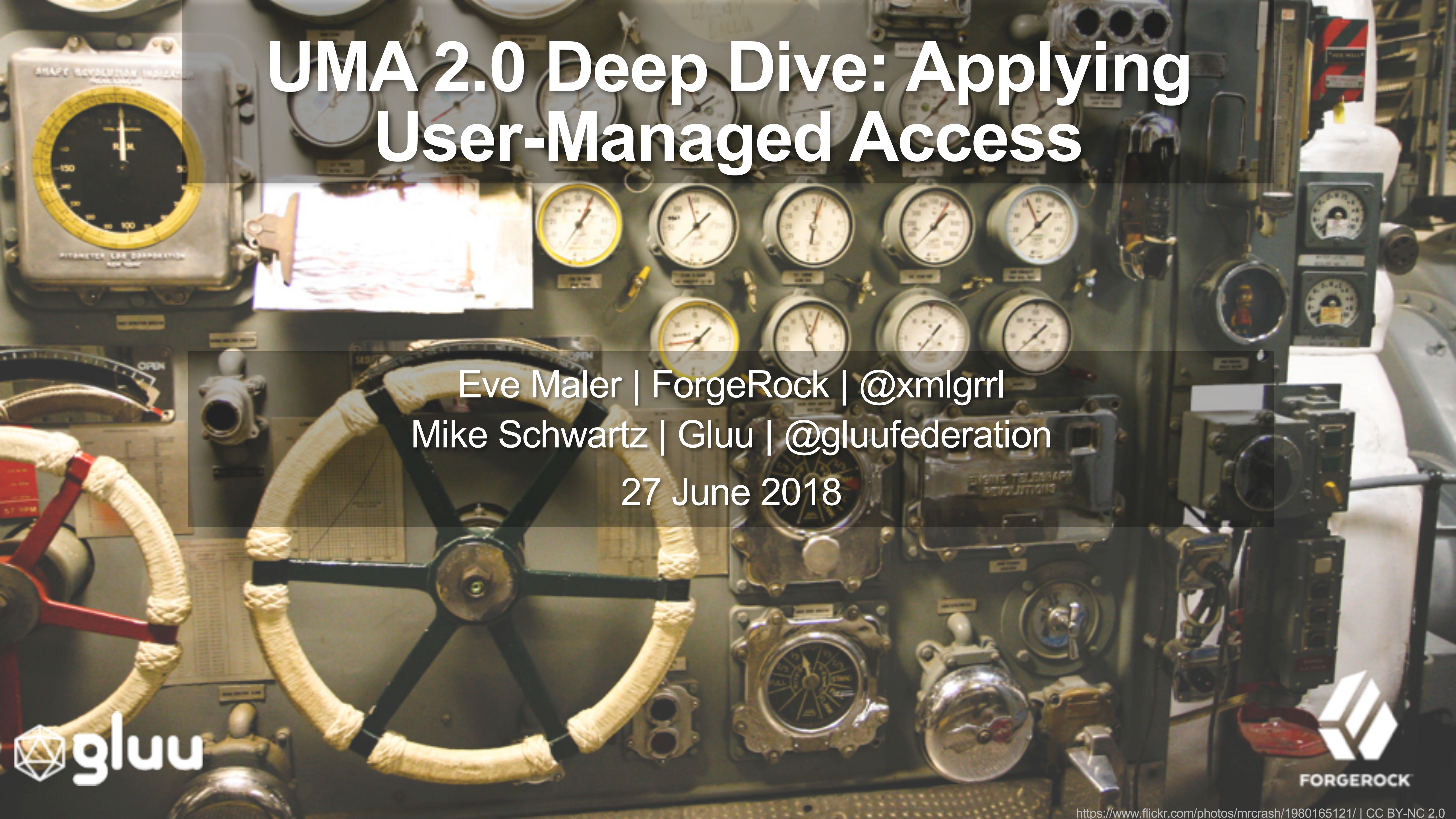
UMA 2.0 Deep Dive: Applying User-Managed Access

Eve Maler | ForgeRock | @xmlgrrl

Mike Schwartz | Gluu | @gluufederation

27 June 2018

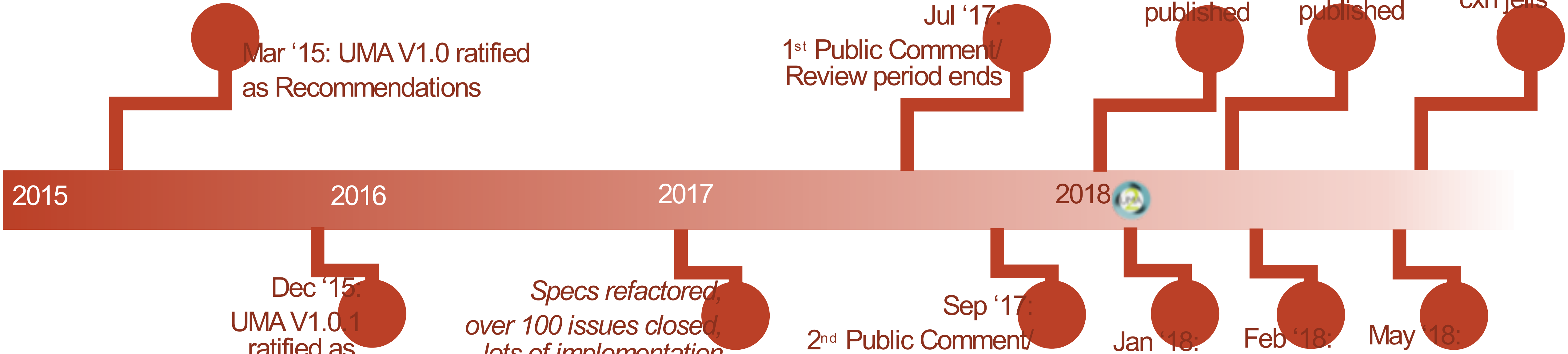


https://www.flickr.com/photos/mrcrash/1980165121/ | CC BY-NC 2.0

Lots to cover so let’s jump in

- A User-Managed Access timeline
- UMA architecture in the OAuth and OpenID Connect context
- UMA use cases
- UMA flows
- Demonstration focusing on an enterprise use case and “interactive claims gathering”
- Walkthrough focusing on a consumer health IoT use case and “pushed claims”
- Q&A

The UMA standard's progress



2015

2016

2017

2018

Mar '15: UMA V1.0 ratified as Recommendations

Dec '15: UMA V1.0.1 ratified as Recommendations

Specs refactored, over 100 issues closed, lots of implementation input received, Disposition of Comments doc written…

Jul '17: 1st Public Comment/ Review period ends

Sep '17: 2nd Public Comment/ Review period ends

Jan '18: Final Recommendations published

Jan '18: UMA2 logo (designed by @domcat)

Jan '18: Draft UMA Business Model Report published

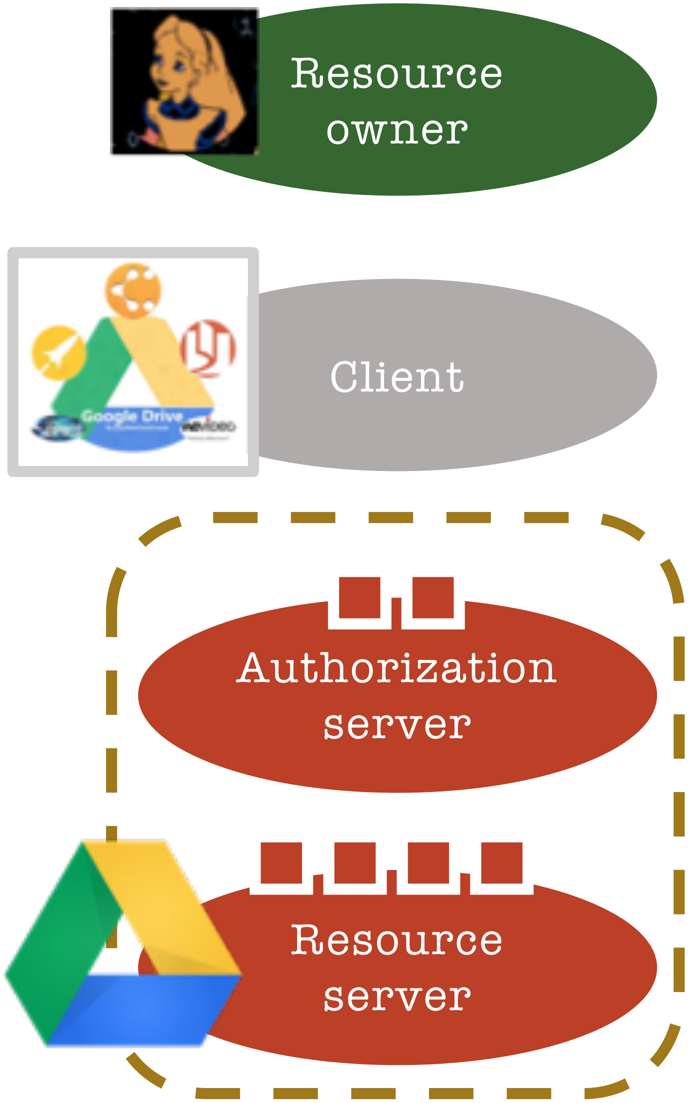
Feb '18: Charter update

May '18: Vendors supporting UMA2: Gluu, ForgeRock, Keycloak (WSO2 coming Q3)

Jun '18: Business model/ IRM cxn jells

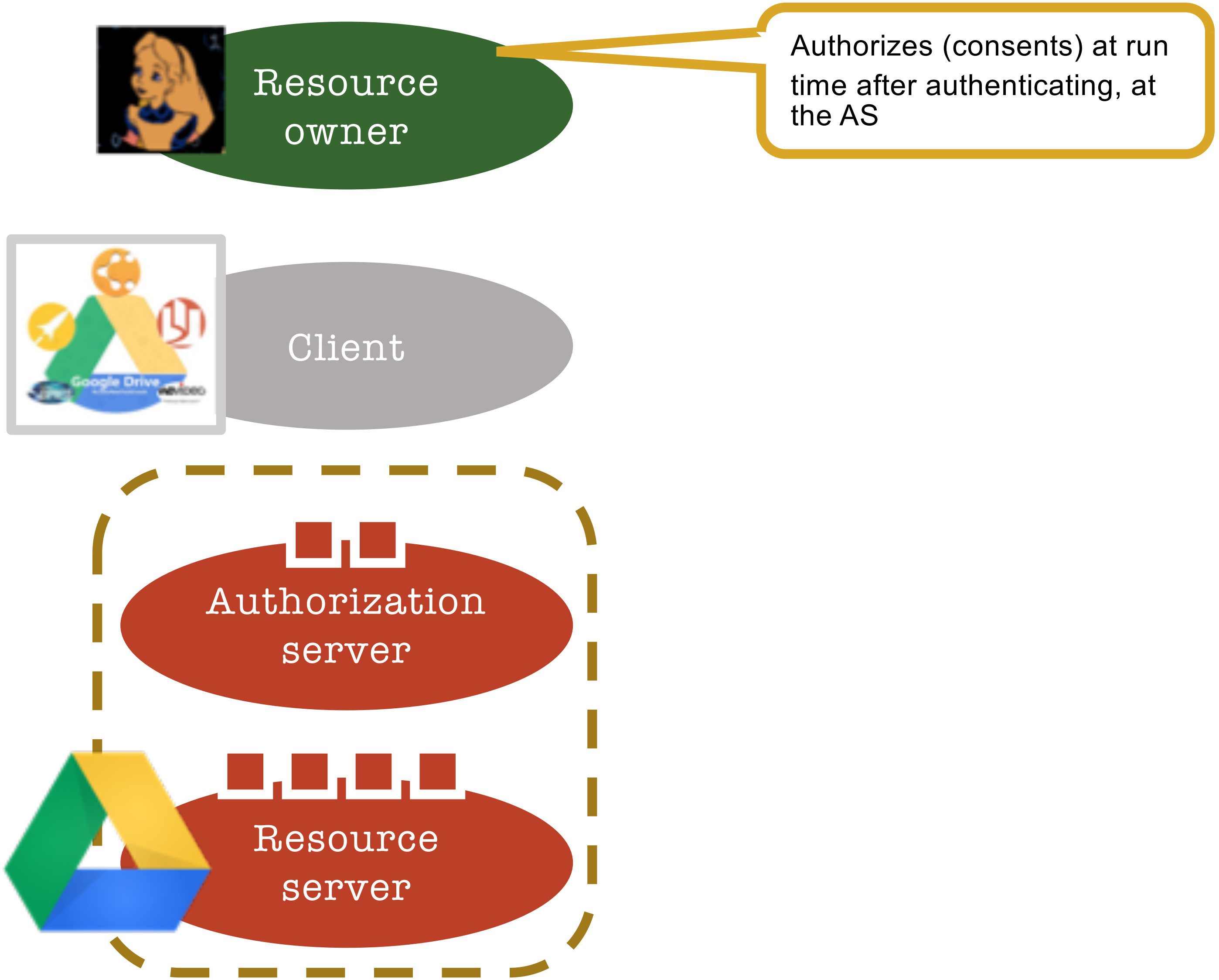
OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”



OAuth is for constrained delegation to apps

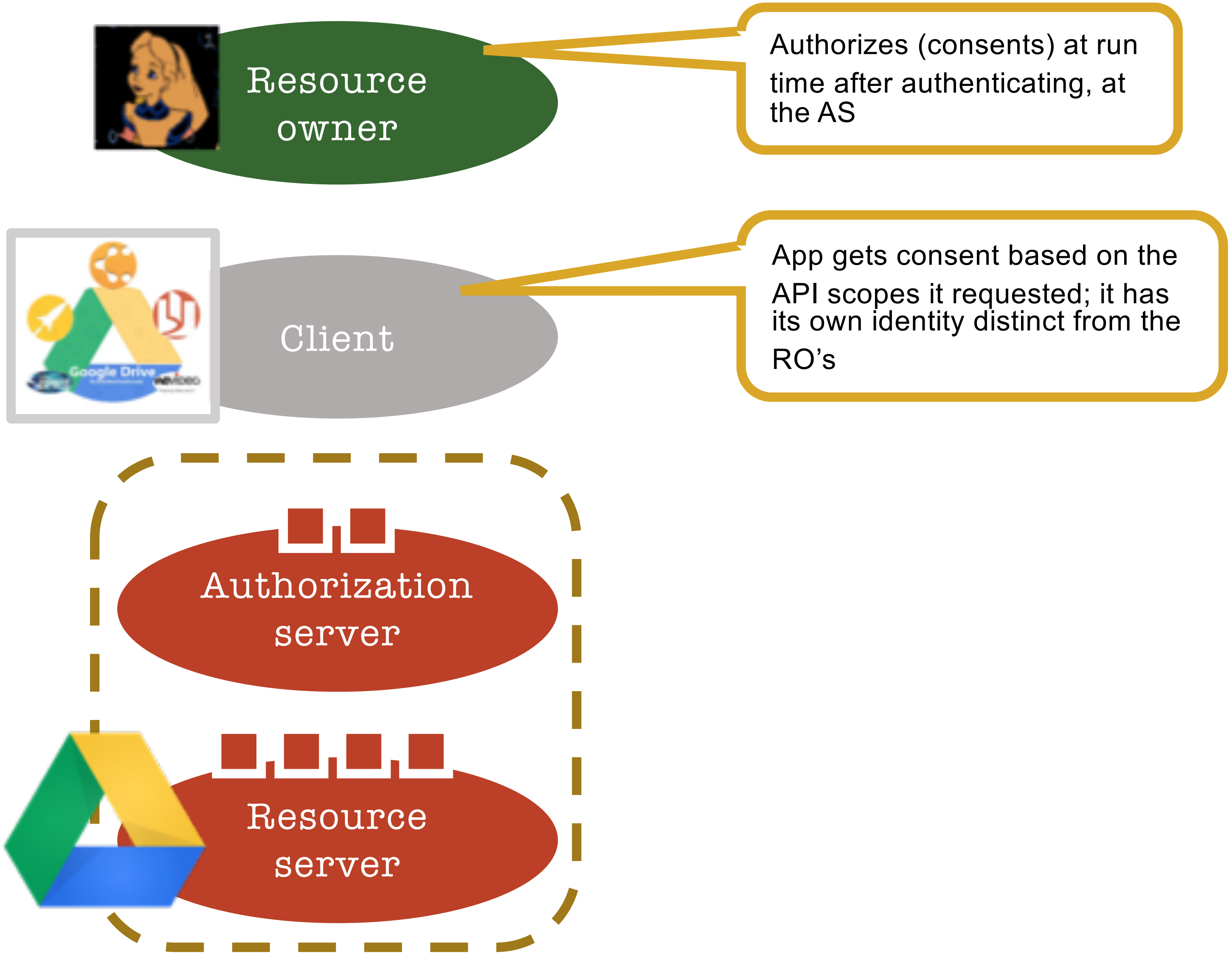
It has helped to kill the “password anti-pattern”



Authorizes (consents) at run time after authenticating, at the AS

OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”

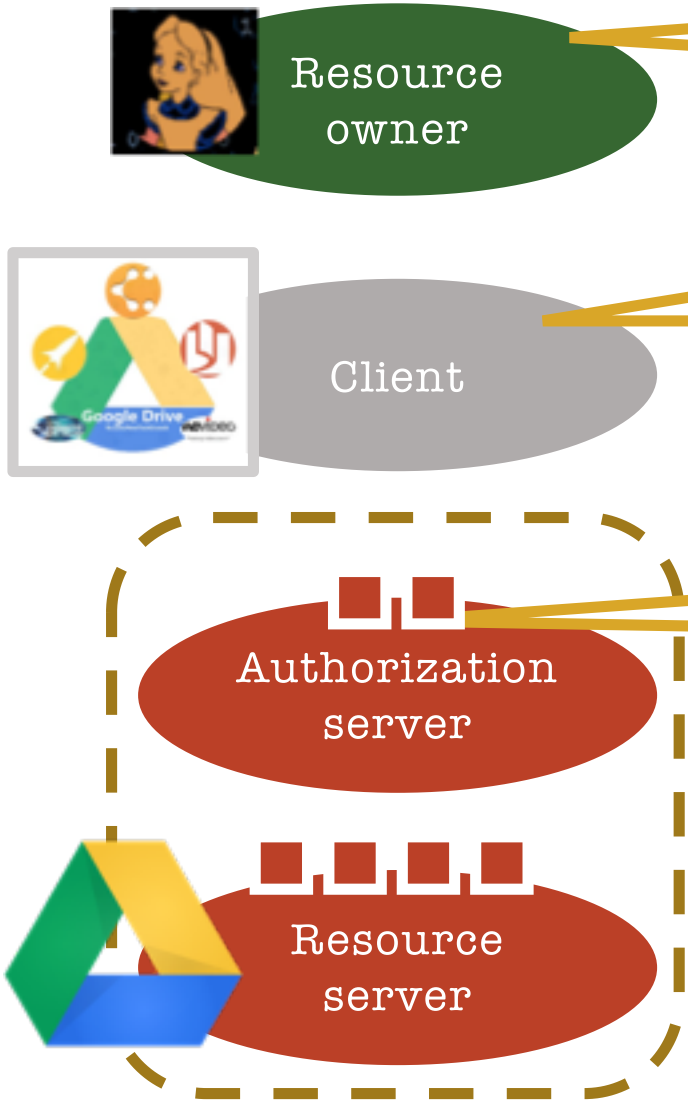


Authorizes (consents) at run time after authenticating, at the AS

App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”



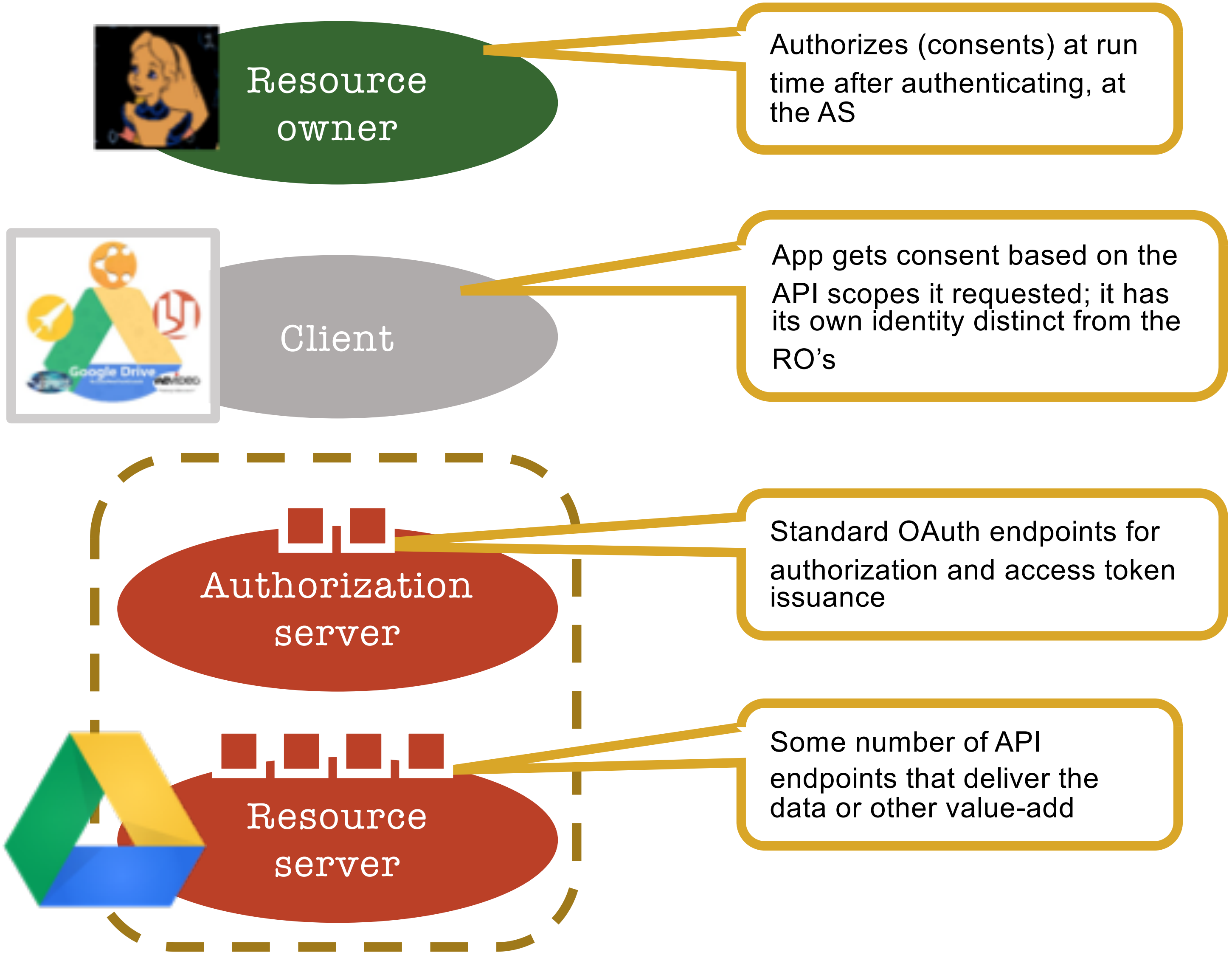
Authorizes (consents) at run time after authenticating, at the AS

