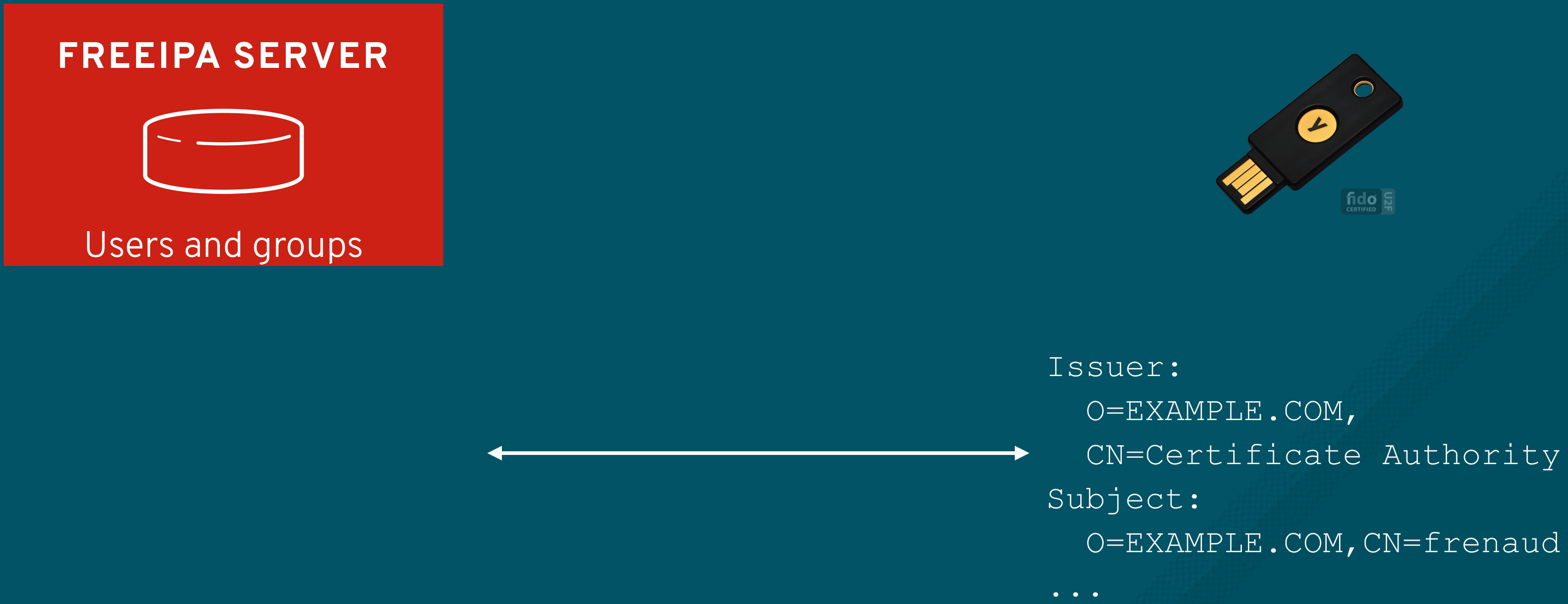
FREEIPA SERVER

Users and groups

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)



FREEIPA SERVER

Users and groups

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

FREEIPA SERVER

Users and groups

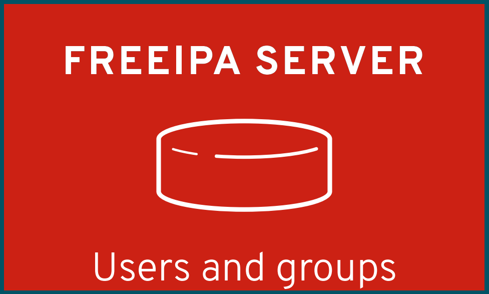
```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
ipaCertMapData:
  X509:<I>O=EXAMPLE.COM,CN=Certificate
  Authority<S>O=EXAMPLE.COM,CN=frenaud
```

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

- Admin can provision the user *before* the certificate is created



FREEIPA SERVER

Users and groups

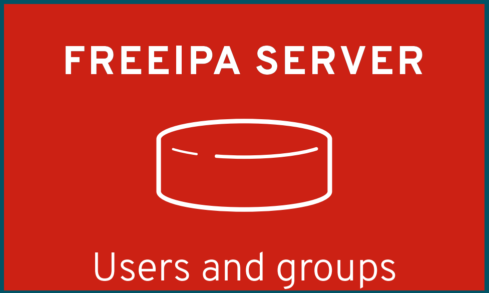
```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
ipaCertMapData:
  X509:<I>O=EXAMPLE.COM,CN=Certificate
  Authority<S>O=EXAMPLE.COM,CN=frenaud
```

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

- Admin can provision the user *before* the certificate is created
- No need to publish the certificate in the user entry


FREEIPA SERVER

Users and groups

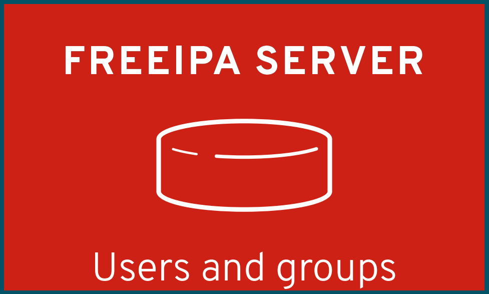
```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
ipaCertMapData:
  X509:<I>O=EXAMPLE.COM,CN=Certificate
  Authority<S>O=EXAMPLE.COM,CN=frenaud
```

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

- Admin can provision the user *before* the certificate is created
- No need to publish the certificate in the user entry
- Possible to map multiple users with a single certificate (user roles)



```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
ipaCertMapData:
  X509:<I>O=EXAMPLE.COM,CN=Certificate
  Authority<S>O=EXAMPLE.COM,CN=frenaud
```

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

MAPPING A CERTIFICATE AND A USER

WITH FLEXIBLE MAPPING (RHEL 7.4 / FREEIPA 4.5)

- Admin can provision the user *before* the certificate is created
- No need to publish the certificate in the user entry
- Possible to map multiple users with a single certificate (user roles)
- Pkinit support: Kerberos certificate-based authentication

FREEIPA SERVER

Users and groups

```
dn: uid=flo,cn=users,cn=accounts,$BASEDN
uid: flo
ipaCertMapData:
  X509:<I>O=EXAMPLE.COM,CN=Certificate
  Authority<S>O=EXAMPLE.COM,CN=frenaud
```

```
Issuer:
  O=EXAMPLE.COM,
  CN=Certificate Authority
Subject:
  O=EXAMPLE.COM,CN=frenaud
...
```

QUESTIONS?

RESOURCES

FREEIPA

- Project wiki: http://www.freeipa.org
- Project repo and issues: https://pagure.io/freeipa/
- Blog aggregation: http://planet.freeipa.org/
- FreeIPA demo instance in the cloud: http://www.freeipa.org/page/Demo
- Mailing lists:
  - freeipa-users@lists.fedorahosted.org
  - freeipa-devel@lists.fedorahosted.org
  - freeipa-users@lists.fedorahosted.org

THANK YOU!