App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

Standard OAuth endpoints for authorization and access token issuance

OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”



Authorizes (consents) at run time after authenticating, at the AS

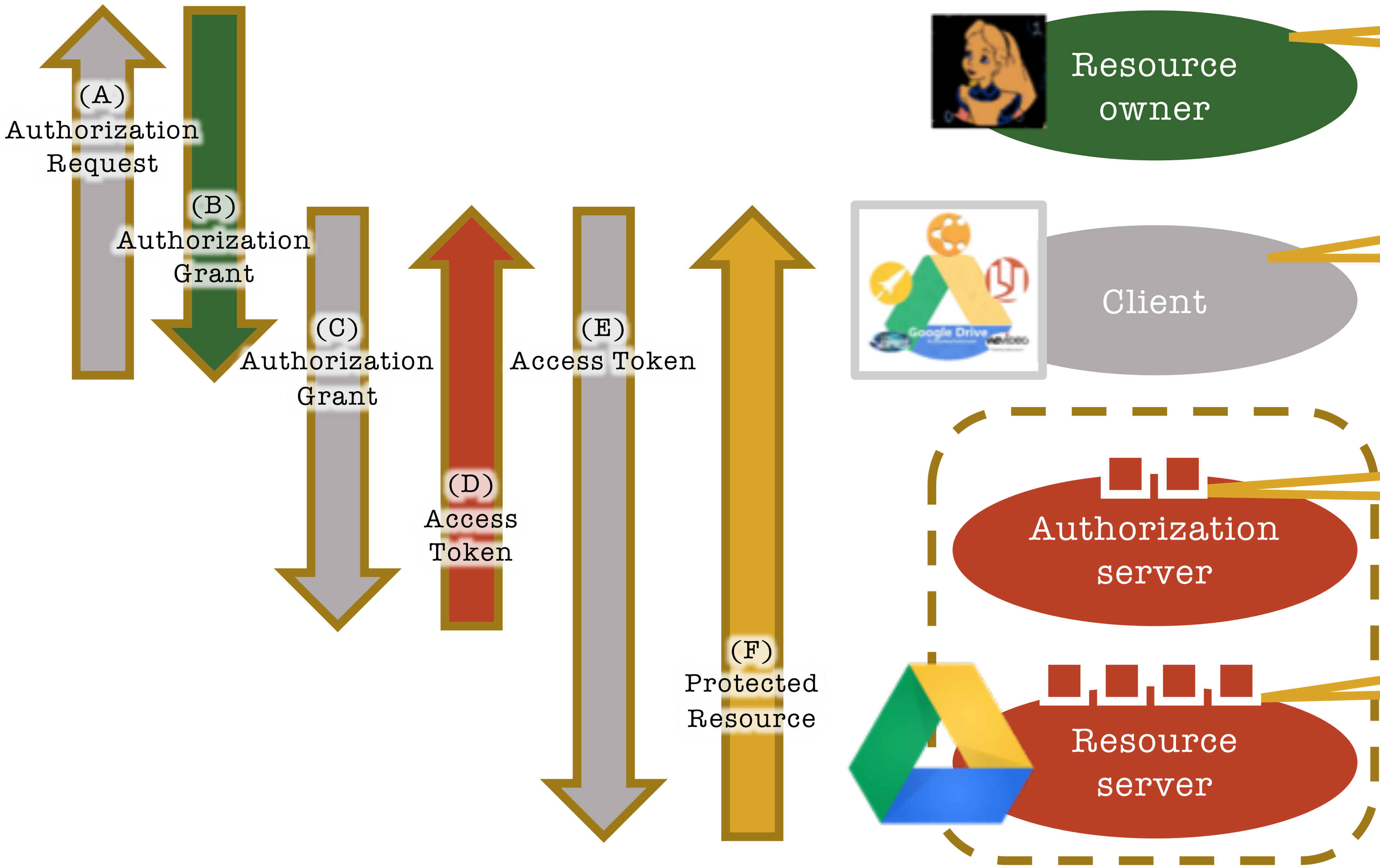
App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

Standard OAuth endpoints for authorization and access token issuance

Some number of API endpoints that deliver the data or other value-add

OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”



Authorizes (consents) at run time after authenticating, at the AS

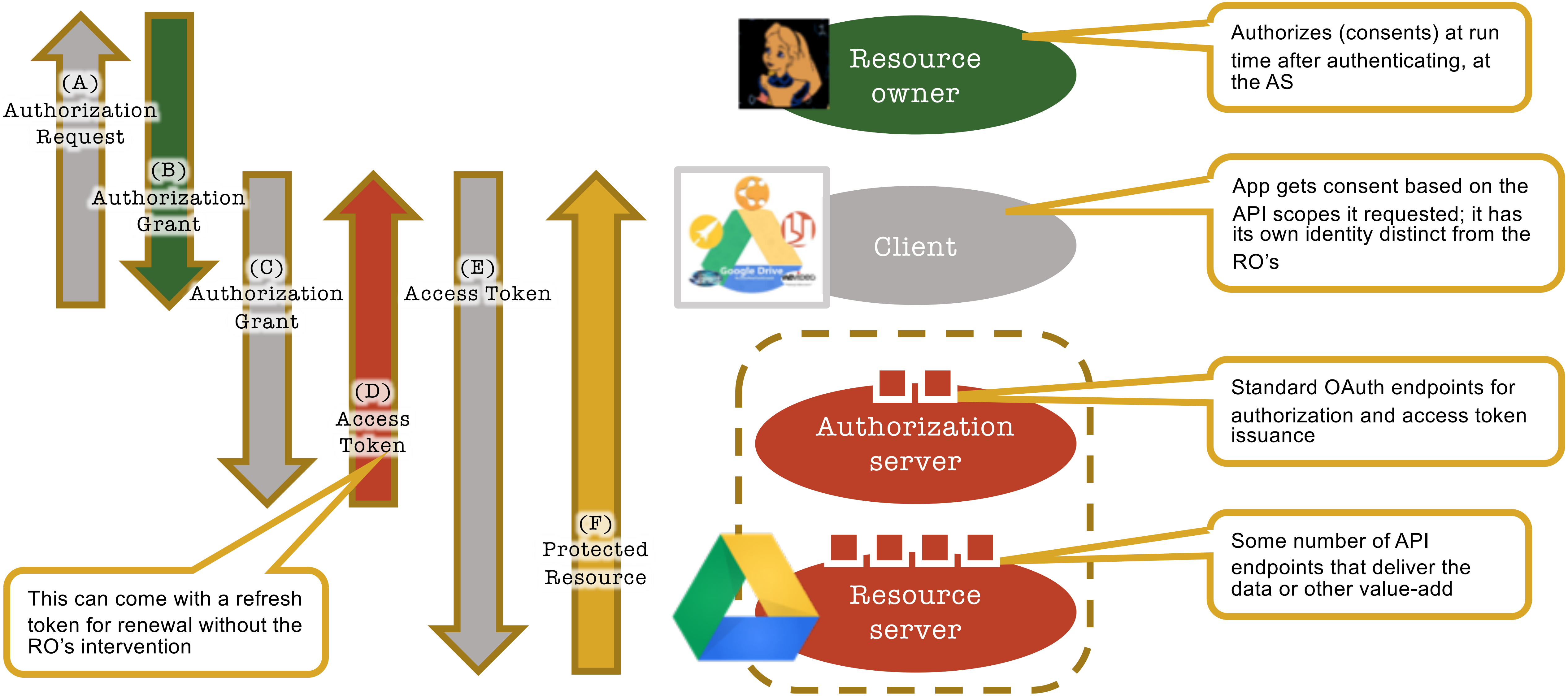
App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

Standard OAuth endpoints for authorization and access token issuance

Some number of API endpoints that deliver the data or other value-add

OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”



Authorizes (consents) at run time after authenticating, at the AS

App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

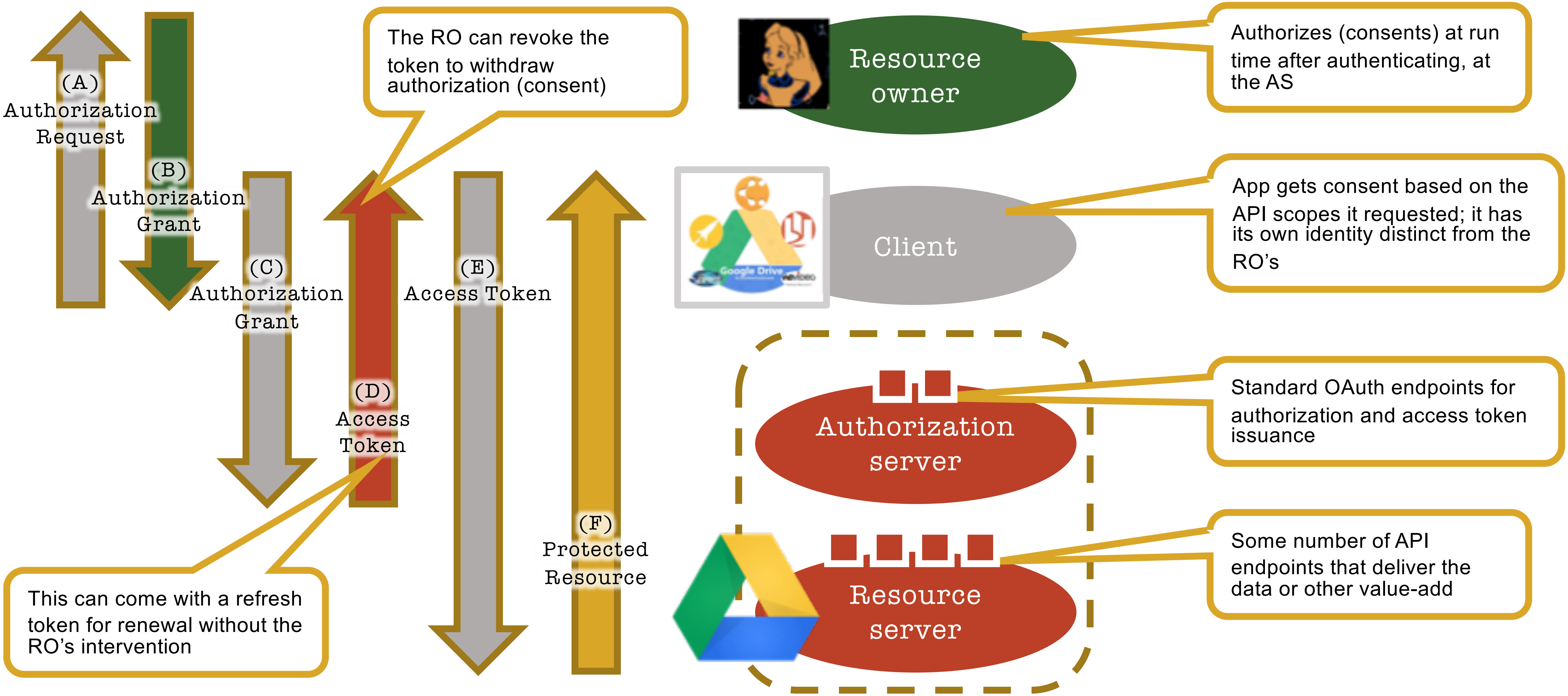
Standard OAuth endpoints for authorization and access token issuance

Some number of API endpoints that deliver the data or other value-add

This can come with a refresh token for renewal without the RO’s intervention

OAuth is for constrained delegation to apps

It has helped to kill the “password anti-pattern”



The RO can revoke the token to withdraw authorization (consent)

This can come with a refresh token for renewal without the RO’s intervention

Authorizes (consents) at run time after authenticating, at the AS

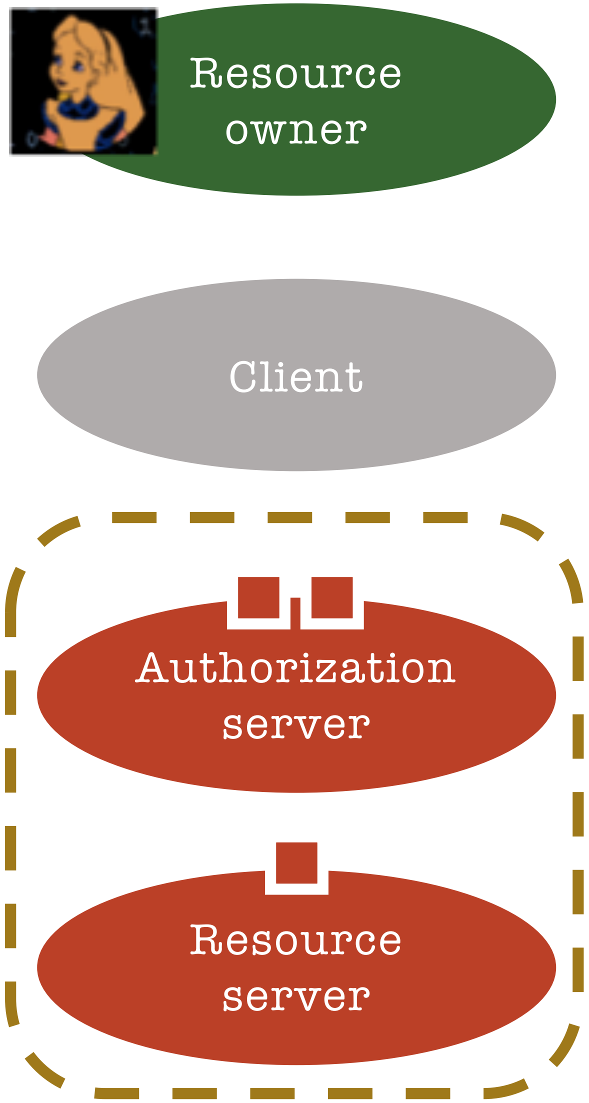
App gets consent based on the API scopes it requested; it has its own identity distinct from the RO’s

Standard OAuth endpoints for authorization and access token issuance

Some number of API endpoints that deliver the data or other value-add

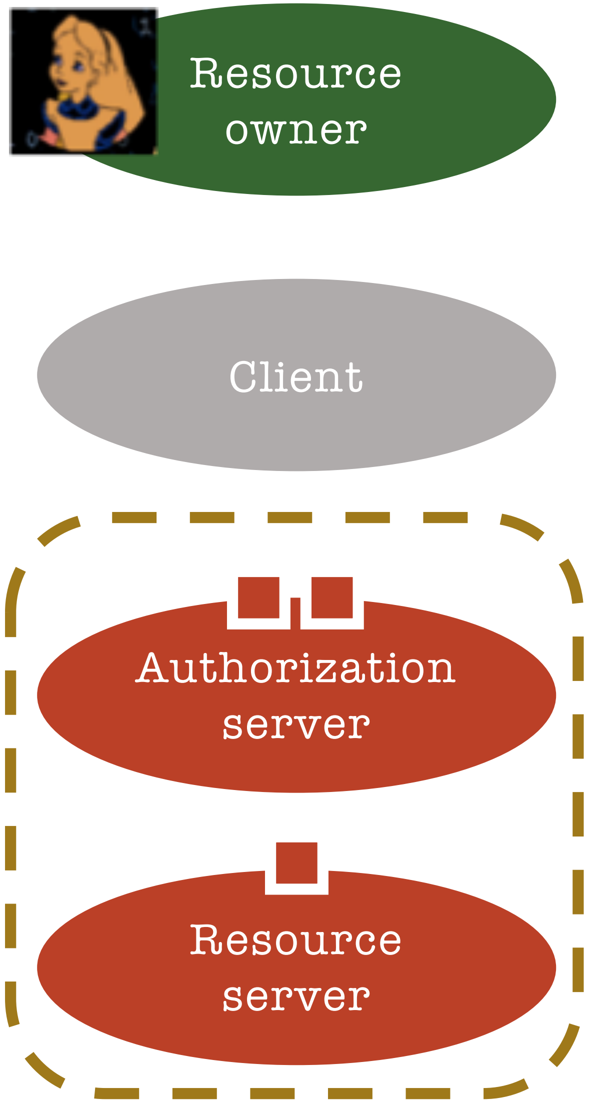
OpenID Connect does modern-day federation

It is an OAuth-protected identity API, plus a bit more



OpenID Connect does modern-day federation

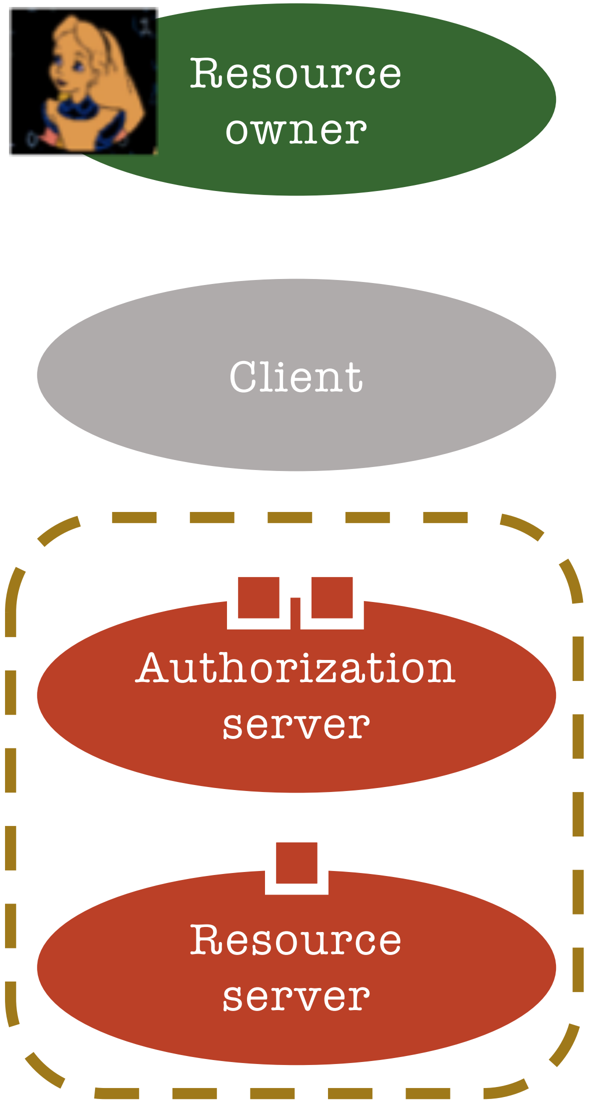
It is an OAuth-protected identity API, plus a bit more



= Federation user

OpenID Connect does modern-day federation

It is an OAuth-protected identity API, plus a bit more



Resource owner = Federation user

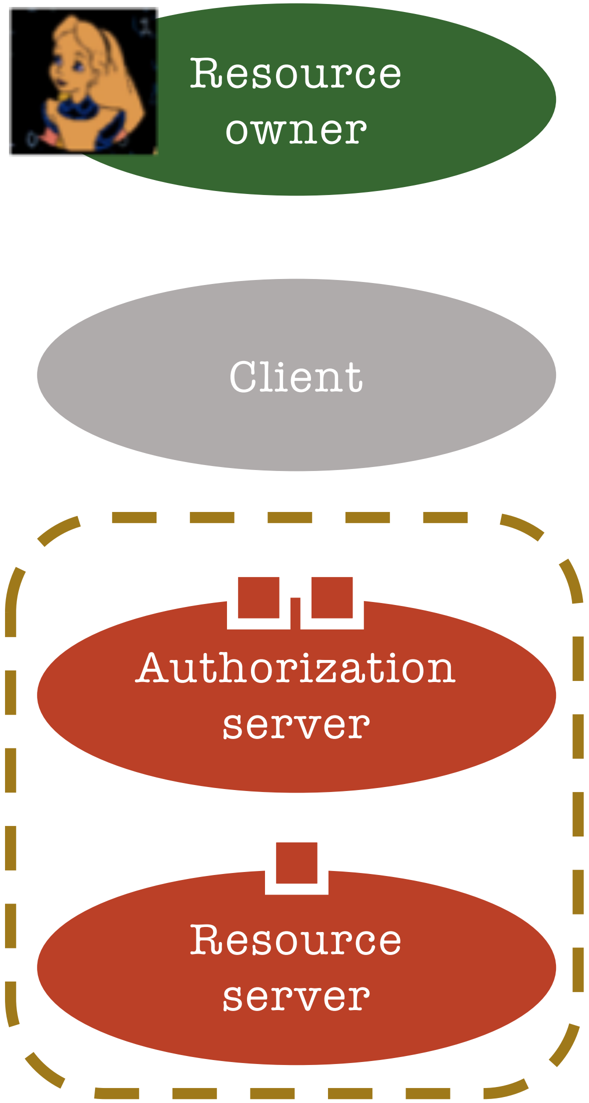
Client = Relying party

Authorization server

Resource server

OpenID Connect does modern-day federation

It is an OAuth-protected identity API, plus a bit more



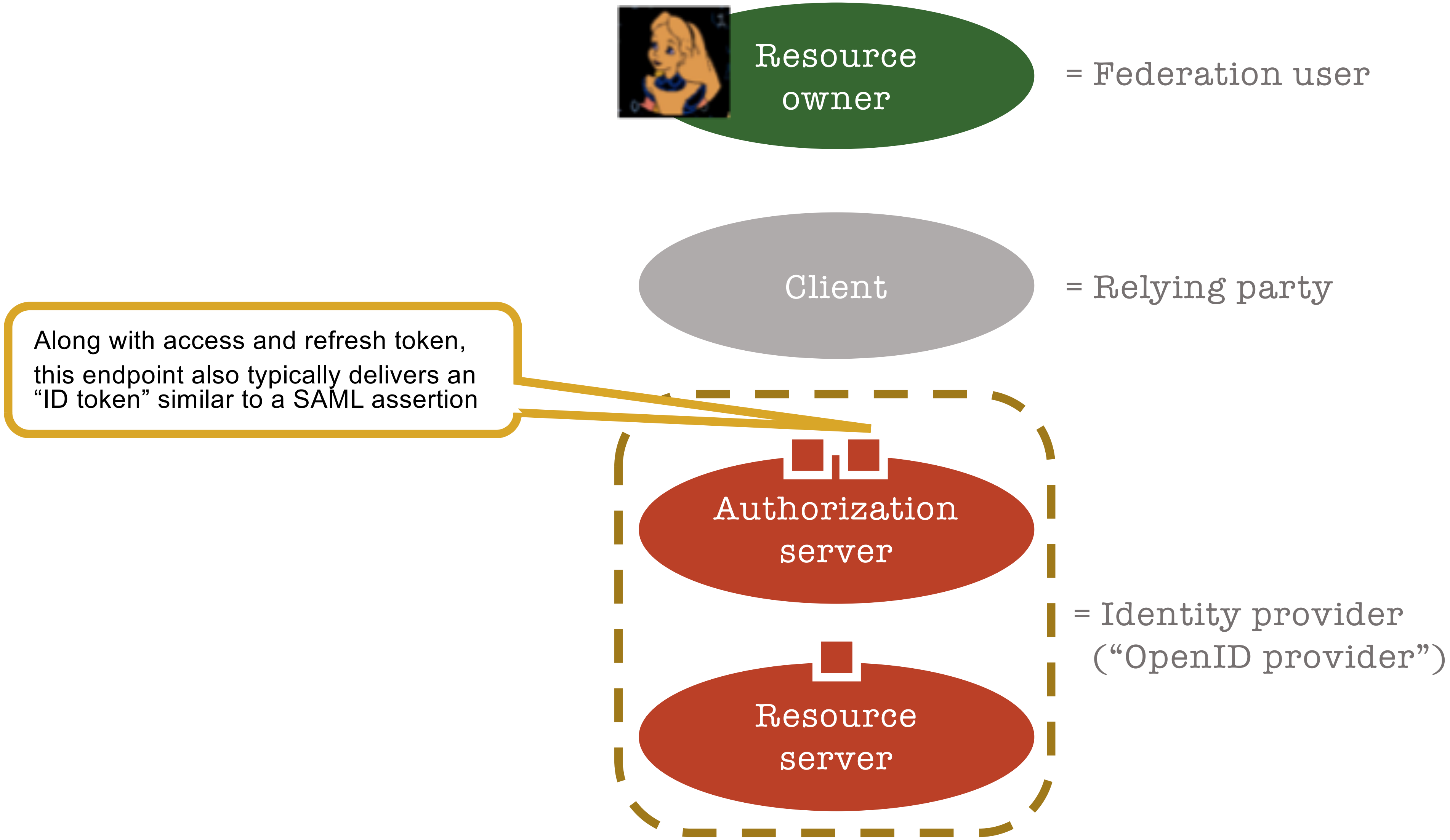
= Federation user

= Relying party

= Identity provider
(“OpenID provider”)

OpenID Connect does modern-day federation

It is an OAuth-protected identity API, plus a bit more



Resource owner = Federation user

Client = Relying party

Authorization server

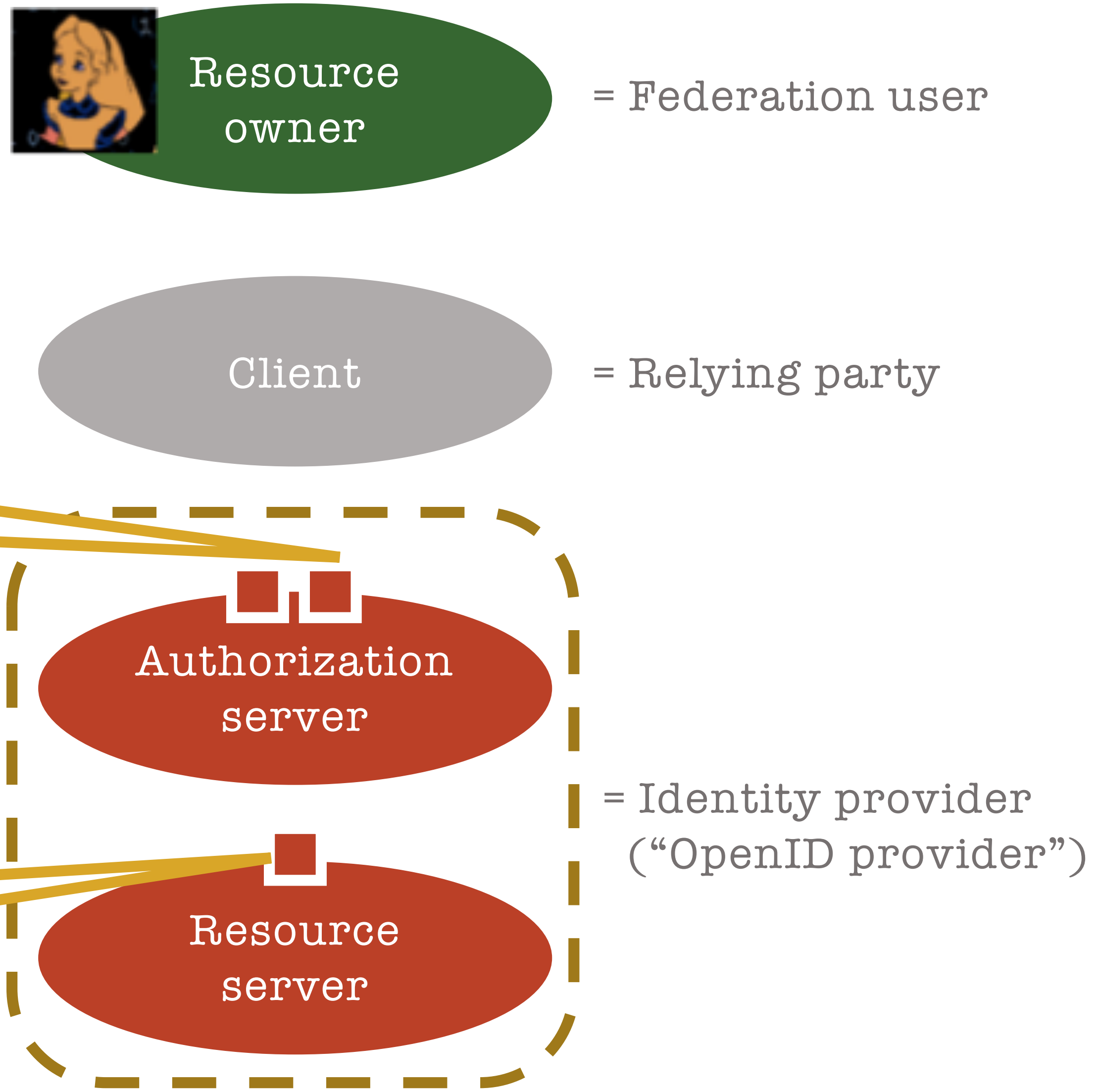
Resource server

= Identity provider (“OpenID provider”)

Along with access and refresh token, this endpoint also typically delivers an “ID token” similar to a SAML assertion

OpenID Connect does modern-day federation

It is an OAuth-protected identity API, plus a bit more

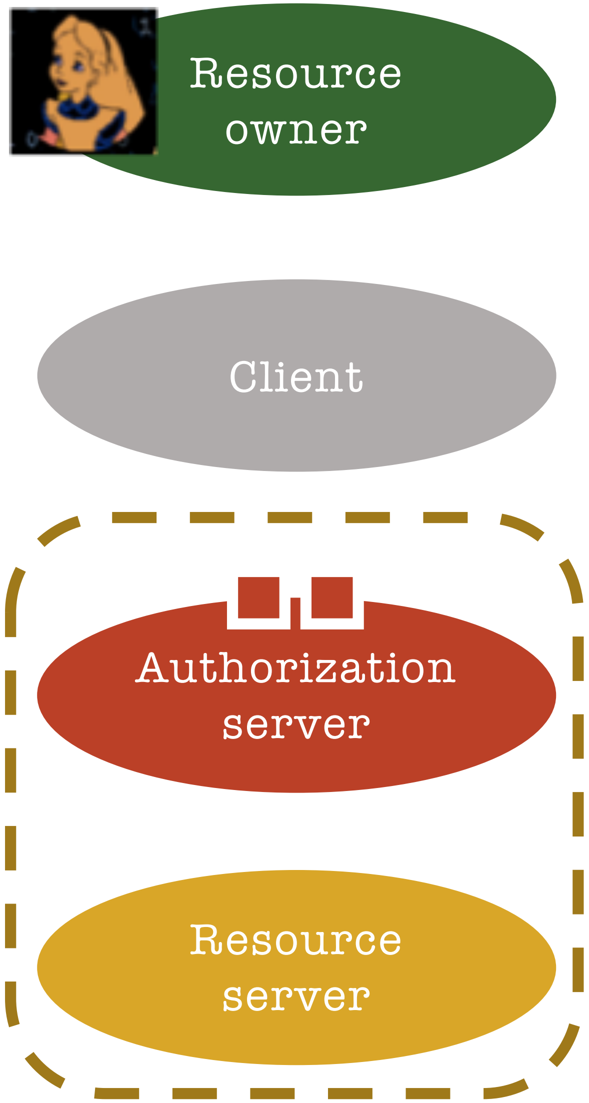


Along with access and refresh token, this endpoint also typically delivers an “ID token” similar to a SAML assertion

Standard UserInfo endpoint can be called with an access token to look up identity claims

User-Managed Access is for cross-party sharing

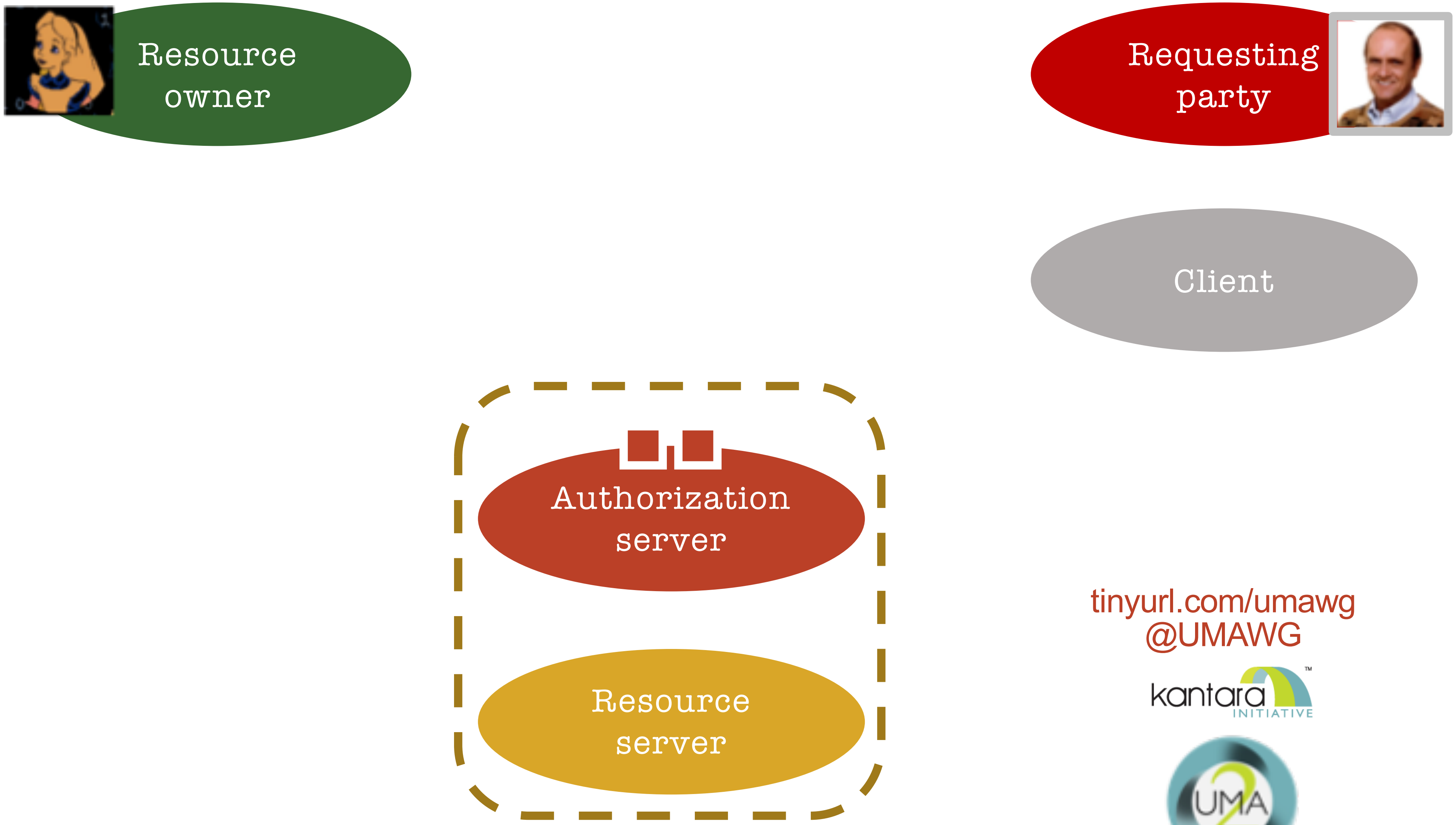
UMA brings next-gen delegation and consent to OAuth



tinyurl.com/umawg
@UMAWG

User-Managed Access is for cross-party sharing

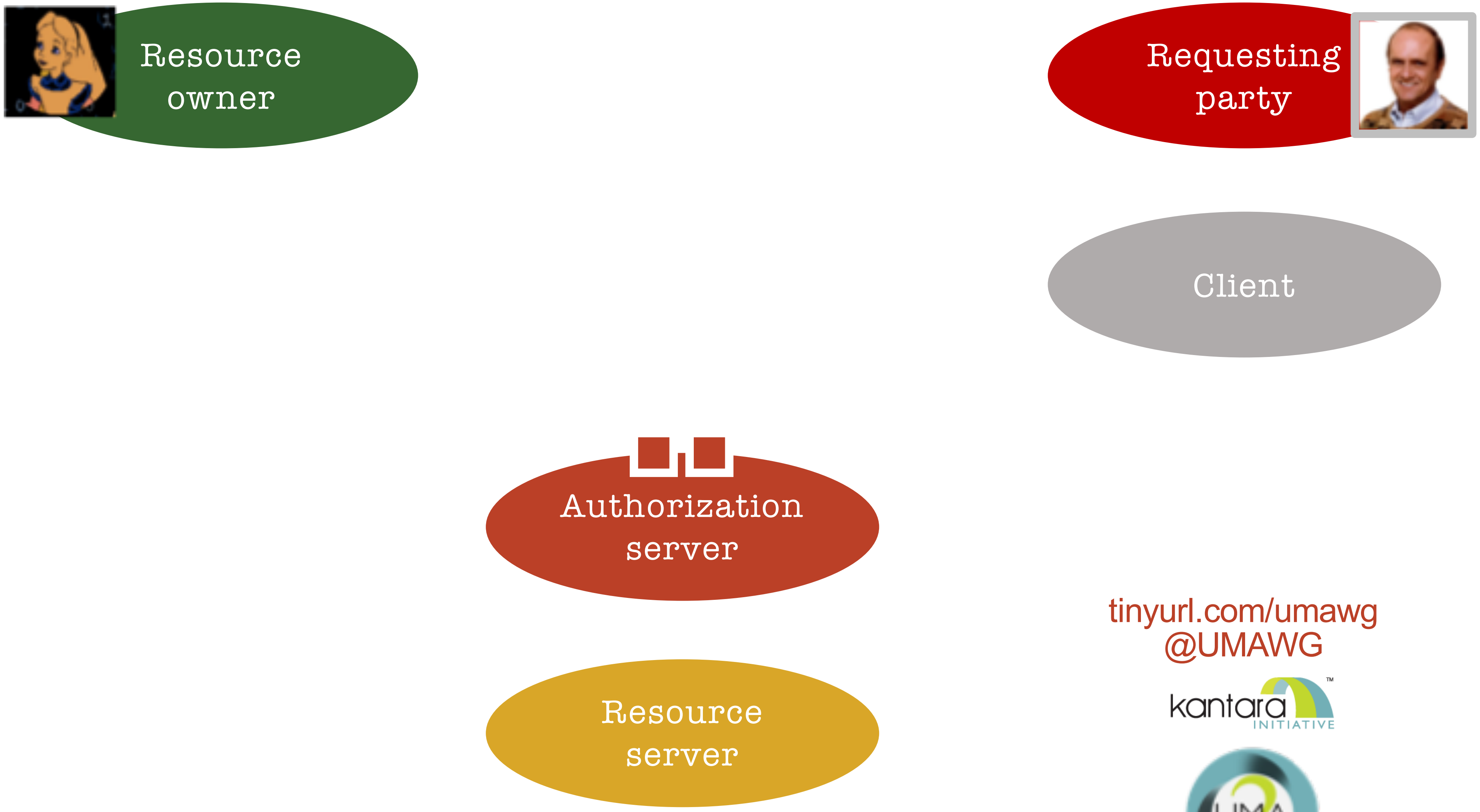
UMA brings next-gen delegation and consent to OAuth



tinyurl.com/umawg
@UMAWG

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth



Resource owner

Requesting party

Client

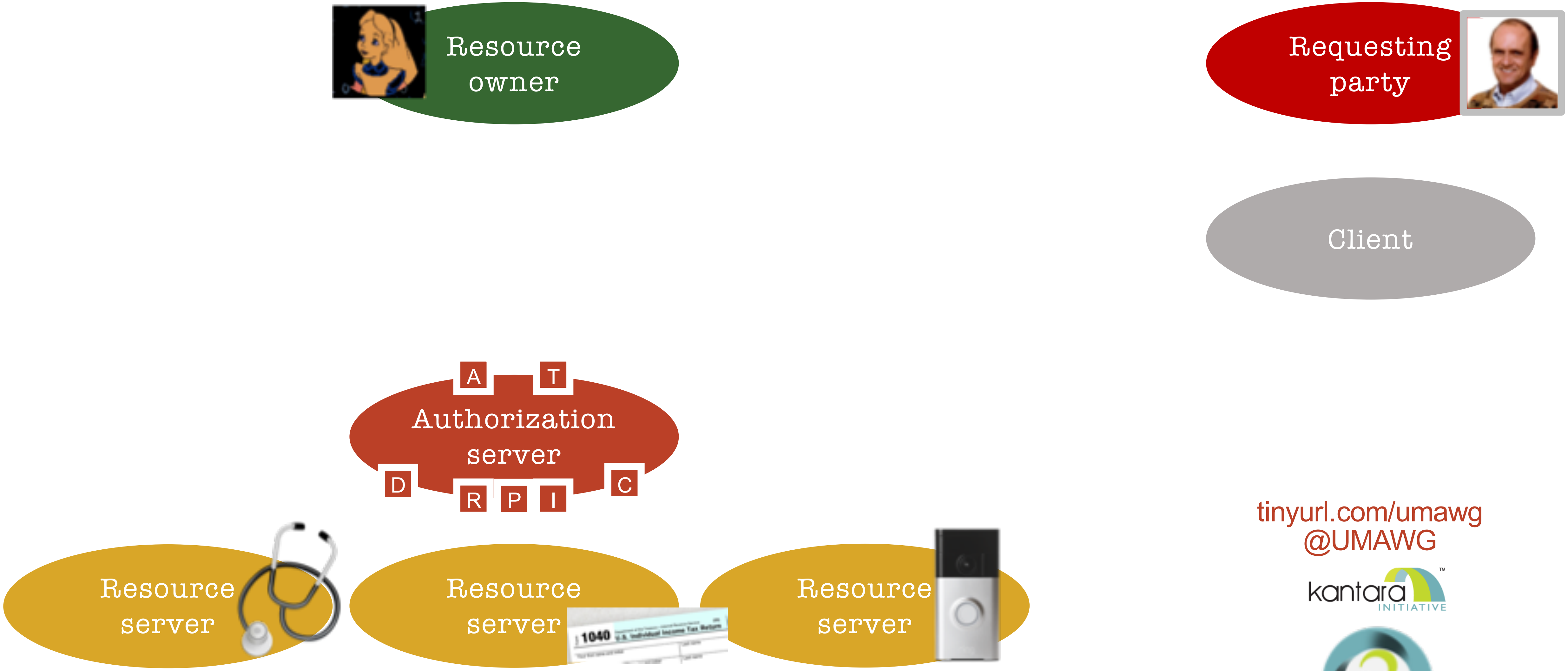
Authorization server

Resource server

tinyurl.com/umawg
@UMAWG

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth



Resource owner

Requesting party

Client

A T

Authorization server

D R P I C

Resource server

Resource server

Resource server

tinyurl.com/umawg
@UMAWG

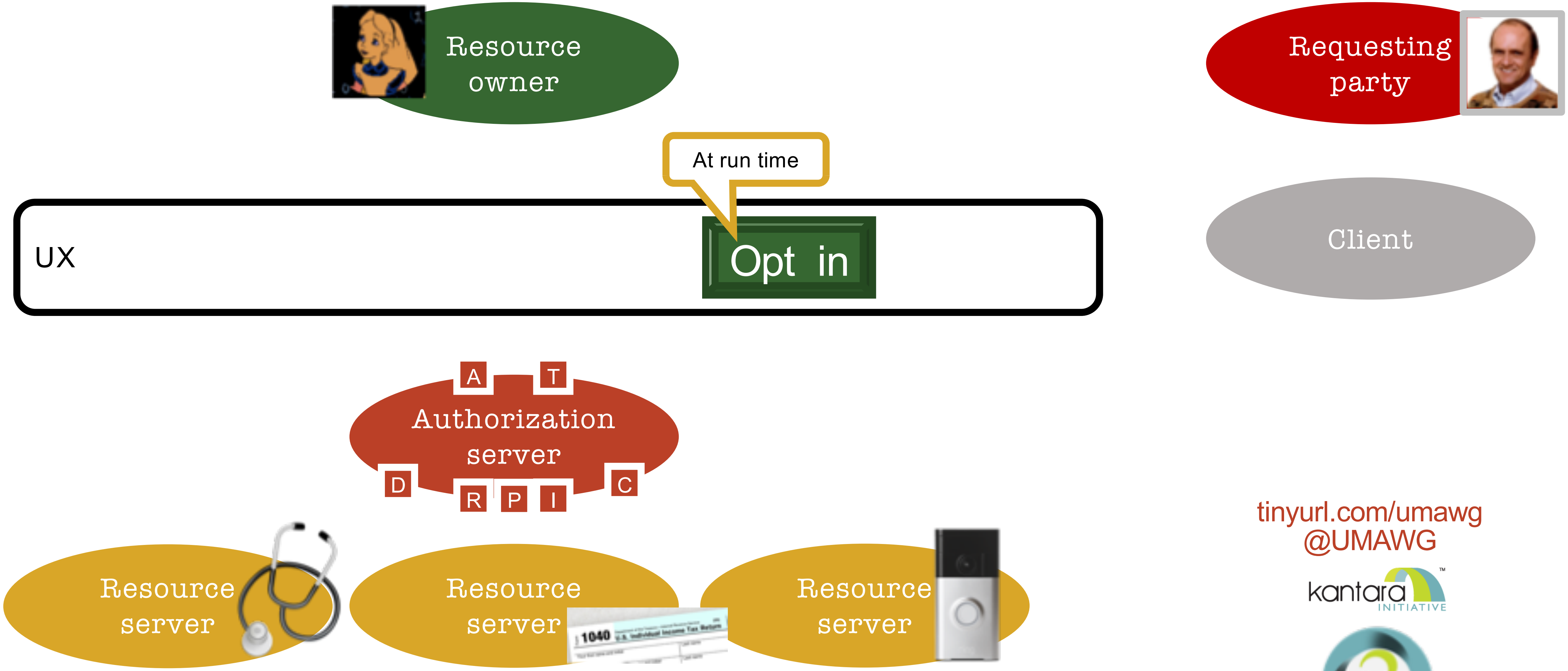
kantara INITIATIVE

UMA 2

A authorization T token D discovery R resource registration P permission I token introspection C claims interaction

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth

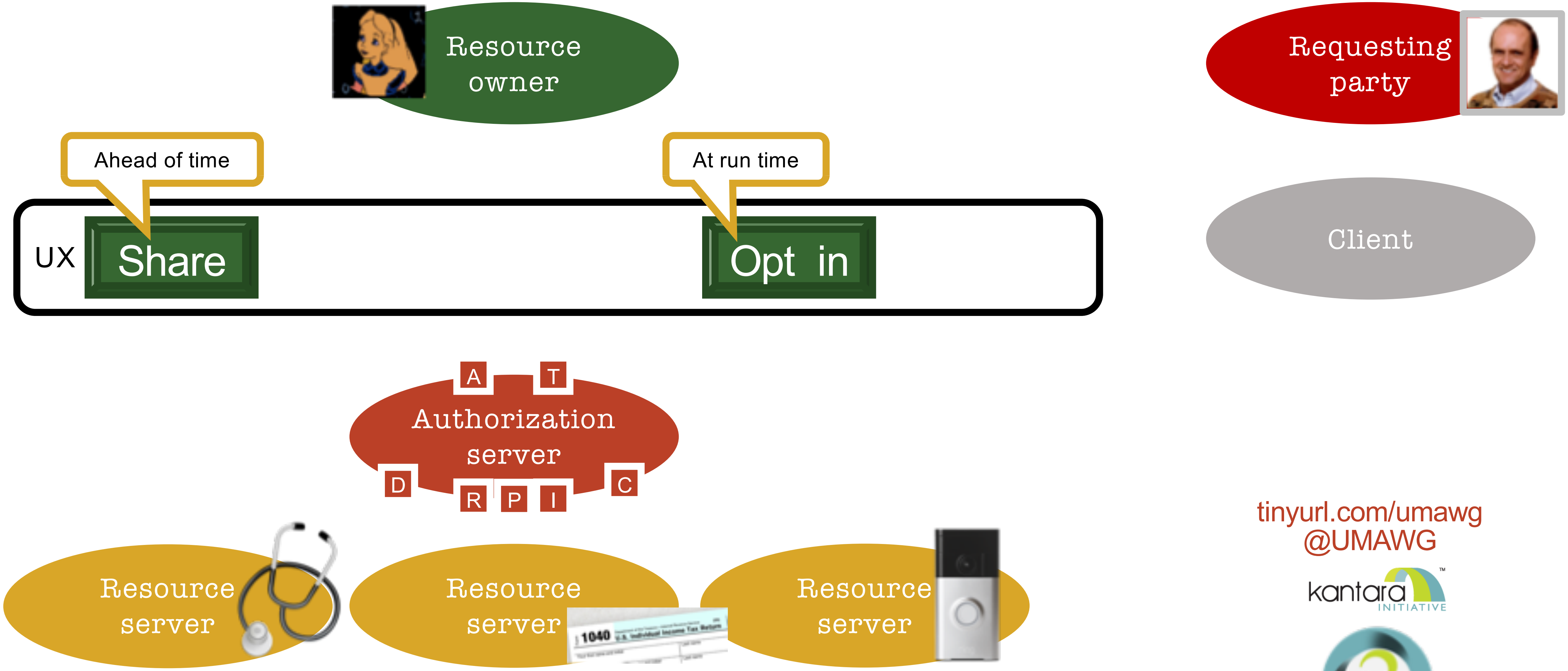


tinyurl.com/umawg
@UMAWG

A authorization T token D discovery R resource registration P permission I token introspection C claims interaction

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth



Resource owner

Requesting party

Ahead of time

At run time

UX

Share

Opt in

Client

A

T

Authorization server

D

C

R

P

I

Resource server

Resource server

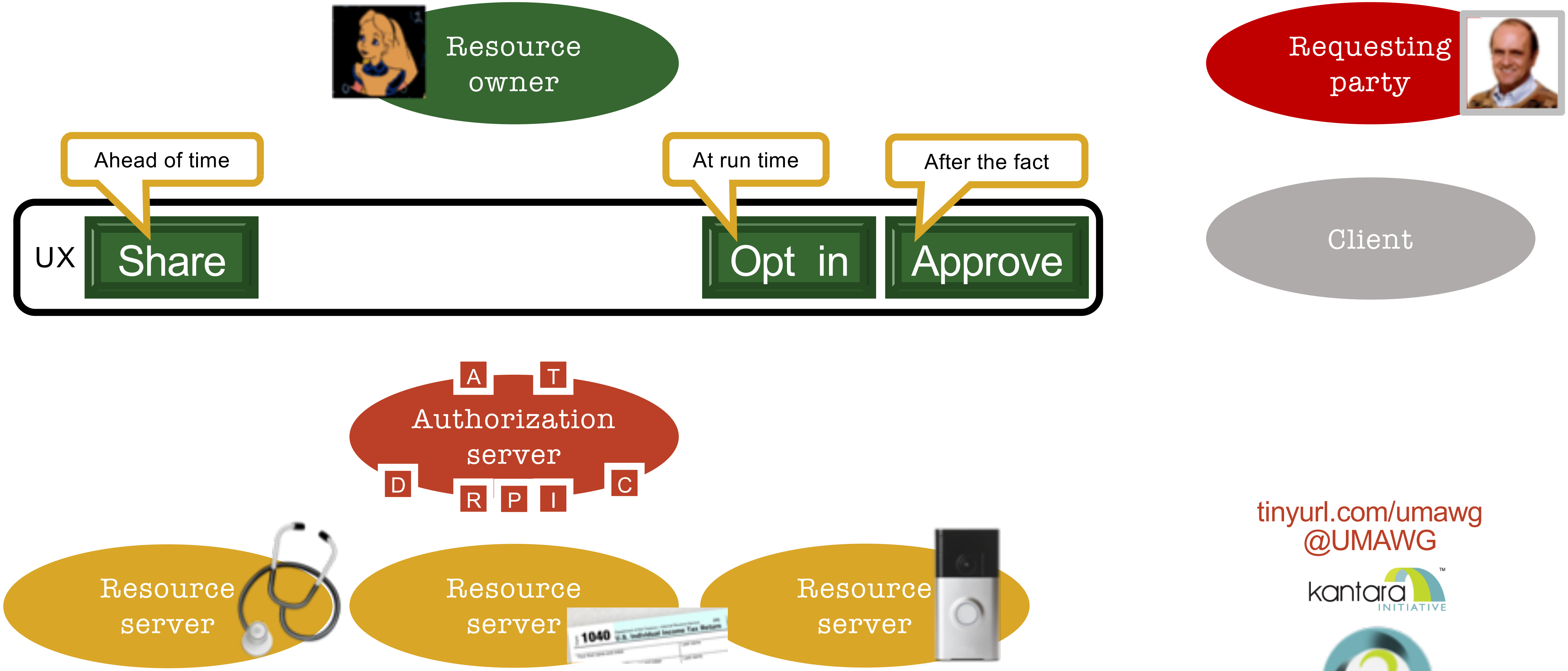
Resource server

tinyurl.com/umawg
@UMAWG

A authorization T token D discovery R resource registration P permission I token introspection C claims interaction

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth

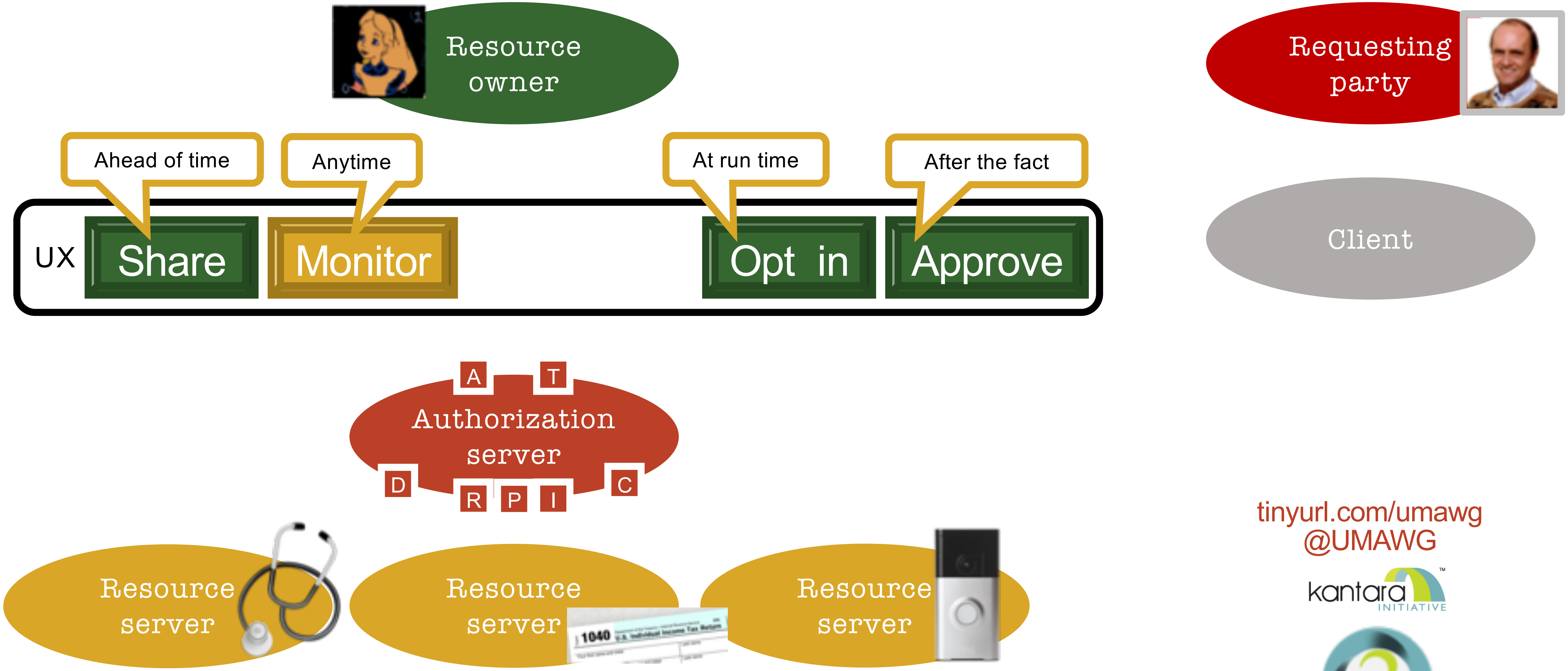


tinyurl.com/umawg
@UMAWG

A authorization T token D discovery R resource registration P permission I token introspection C claims interaction

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth

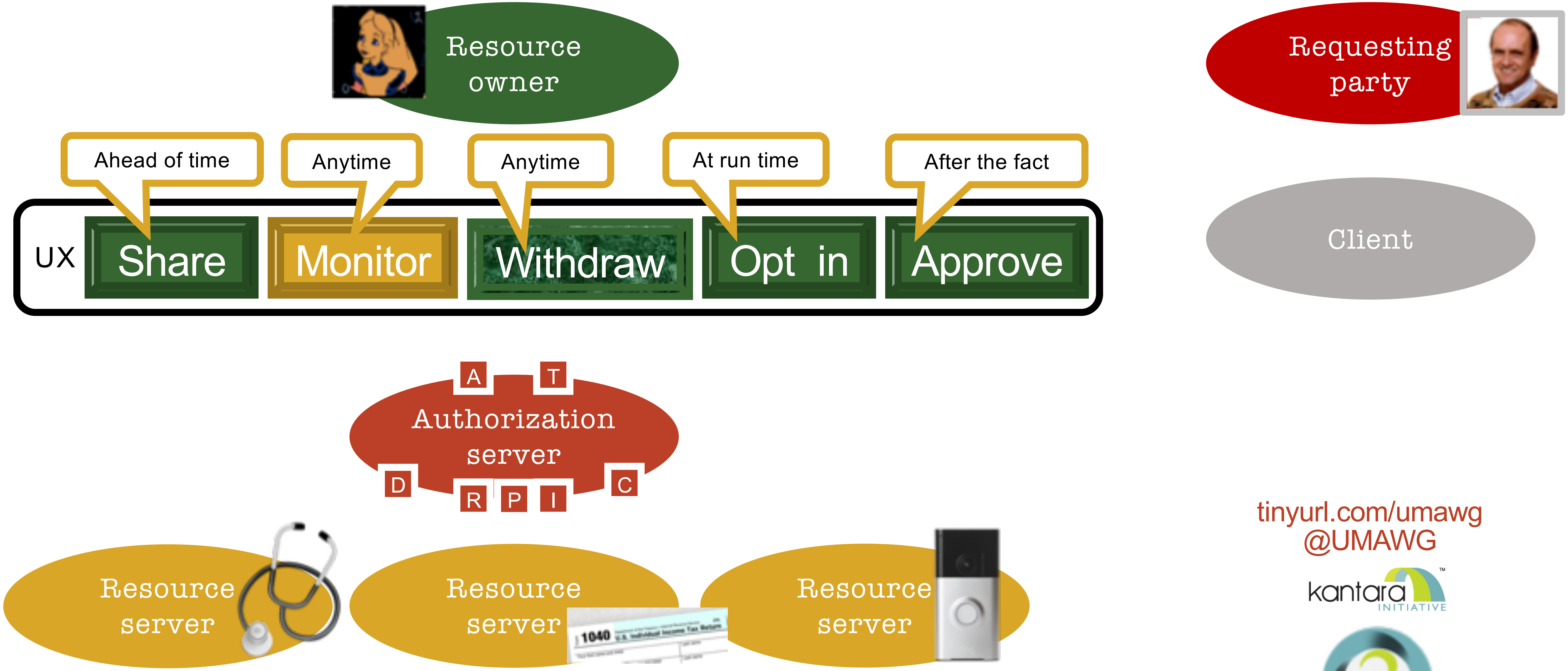


tinyurl.com/umawg
@UMAWG

A authorization T token D discovery R resource registration P permission I token introspection C claims interaction

User-Managed Access is for cross-party sharing

UMA brings next-gen delegation and consent to OAuth



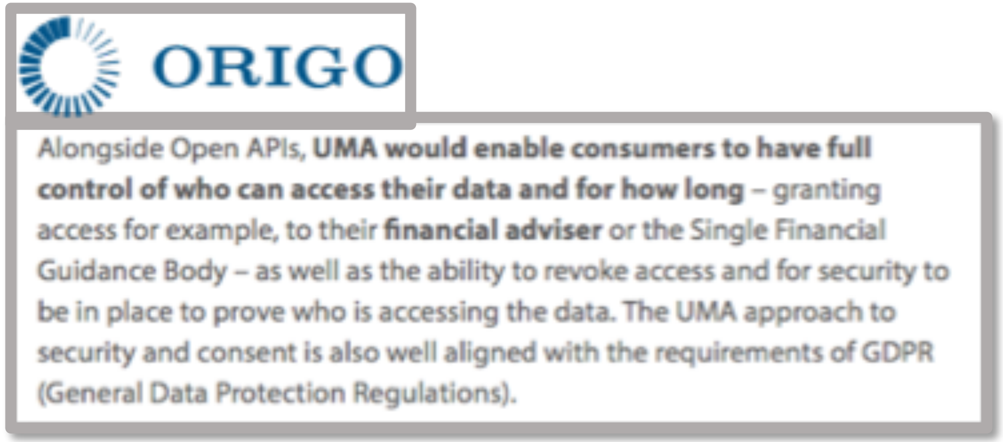
tinyurl.com/umawg
@UMAWG

A	authorization
T	token
D	discovery
R	resource registration
P	permission
I	token introspection
C	claims interaction

Like OpenID Connect for *identity*, UMA adds an API *access management* layer to OAuth2

Some use cases for UMA:

- Enterprise API protection
- For financial consumers
  - Discovering and aggregating UK pension accounts and sharing access to financial advisors
- In industrial and consumer IoT
  - For proactively or dynamically sharing smart device control or data with others
- Healthcare
  - As profiled in the Health Relationship Trust (HEART) WG at OpenID Foundation
  - Part of the new OpenMedReady framework for trustworthy remote care



ORIGO

Alongside Open APIs, **UMA would enable consumers to have full control of who can access their data and for how long** – granting access for example, to their **financial adviser** or the Single Financial Guidance Body – as well as the ability to revoke access and for security to be in place to prove who is accessing the data. The UMA approach to security and consent is also well aligned with the requirements of GDPR (General Data Protection Regulations).

To sum up:
UMA enhances OAuth as follows

The UMA2 Grant spec adds to OAuth2

- The resource owner authorizes protected resource access to clients used by entities that are in a requesting party role. This enables **party-to-party authorization**, rather than authorization of application access alone.
- The authorization server and resource server interact with the client and requesting party in a way that is **asynchronous** with respect to resource owner interactions.
- This lets a resource owner **configure an authorization server with policy conditions at will**, rather than authorizing access token issuance synchronously just after authenticating.

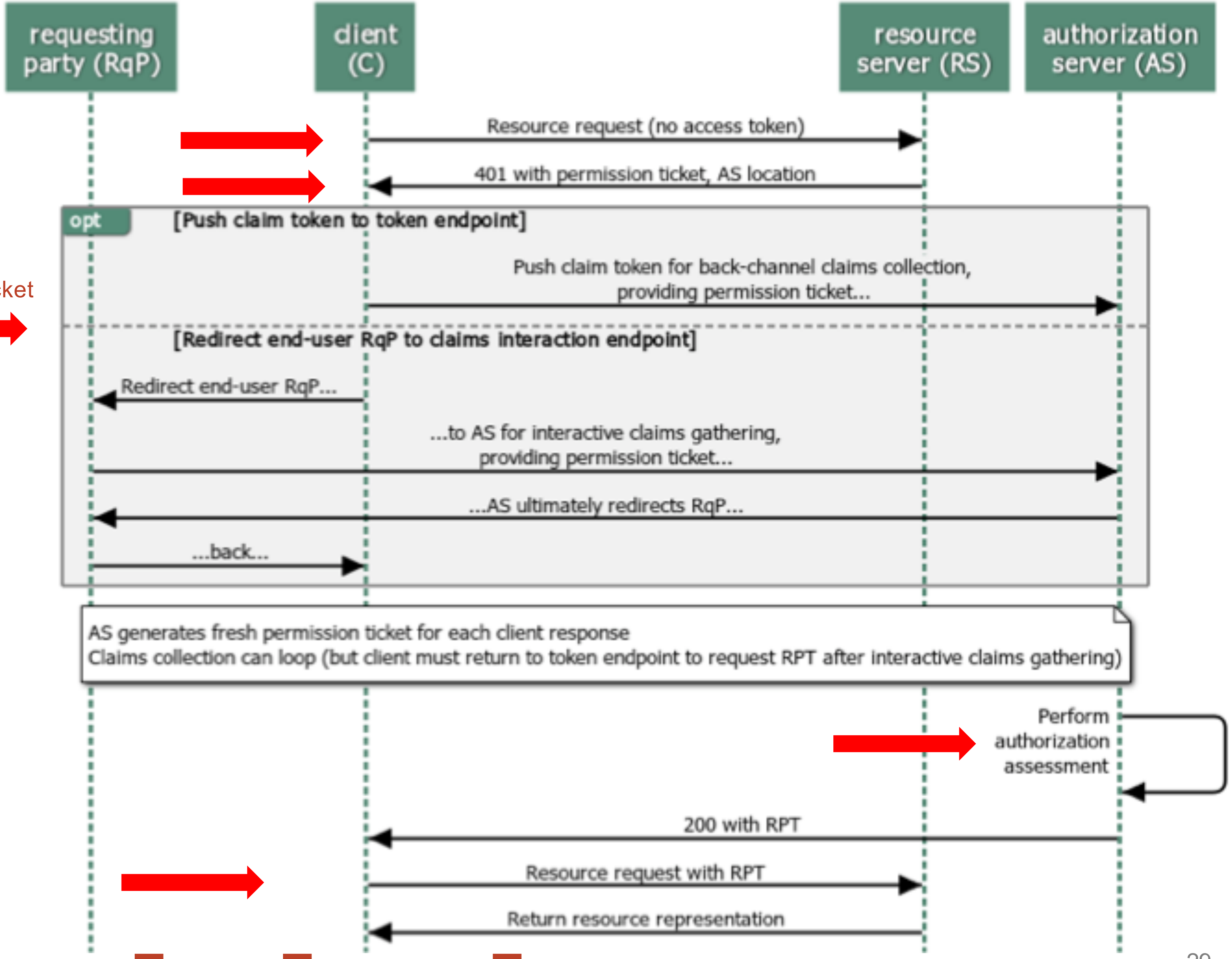
The UMA2 Federated Authorization spec adds to the UMA2 Grant

- **Multiple** resource servers operating in different domains can communicate with a **single** authorization server operating in yet another domain that acts on behalf of a resource owner.
- A service ecosystem can thus automate resource protection, and the **resource owner can monitor and control** authorization grant rules through the authorization server over time.
- Authorization grants can **increase and decrease** at the level of individual resources and scopes.

The UMA2 grant of OAuth: the basics

urn:ietf:params:oauth:grant-type:uma-ticket

(see also tinyurl.com/uma2grantwsd)



requesting party (RqP)

client (C)

resource server (RS)

authorization server (AS)

Resource request (no access token)

401 with permission ticket, AS location

opt [Push claim token to token endpoint]

Push claim token for back-channel claims collection, providing permission ticket...

[Redirect end-user RqP to claims interaction endpoint]

Redirect end-user RqP...

...to AS for interactive claims gathering, providing permission ticket...

...AS ultimately redirects RqP...

...back...

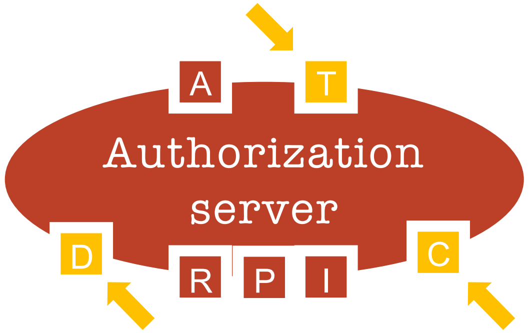
AS generates fresh permission ticket for each client response
Claims collection can loop (but client must return to token endpoint to request RPT after interactive claims gathering)

Perform authorization assessment

200 with RPT

Resource request with RPT

Return resource representation



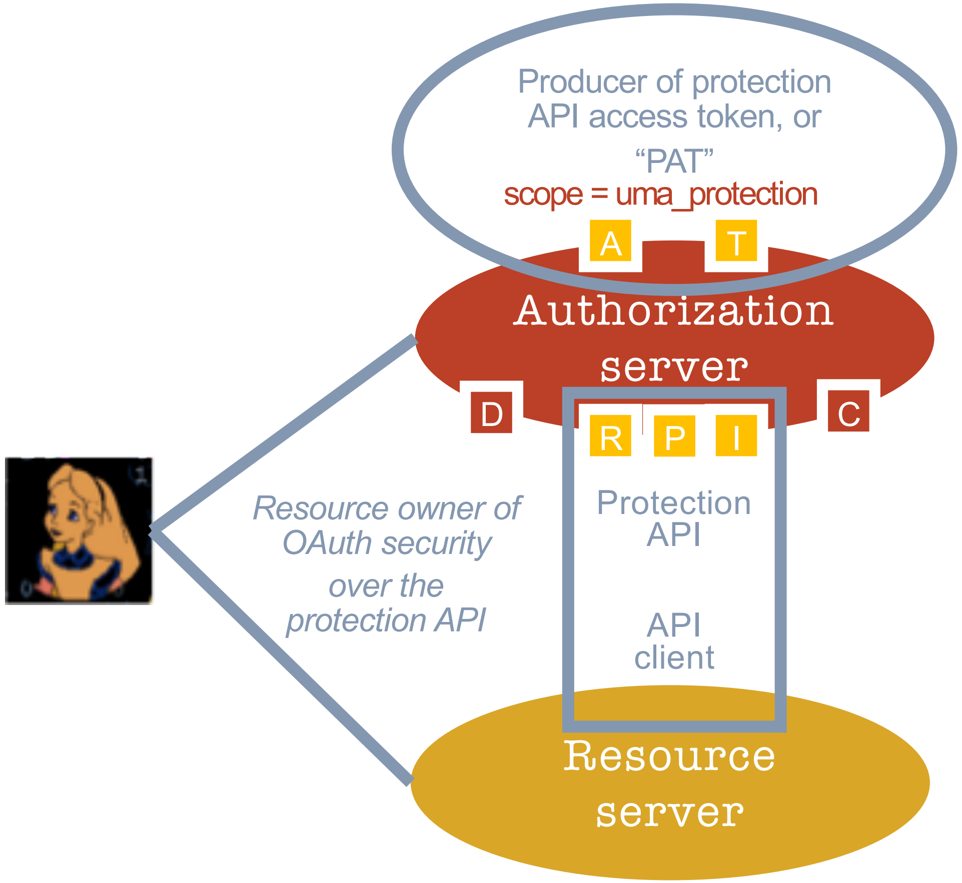
A authorization T token D discovery R resource registration P permission I token introspection C claims interaction

Other things to note about the UMA2 grant

- Types of token endpoint errors beyond vanilla OAuth:
  - need_info (403) with optional hints about what claims are needed
  - request_submitted (403) for RO action with optional polling interval
  - request_denied (403)
- The AS can issue a persisted claims token (**PCT**) with an RPT
  - The client can supply the PCT at the token endpoint later, refresh token-like, in hopes it will hasten RPT issuance without RqP involvement
- The client can ask for an RPT to be **upgraded**
- The client can ask for an RPT to be **revoked**
- Like some other grants, this one accommodates both **public** and **confidential** clients

Breaking apart the authorization server and resource server (externalizing authorization)

(see also tinyurl.com/uma2fawsd)


Producer of protection API access token, or "PAT"
scope = uma_protection
A
T
Authorization server
D
R
P
I
C
Resource owner of OAuth security over the protection API
Protection API
API client
Resource server

Protection API endpoints:

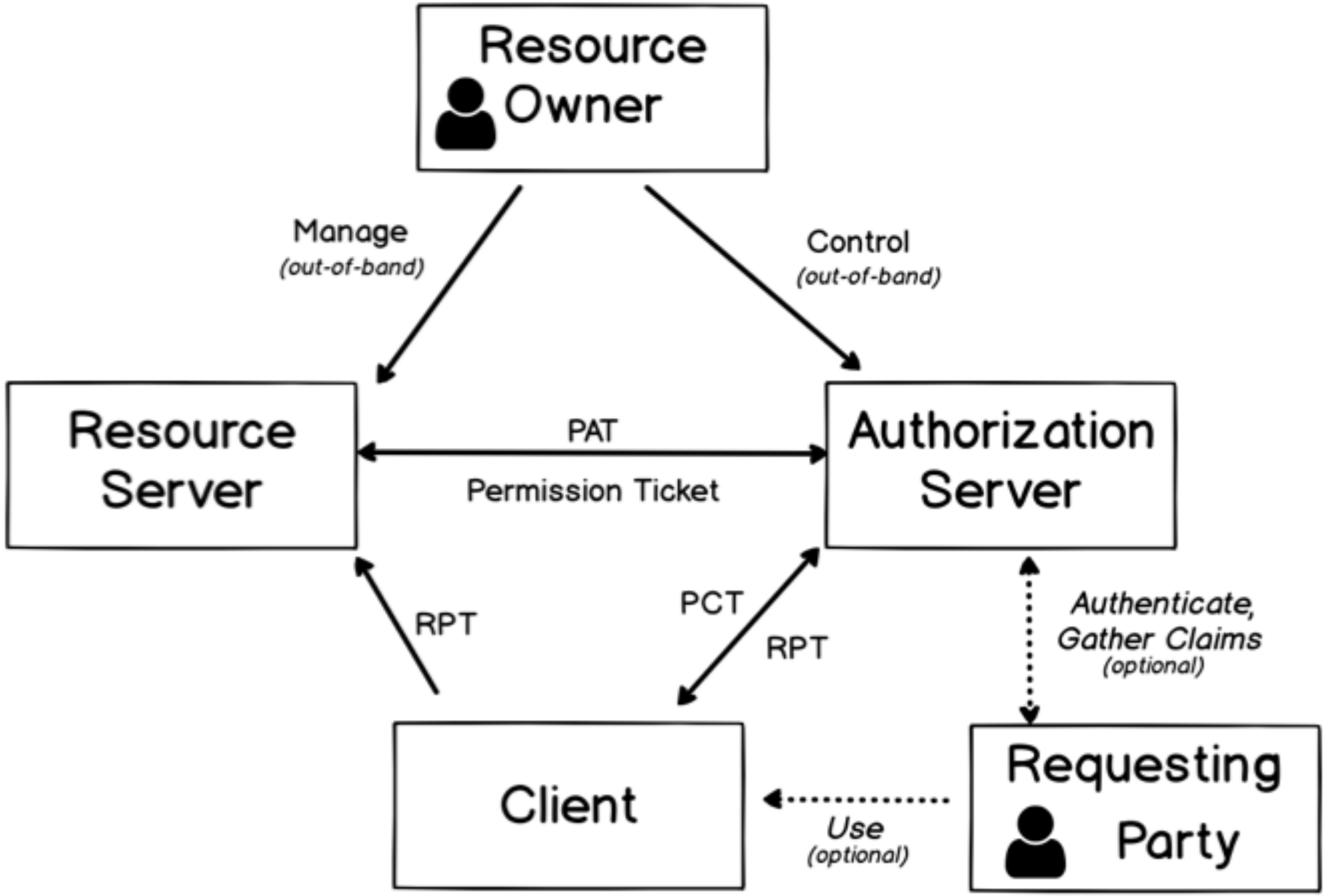
- **Resource registration:** Puts resources under AS protection;

  AS responds with resource IDs; resources can have ***unique scopes***

- **Permission:** Requests a ***permission ticket*** to deliver to the client after the tokenless resource request

- **Token introspection:** Customizes OAuth Token Introspection (RFC 7662) to ***enhance*** the token introspection response object

Demonstration by Mike



Resource Owner

Manage

(out-of-band)

Control

(out-of-band)

Resource Server

PAT

Permission Ticket

Authorization Server

RPT

PCT

RPT

Authenticate, Gather Claims

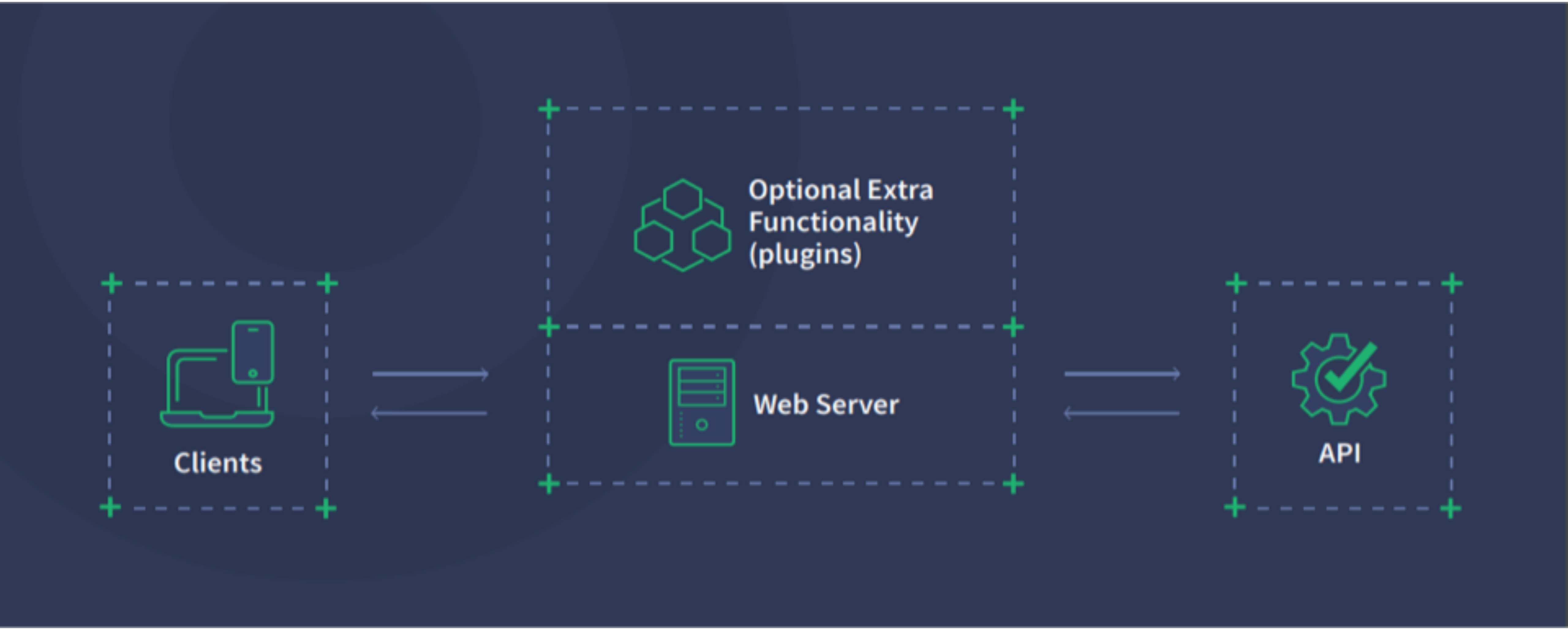
(optional)

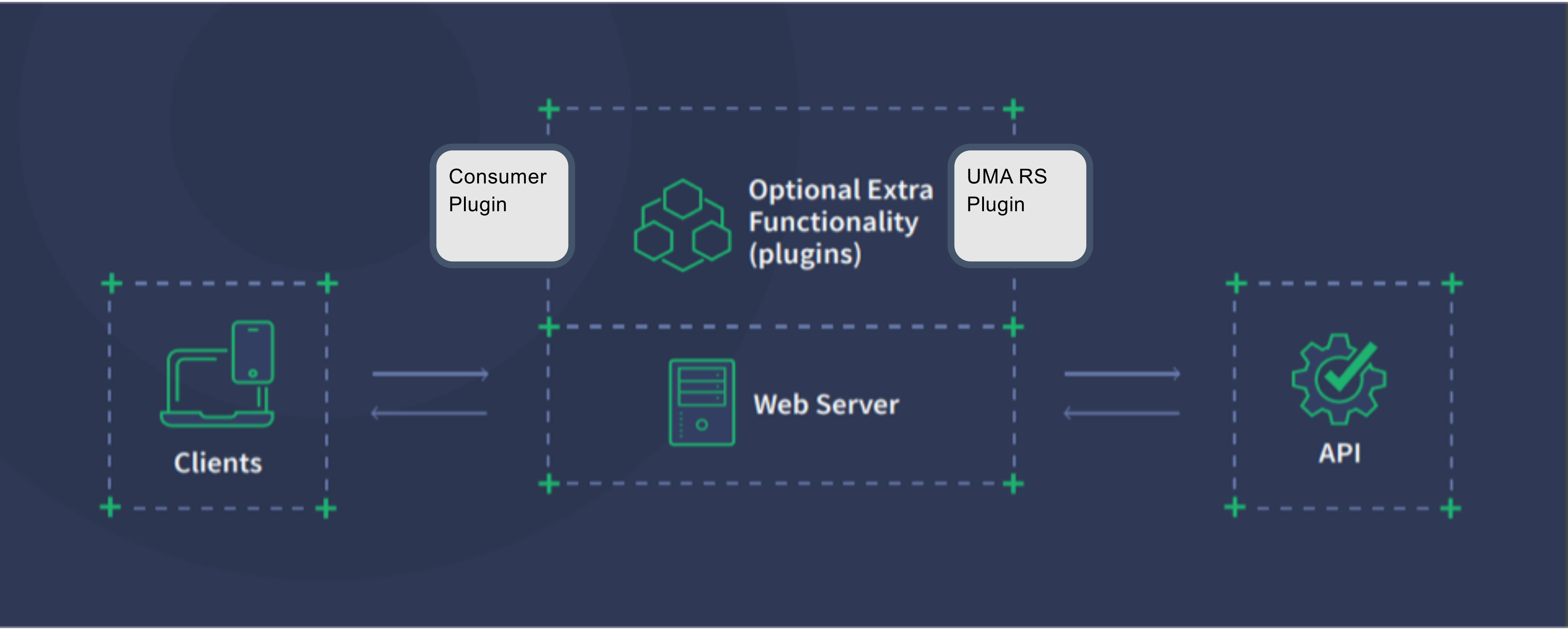
Client

Use

(optional)

Requesting Party


Optional Extra Functionality (plugins)
Clients
Web Server
API



Consumer Plugin

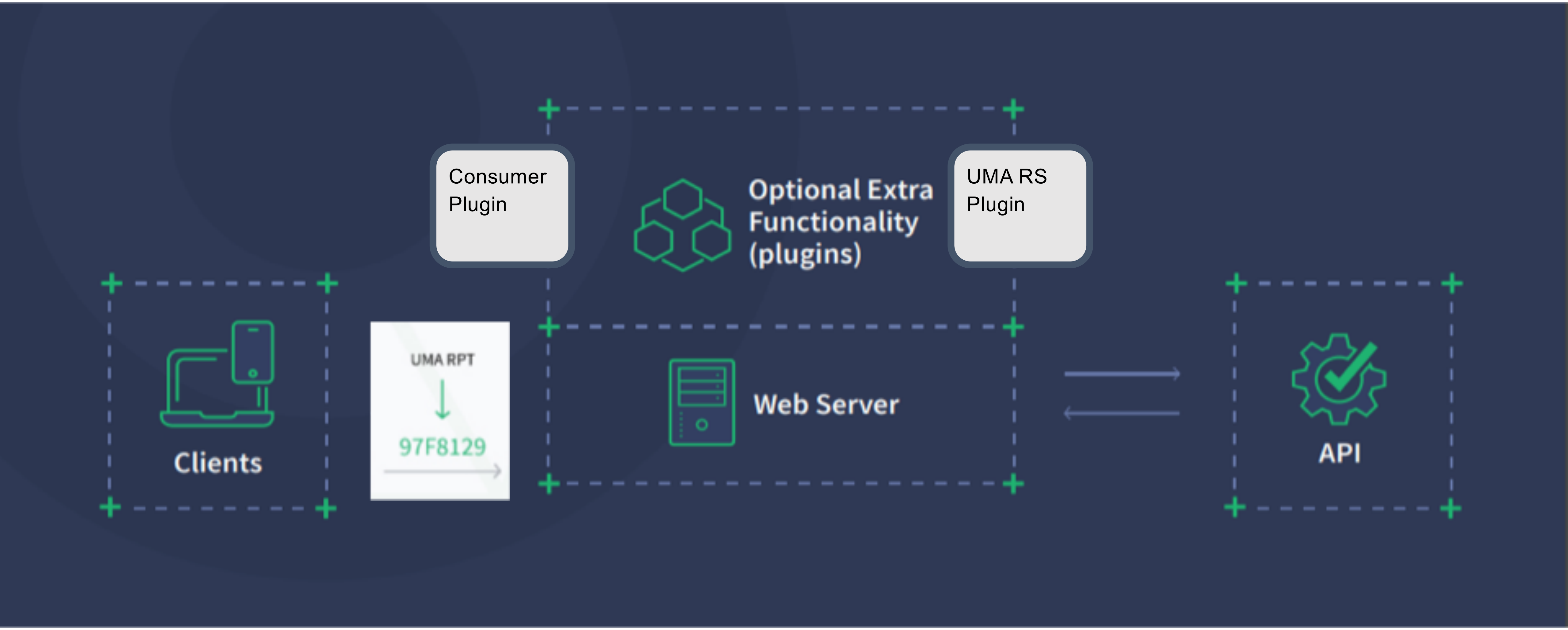
Optional Extra Functionality (plugins)

UMA RS Plugin

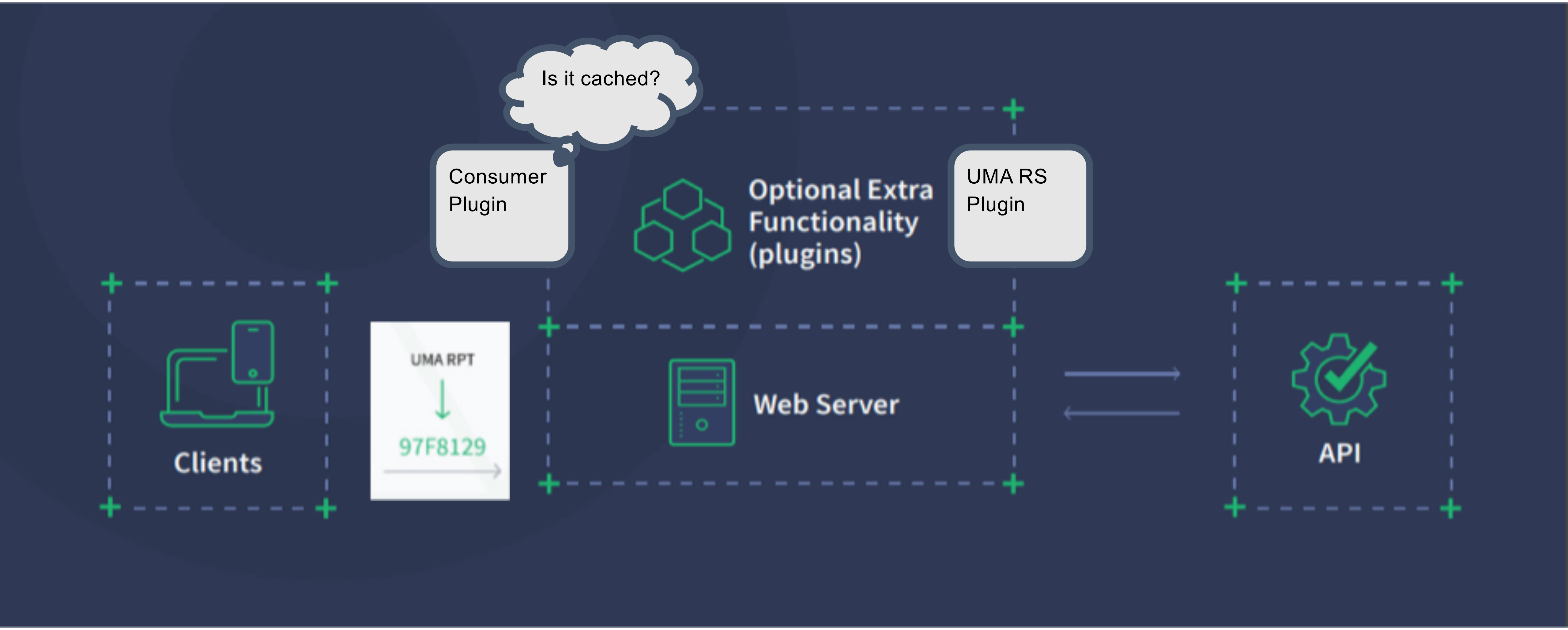
Clients

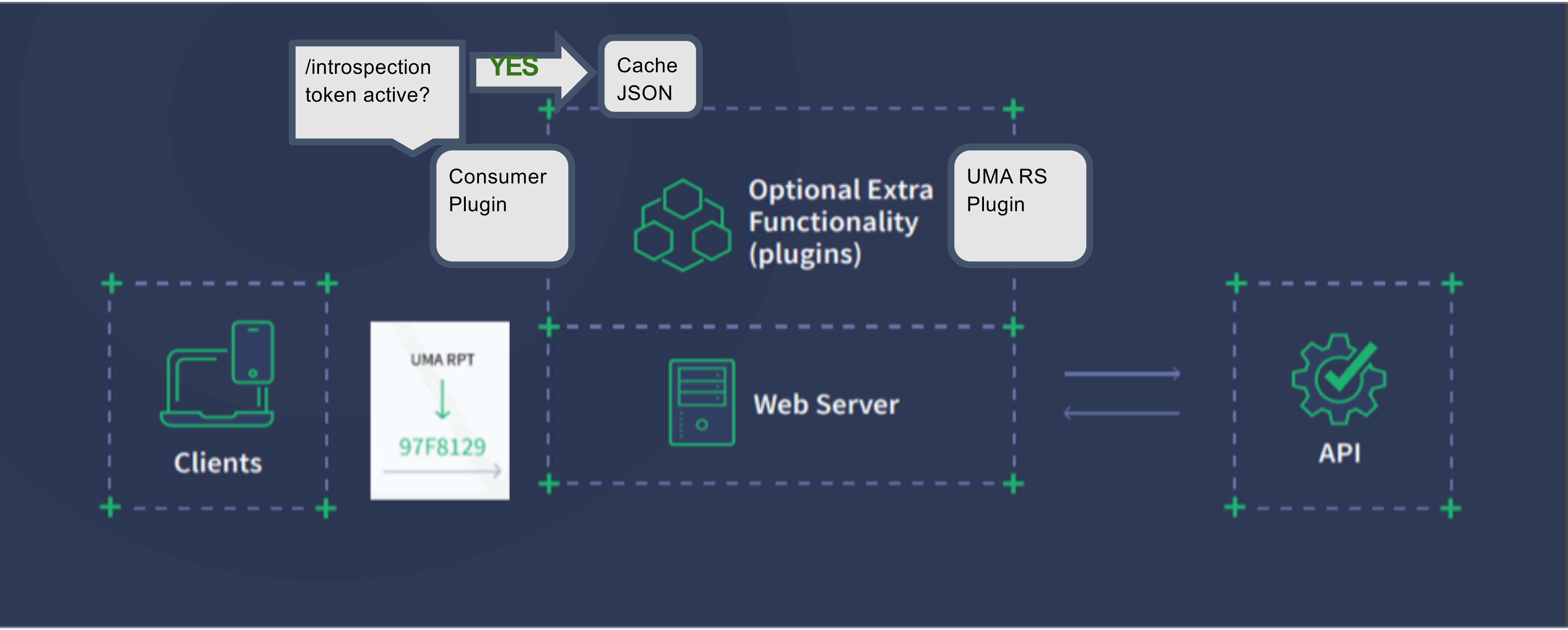
Web Server

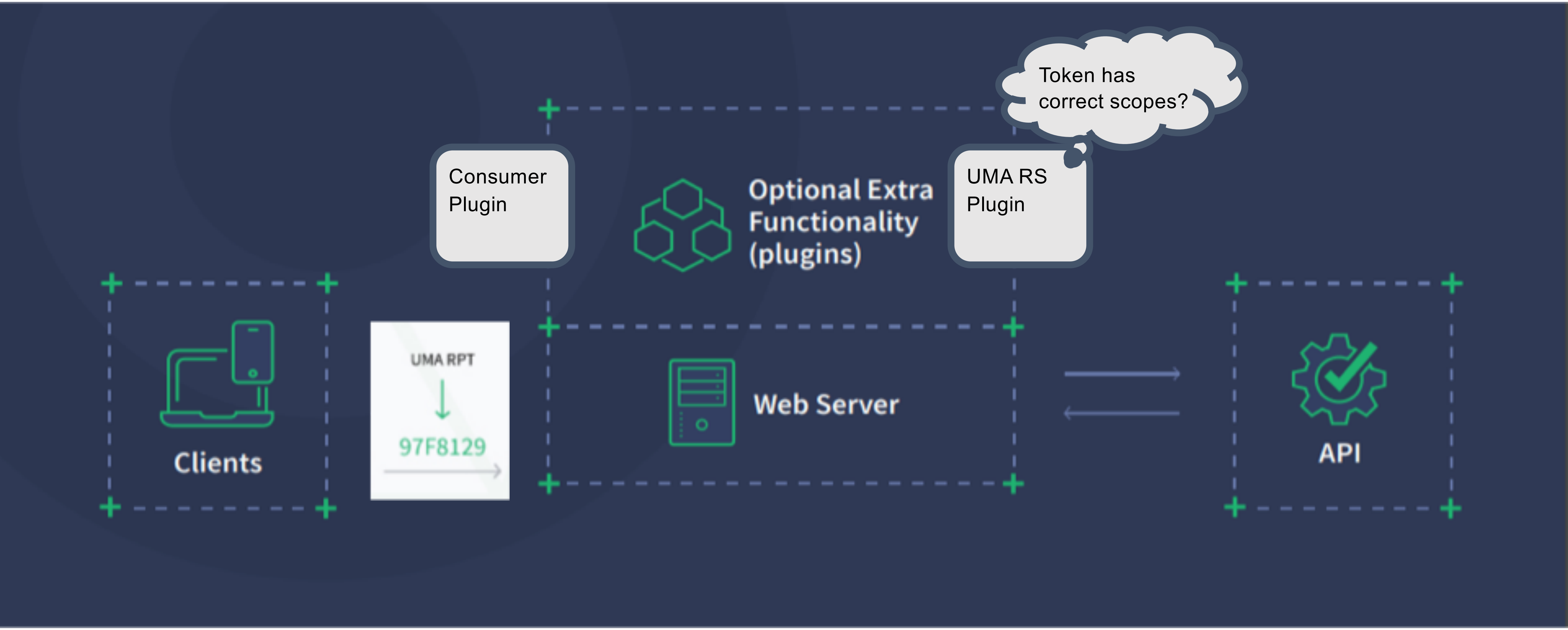
API

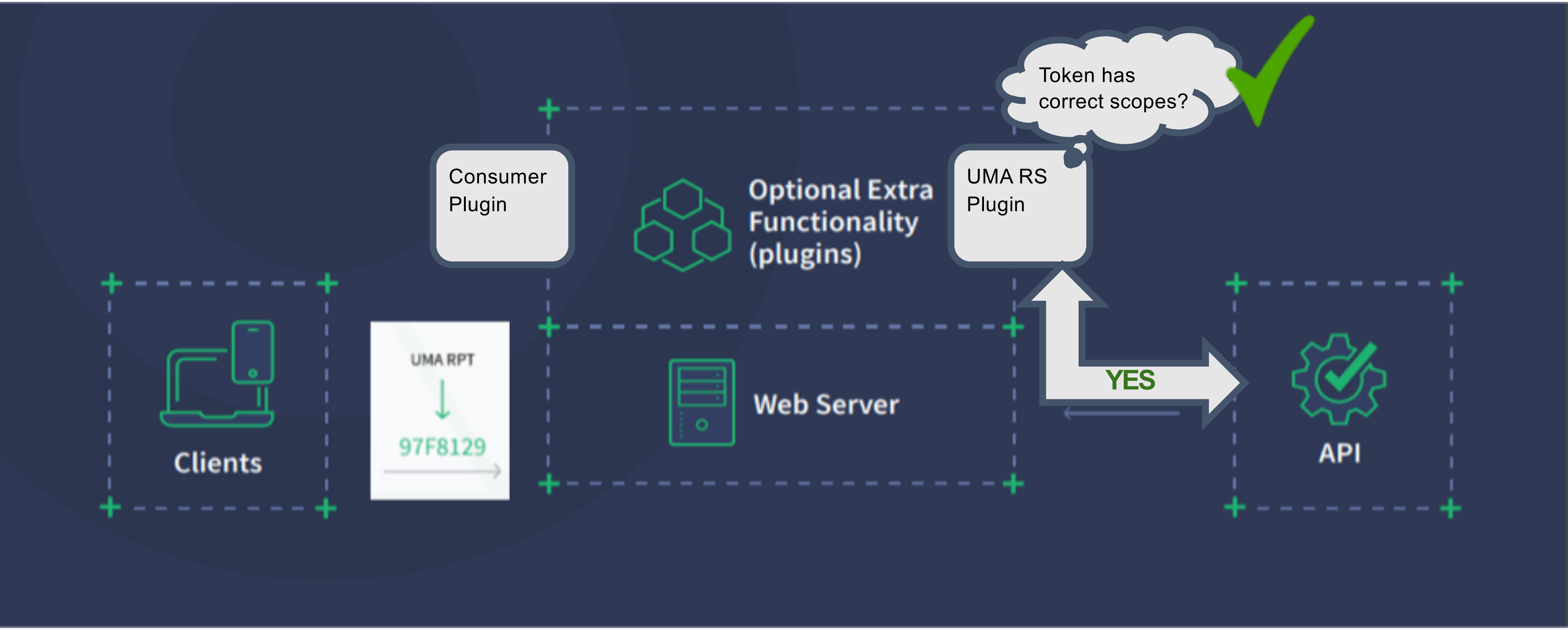


Consumer Plugin
Optional Extra Functionality (plugins)
UMA RS Plugin
Clients
UMA RPT
97F8129
Web Server
API


Is it cached?
Consumer Plugin
Optional Extra Functionality (plugins)
UMA RS Plugin
Clients
UMA RPT
97F8129
Web Server
API


/introspection
token active?
YES
Cache
JSON
Consumer
Plugin
Optional Extra
Functionality
(plugins)
UMA RS
Plugin
Clients
UMA RPT
97F8129
Web Server
API


Token has correct scopes?
Consumer Plugin
Optional Extra Functionality (plugins)
UMA RS Plugin
Clients
UMA RPT
97F8129
Web Server
API



Token has correct scopes?

Consumer Plugin

Optional Extra Functionality (plugins)

UMA RS Plugin

Clients

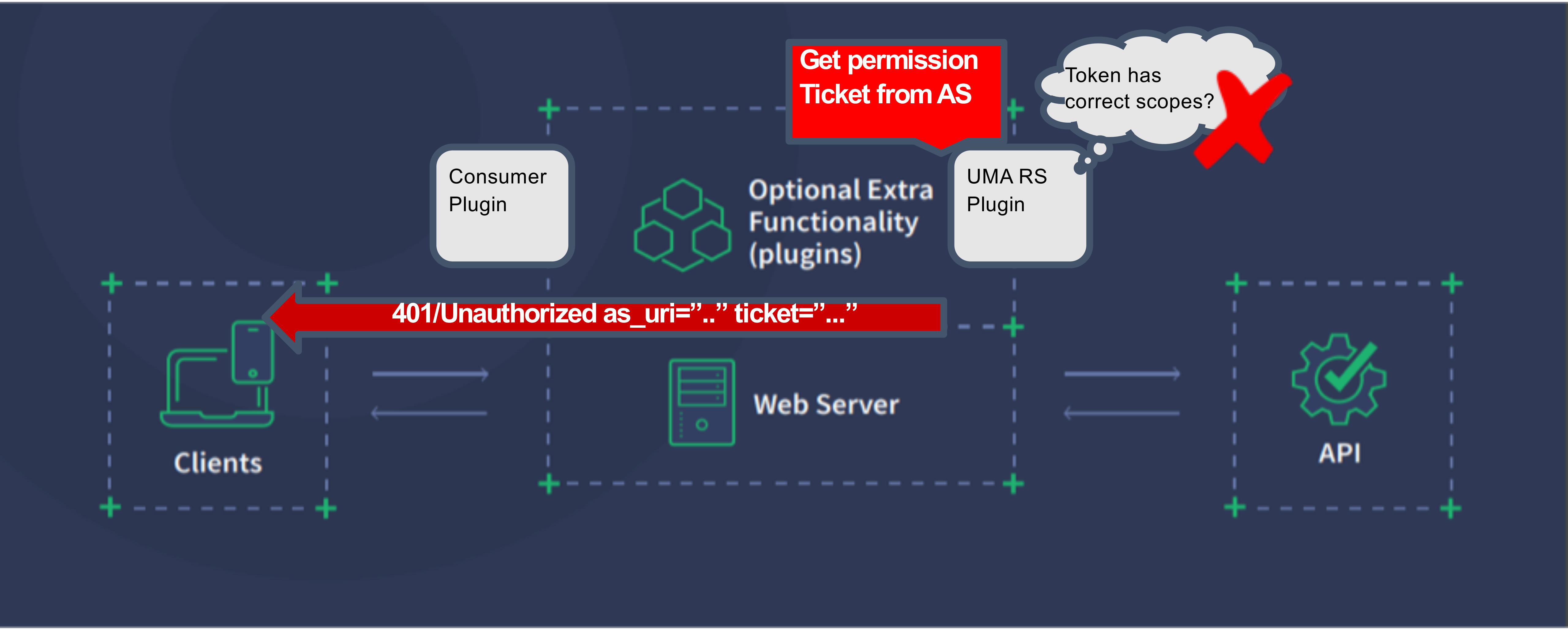
UMA RPT

97F8129

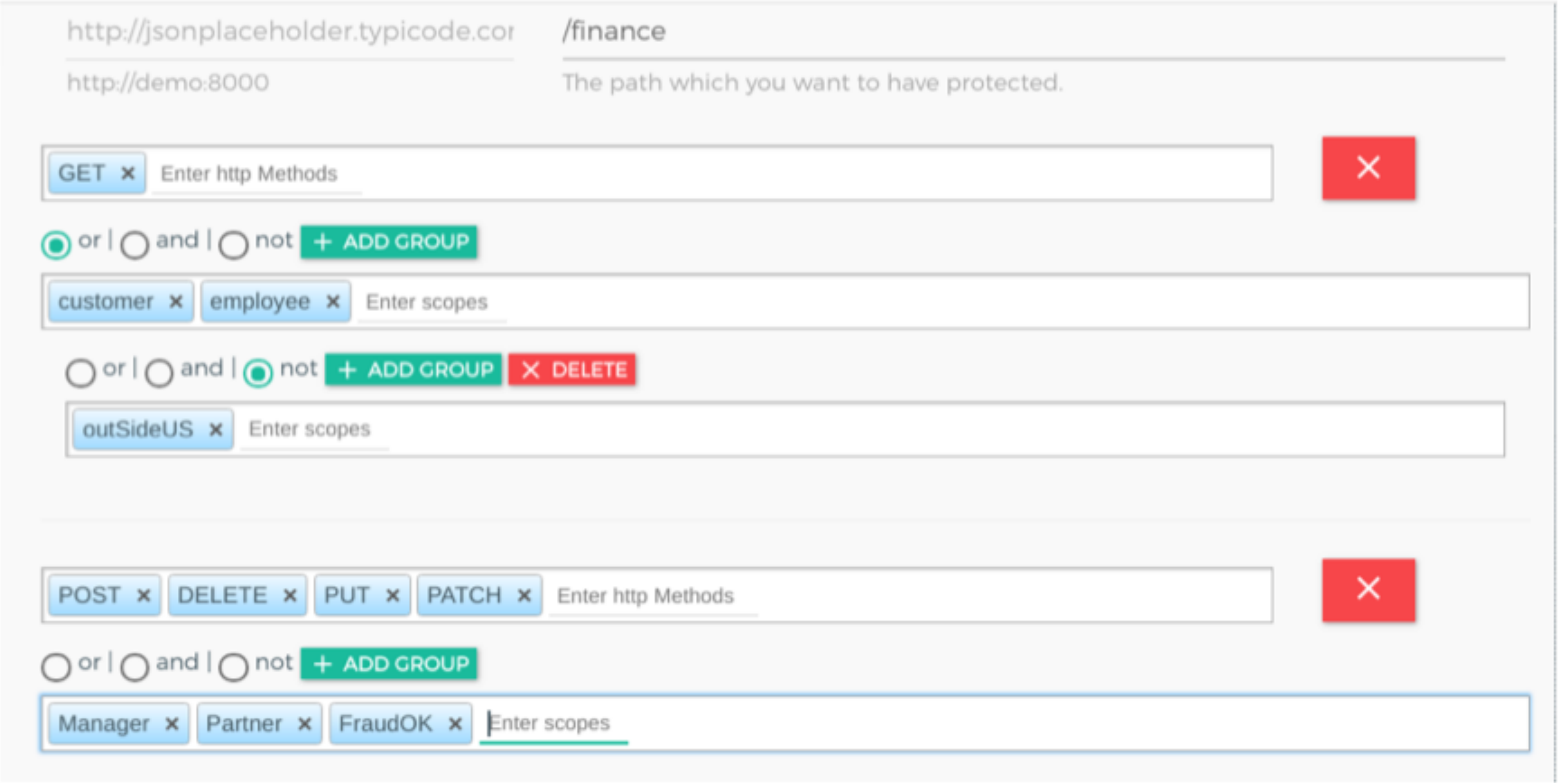
Web Server

YES

API

Get permission Ticket from AS
Token has correct scopes?
Consumer Plugin
Optional Extra Functionality (plugins)
UMA RS Plugin
401/Unauthorized as_uri=”..” ticket=”...”
Clients
Web Server
API

Restrict (URL & METHOD) to UMA scopes



http://jsonplaceholder.typicode.cor

http://demo:8000

/finance

The path which you want to have protected.

GET × Enter http Methods

● or | ○ and | ○ not

customer × employee × Enter scopes

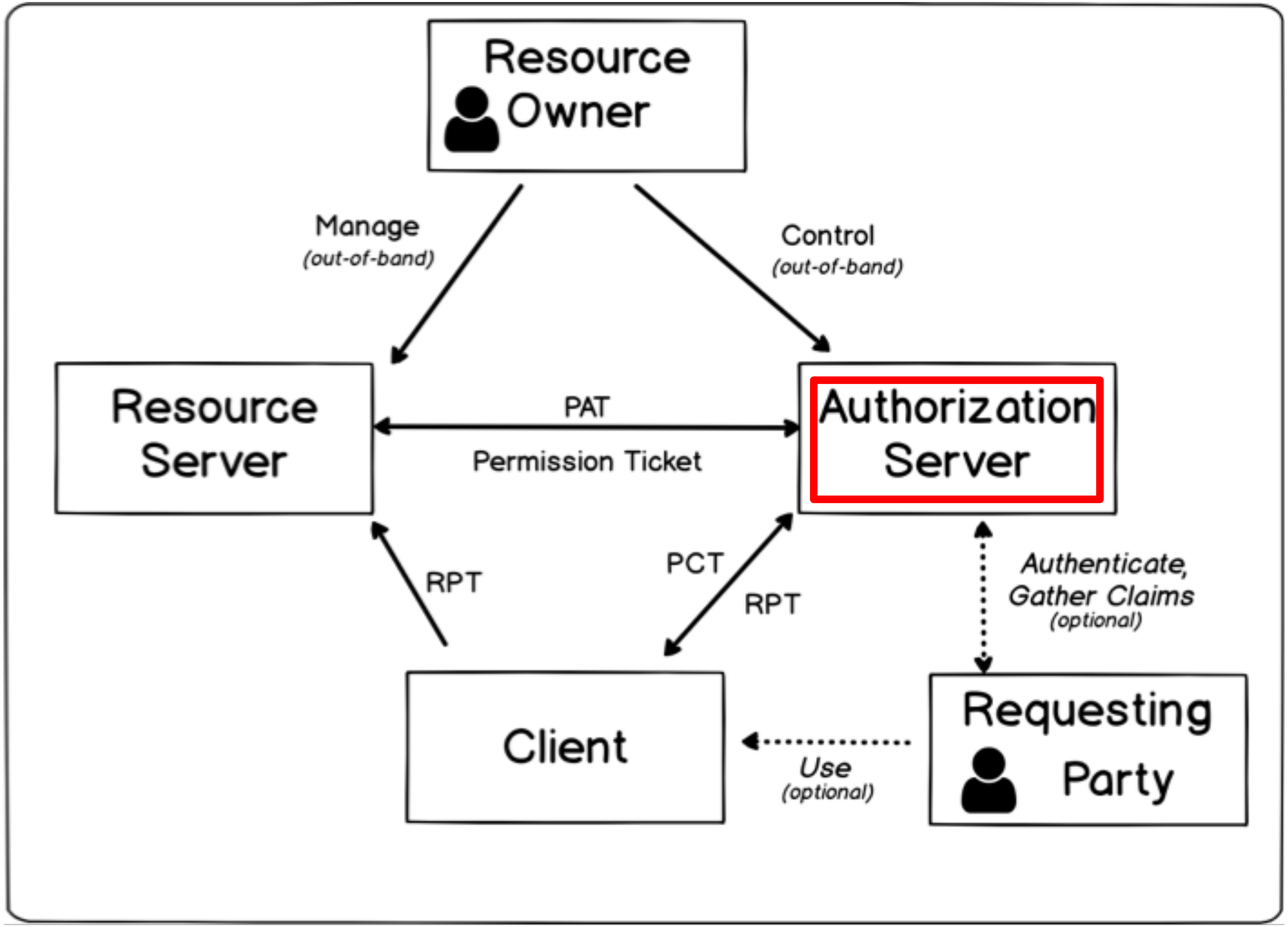
○ or | ○ and | ● not

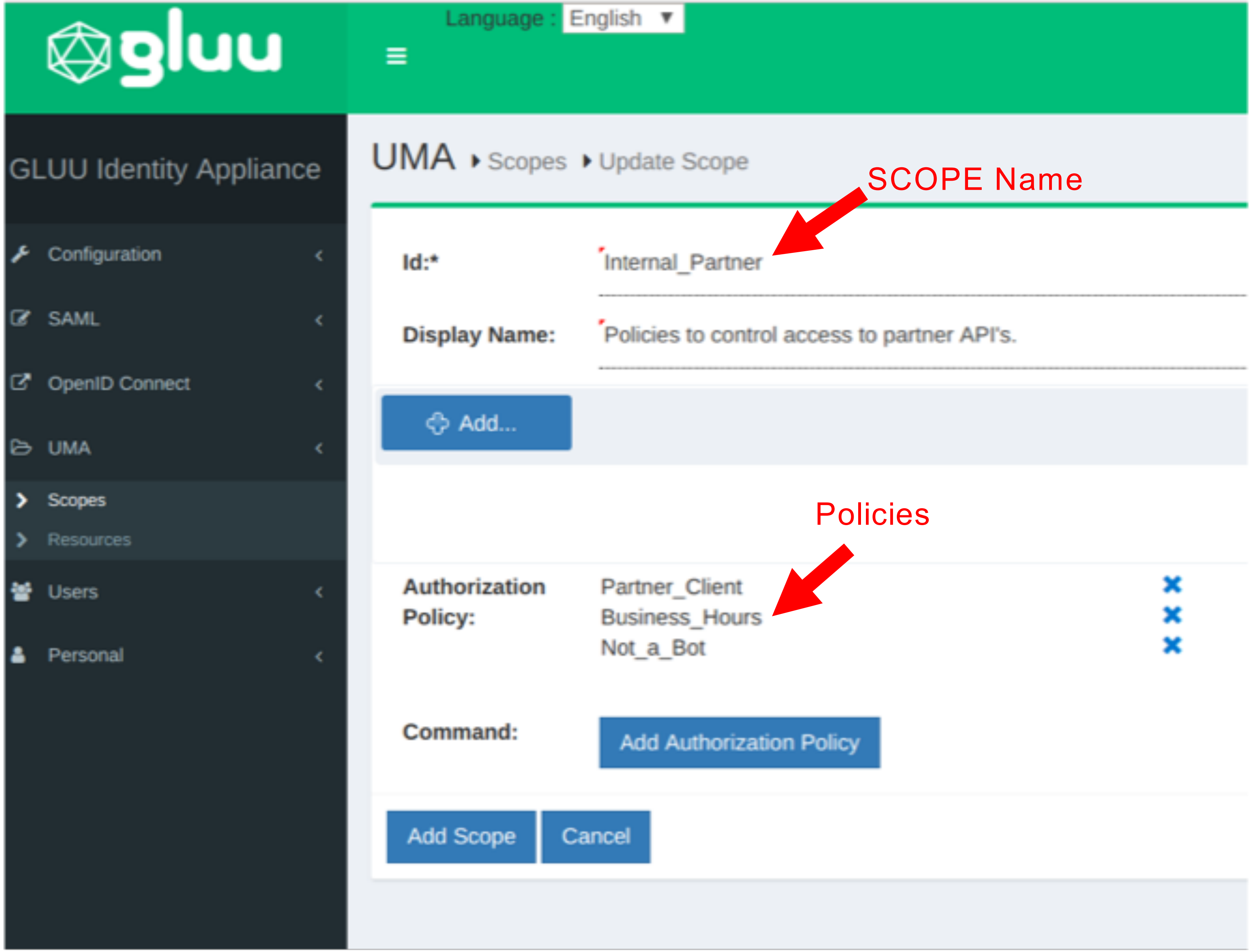
outSideUS × Enter scopes

POST × DELETE × PUT × PATCH × Enter http Methods

○ or | ○ and | ○ not

Manager × Partner × FraudOK × Enter scopes


Resource Owner
Manage
(out-of-band)
Control
(out-of-band)
Resource Server
PAT
Permission Ticket
Authorization Server
RPT
PCT
RPT
Authenticate,
Gather Claims
(optional)
Client
Use
(optional)
Requesting Party



UMA ▸ Scopes ▸ Update Scope

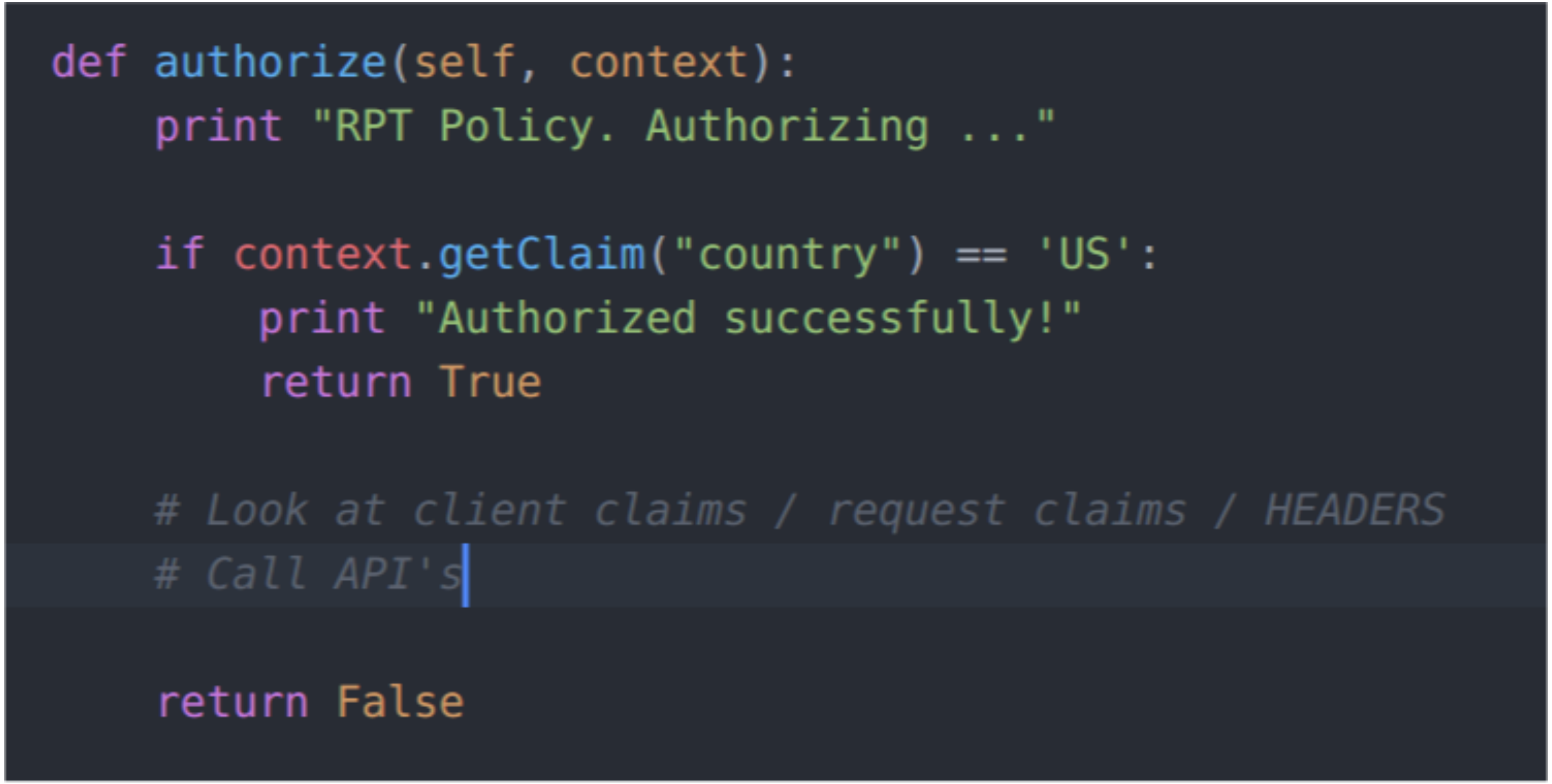
Id:* Internal_Partner

Display Name: Policies to control access to partner API's.

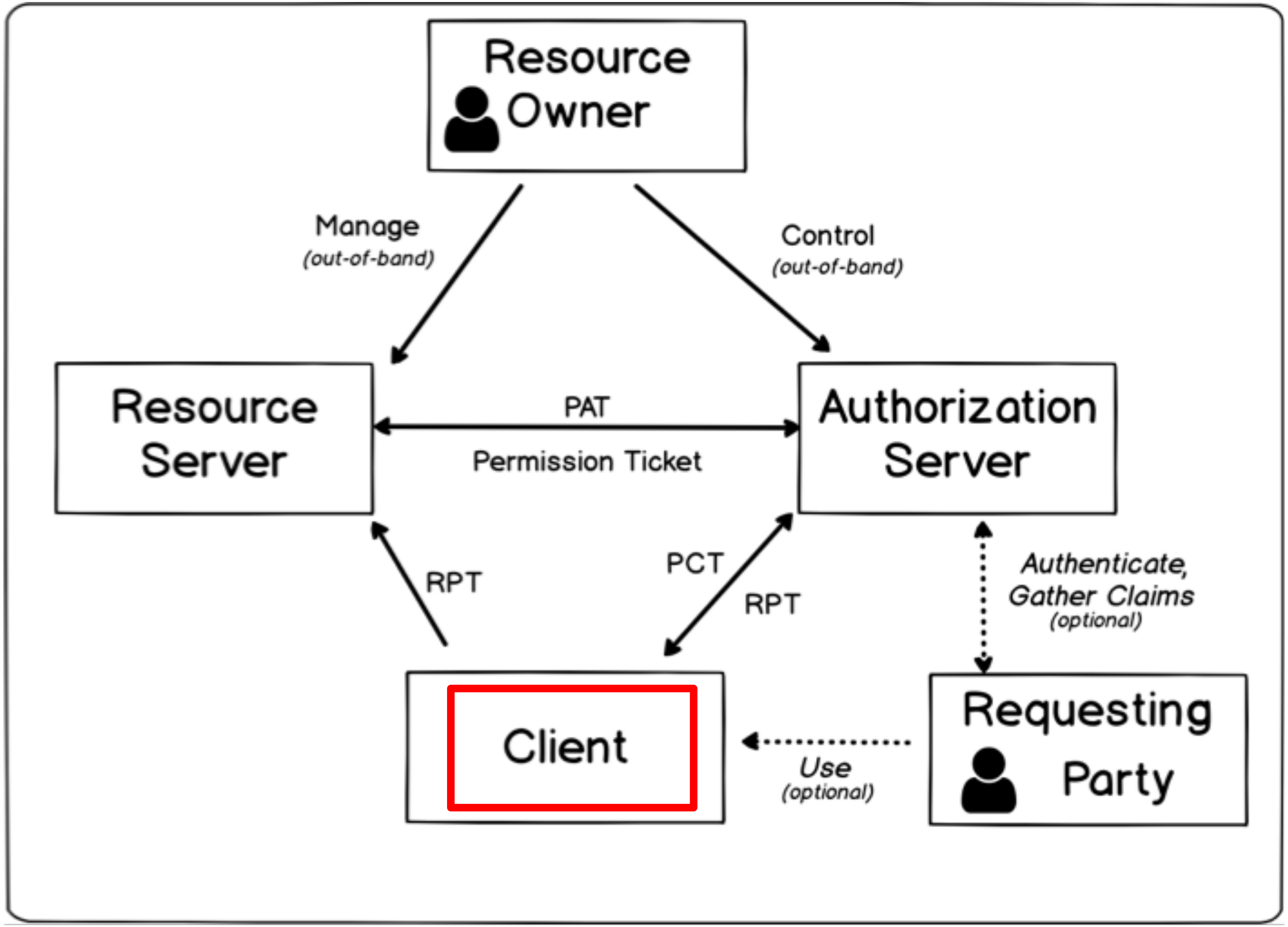
Authorization Policy:
Partner_Client
Business_Hours
Not_a_Bot

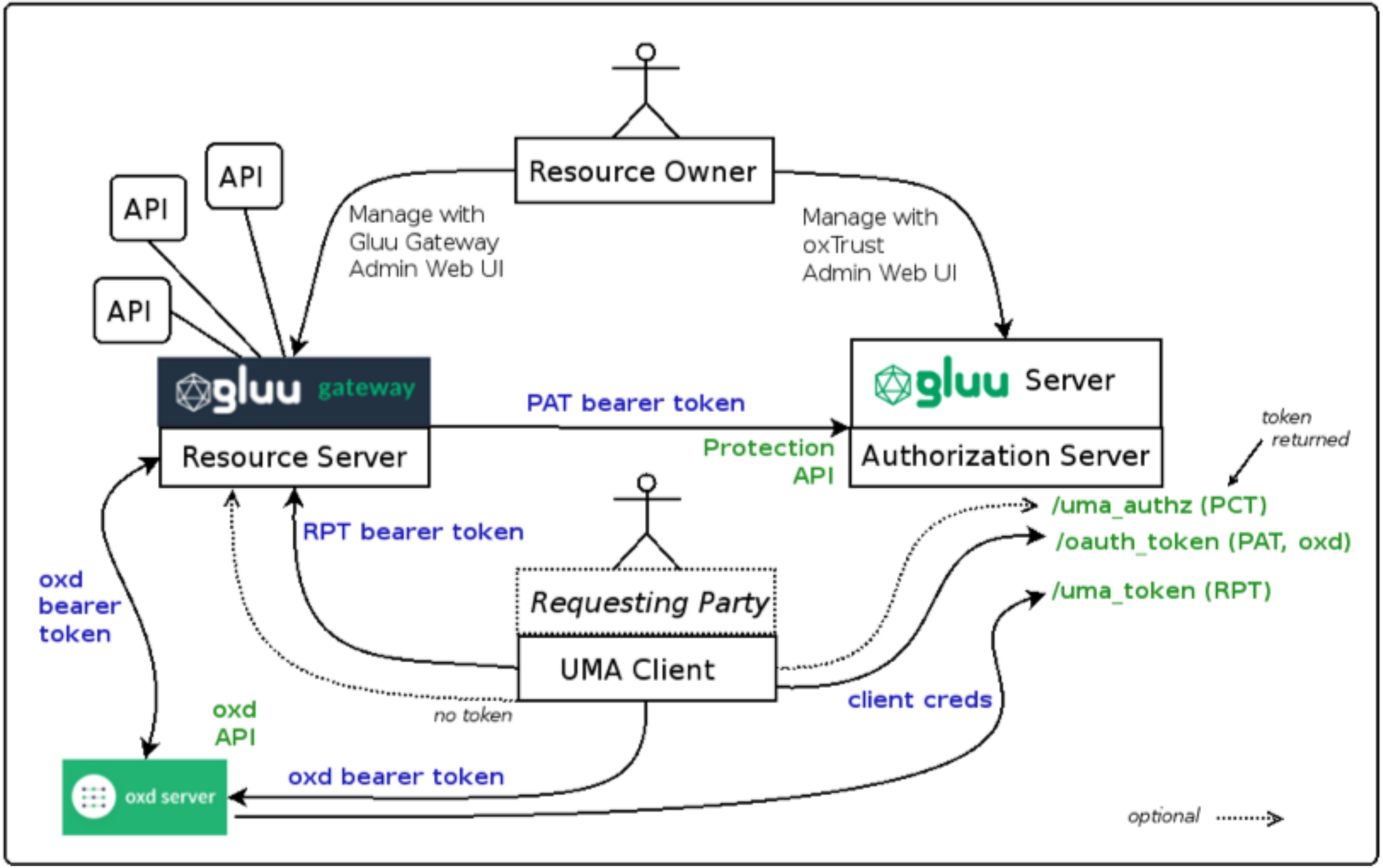
Command: Add Authorization Policy

Sample RPT policy



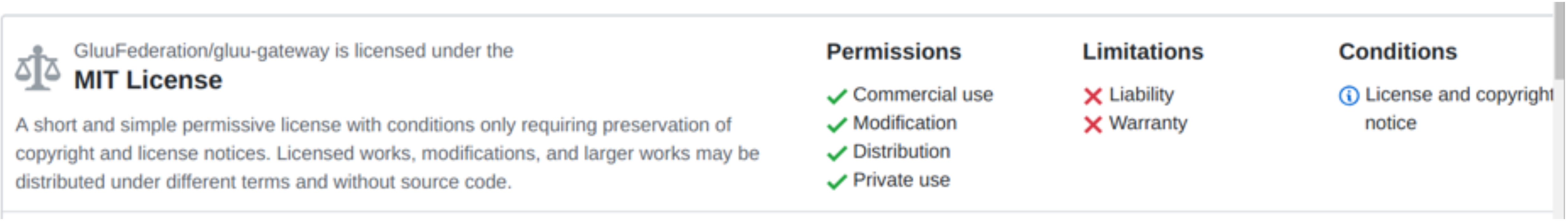
```
def authorize(self, context):
    print "RPT Policy. Authorizing ..."

    if context.getClaim("country") == 'US':
        print "Authorized successfully!"
        return True

    # Look at client claims / request claims / HEADERS
    # Call API's

    return False
```


Resource Owner
Manage
(out-of-band)
Control
(out-of-band)
Resource Server
PAT
Permission Ticket
Authorization Server
RPT
PCT
RPT
Authenticate,
Gather Claims
(optional)
Client
Use
(optional)
Requesting Party


Resource Owner
API
API
API
Manage with
Gluu Gateway
Admin Web UI
Manage with
oxTrust
Admin Web UI
gluu gateway
Resource Server
PAT bearer token
gluu Server
Authorization Server
Protection
API
token
returned
/uma_authz (PCT)
/oauth_token (PAT, oxd)
/uma_token (RPT)
RPT bearer token
Requesting Party
UMA Client
oxd
bearer
token
oxd
API
no token
client creds
oxd server
oxd bearer token
optional

Demo Code

https://gluu.co/gg-demo



GluuFederation/gluu-gateway is licensed under the

MIT License

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- ✓ Commercial use
- ✓ Modification
- ✓ Distribution
- ✓ Private use

Limitations

- ✗ Liability
- ✗ Warranty

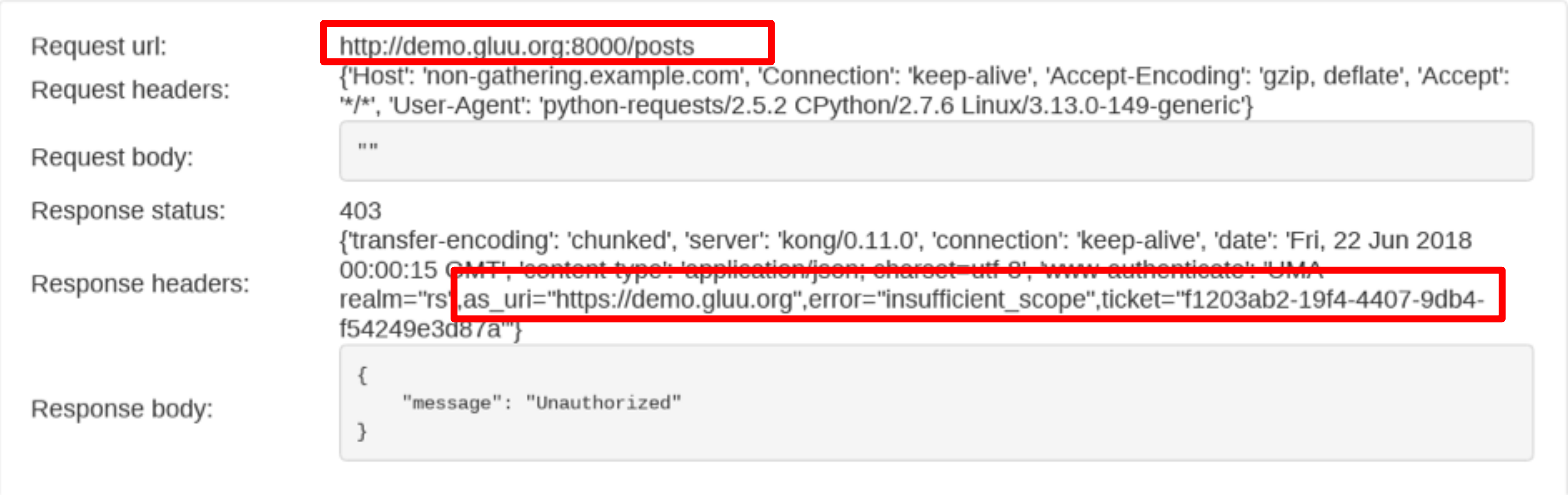
Conditions

- ⓘ License and copyright notice

1. Client calls API with no RPT token

Kong returns as_uri, permission ticket

443 = oxAuth
8000 = kong
8443 = oxd
8080 = client demo



Request url: http://demo.gluu.org:8000/posts

Request headers: {'Host': 'non-gathering.example.com', 'Connection': 'keep-alive', 'Accept-Encoding': 'gzip, deflate', 'Accept': '*/*', 'User-Agent': 'python-requests/2.5.2 CPython/2.7.6 Linux/3.13.0-149-generic'}

Request body:

```
""
```

Response status: 403

Response headers: {'transfer-encoding': 'chunked', 'server': 'kong/0.11.0', 'connection': 'keep-alive', 'date': 'Fri, 22 Jun 2018 00:00:15 GMT', 'content-type': 'application/json; charset=utf-8', 'www-authenticate': 'UMA realm="rs",as_uri="https://demo.gluu.org",error="insufficient_scope",ticket="f1203ab2-19f4-4407-9db4-f54249e3d87a"'}

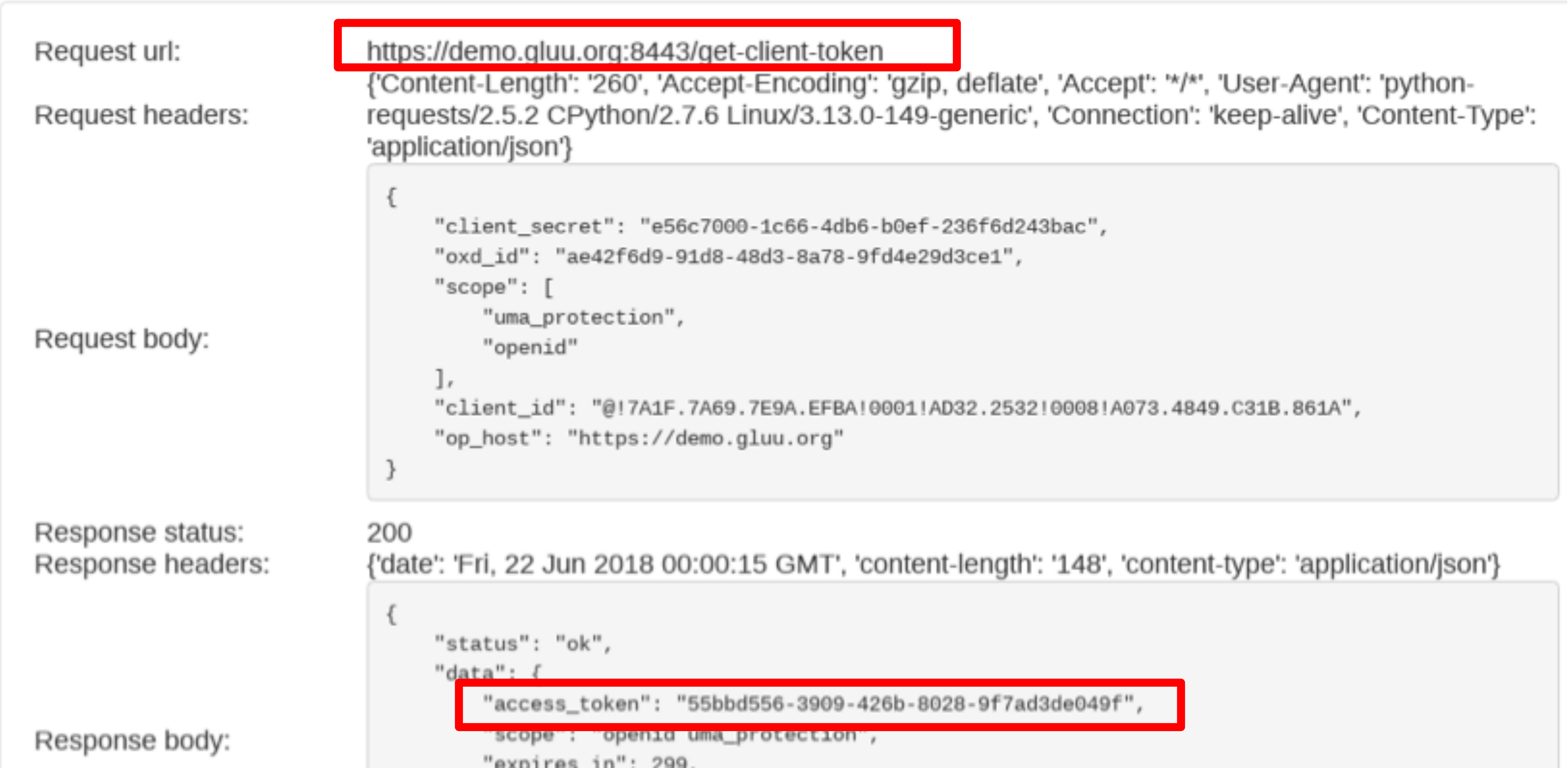
Response body:

```
{
    "message": "Unauthorized"
}
```

2. Client obtains oxd token

Needed to call protected oxd endpoints

443 = oxAuth
8000 = kong
8443 = oxd
8080 = client demo



Request url: https://demo.gluu.org:8443/get-client-token

Request headers: {'Content-Length': '260', 'Accept-Encoding': 'gzip, deflate', 'Accept': '*/*', 'User-Agent': 'python-requests/2.5.2 CPython/2.7.6 Linux/3.13.0-149-generic', 'Connection': 'keep-alive', 'Content-Type': 'application/json'}

Request body:

```
{
    "client_secret": "e56c7000-1c66-4db6-b0ef-236f6d243bac",
    "oxd_id": "ae42f6d9-91d8-48d3-8a78-9fd4e29d3ce1",
    "scope": [
        "uma_protection",
        "openid"
    ],
    "client_id": "@!7A1F.7A69.7E9A.EFBA!0001!AD32.2532!0008!A073.4849.C31B.861A",
    "op_host": "https://demo.gluu.org"
}
```

Response status: 200

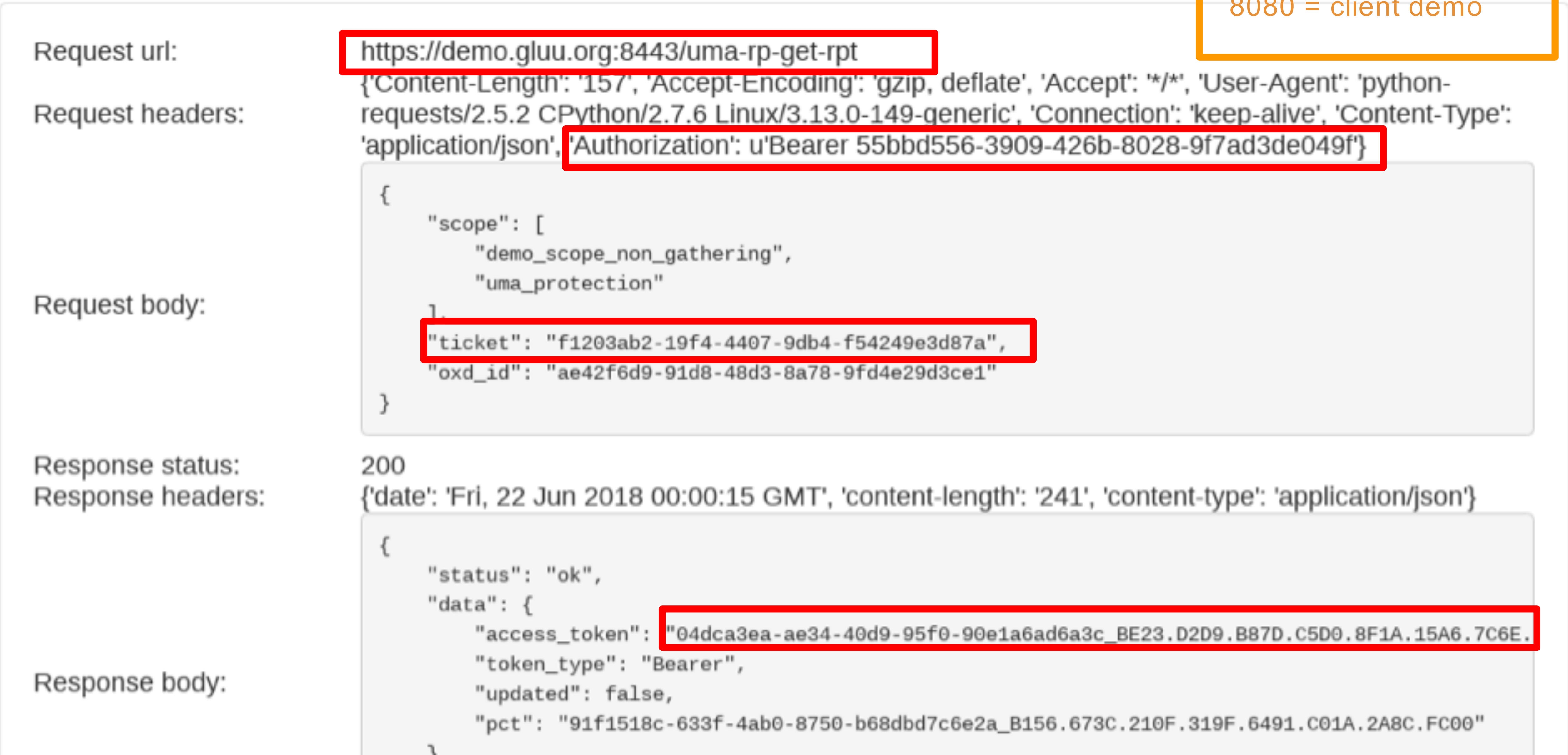
Response headers: {'date': 'Fri, 22 Jun 2018 00:00:15 GMT', 'content-length': '148', 'content-type': 'application/json'}

Response body:

```
{
    "status": "ok",
    "data": {
        "access_token": "55bbd556-3909-426b-8028-9f7ad3de049f",
        "scope": "openid uma_protection",
        "expires_in": 299,
```

3. Client calls /uma_token to get RPT

443 = oxAuth
8000 = kong
8443 = oxd
8080 = client demo



Request url: https://demo.gluu.org:8443/uma-rp-get-rpt

Request headers: {'Content-Length': '157', 'Accept-Encoding': 'gzip, deflate', 'Accept': '*/*', 'User-Agent': 'python-requests/2.5.2 CPython/2.7.6 Linux/3.13.0-149-generic', 'Connection': 'keep-alive', 'Content-Type': 'application/json', 'Authorization': u'Bearer 55bbd556-3909-426b-8028-9f7ad3de049f'}

Request body:

```
{
    "scope": [
        "demo_scope_non_gathering",
        "uma_protection"
    ],
    "ticket": "f1203ab2-19f4-4407-9db4-f54249e3d87a",
    "oxd_id": "ae42f6d9-91d8-48d3-8a78-9fd4e29d3ce1"
}
```

Response status: 200

Response headers: {'date': 'Fri, 22 Jun 2018 00:00:15 GMT', 'content-length': '241', 'content-type': 'application/json'}

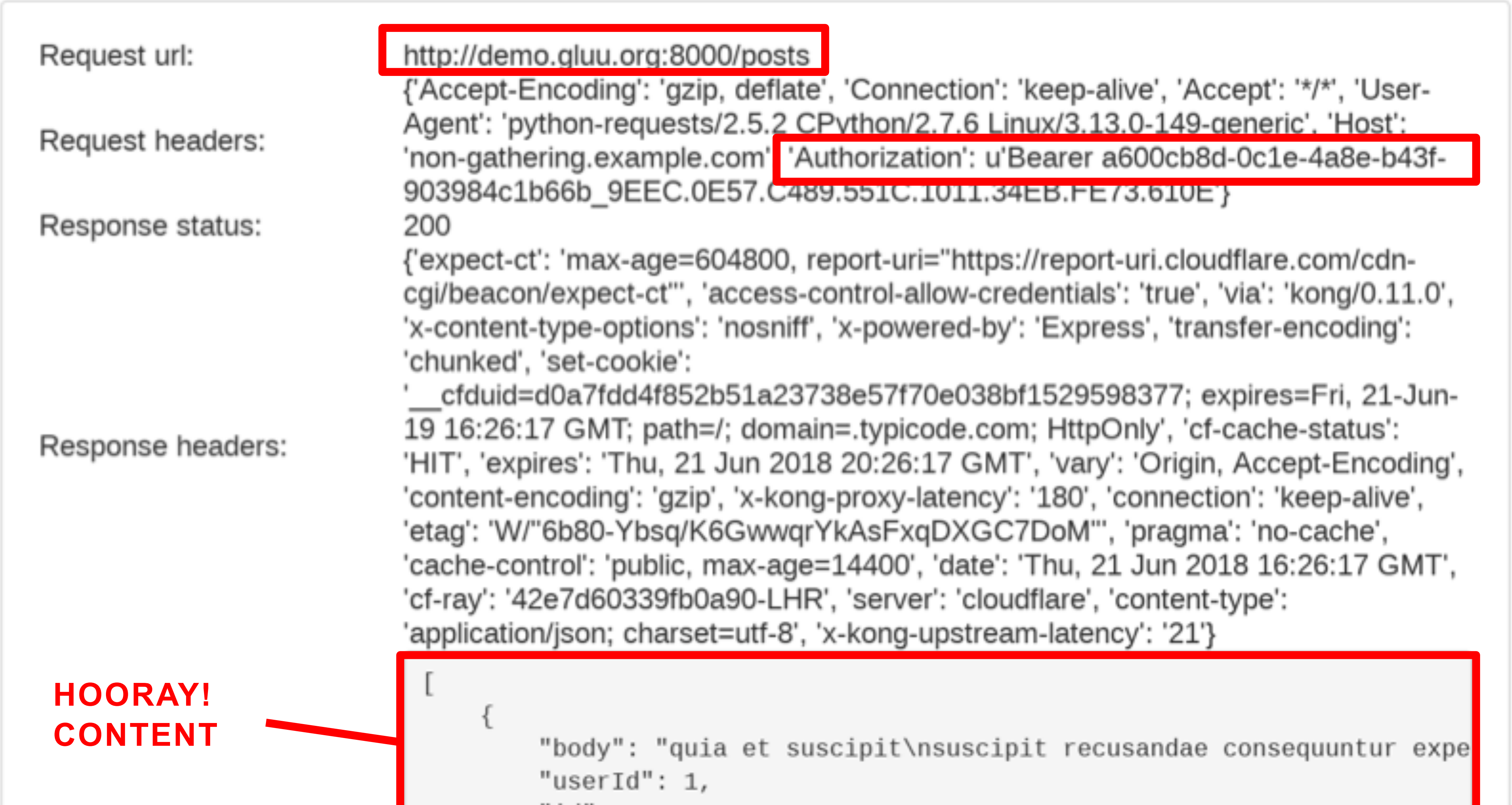
Response body:

```
{
    "status": "ok",
    "data": {
        "access_token": "04dca3ea-ae34-40d9-95f0-90e1a6ad6a3c_BE23.D2D9.B87D.C5D0.8F1A.15A6.7C6E.
        "token_type": "Bearer",
        "updated": false,
        "pct": "91f1518c-633f-4ab0-8750-b68dbd7c6e2a_B156.673C.210F.319F.6491.C01A.2A8C.FC00"
    }
```

4. Client calls API Gateway with RPT

443 = oxAuth
8000 = kong
8443 = oxd
8080 = client demo

Gluu Gateway returns permission ticket, as_uri



Request url: http://demo.gluu.org:8000/posts

Request headers: {'Accept-Encoding': 'gzip, deflate', 'Connection': 'keep-alive', 'Accept': '*/*', 'User-Agent': 'python-requests/2.5.2 CPython/2.7.6 Linux/3.13.0-149-generic', 'Host': 'non-gathering.example.com', 'Authorization': u'Bearer a600cb8d-0c1e-4a8e-b43f-903984c1b66b_9EEC.0E57.C489.551C.1011.34EB.FE73.610E'}

Response status: 200

Response headers: {'expect-ct': 'max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"', 'access-control-allow-credentials': 'true', 'via': 'kong/0.11.0', 'x-content-type-options': 'nosniff', 'x-powered-by': 'Express', 'transfer-encoding': 'chunked', 'set-cookie': '__cfduid=d0a7fdd4f852b51a23738e57f70e038bf1529598377; expires=Fri, 21-Jun-19 16:26:17 GMT; path=/; domain=.typicode.com; HttpOnly', 'cf-cache-status': 'HIT', 'expires': 'Thu, 21 Jun 2018 20:26:17 GMT', 'vary': 'Origin, Accept-Encoding', 'content-encoding': 'gzip', 'x-kong-proxy-latency': '180', 'connection': 'keep-alive', 'etag': 'W/"6b80-Ybsq/K6GwwqrYkAsFxqDXGC7DoM"', 'pragma': 'no-cache', 'cache-control': 'public, max-age=14400', 'date': 'Thu, 21 Jun 2018 16:26:17 GMT', 'cf-ray': '42e7d60339fb0a90-LHR', 'server': 'cloudflare', 'content-type': 'application/json; charset=utf-8', 'x-kong-upstream-latency': '21'}

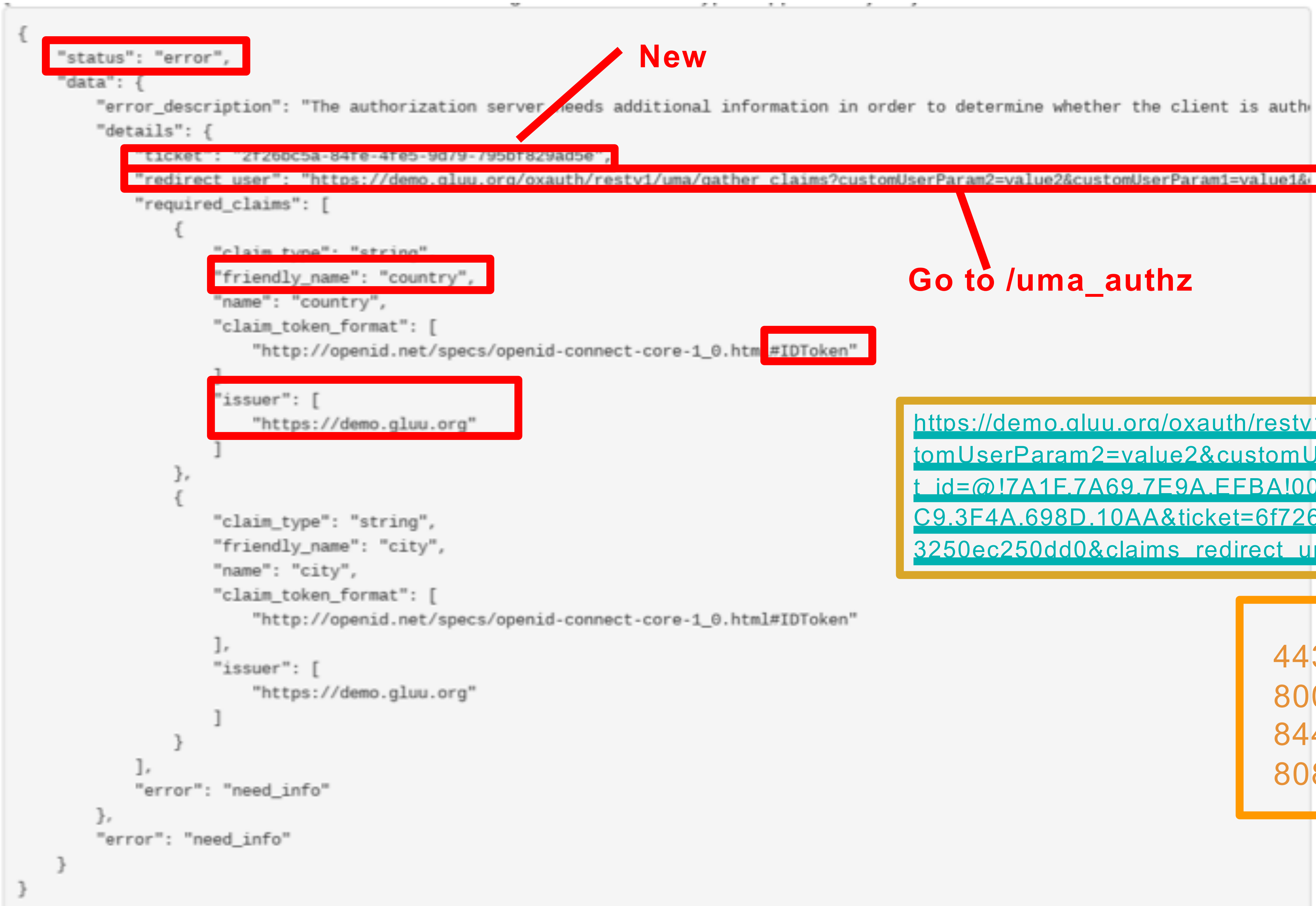
```
[
    {
        "body": "quia et suscipit\nsuscipit recusandae consequuntur expe
        "userId": 1,
```

HOORAY!
CONTENT

Claims gathering

What if one or more of the policies evaluate to False?

No RPT for you! Go directly to Claims Gathering!



```
{
  "status": "error",
  "data": {
    "error_description": "The authorization server needs additional information in order to determine whether the client is auth
    "details": {
      "ticket": "2f26bc5a-84fe-4fe5-9d79-795bf829ad5e",
      "redirect_user": "https://demo.gluu.org/oxauth/restv1/uma/gather_claims?customUserParam2=value2&customUserParam1=value1&
      "required_claims": [
        {
          "claim_type": "string",
          "friendly_name": "country",
          "name": "country",
          "claim_token_format": [
            "http://openid.net/specs/openid-connect-core-1_0.html#IDToken"
          ],
          "issuer": [
            "https://demo.gluu.org"
          ]
        },
        {
          "claim_type": "string",
          "friendly_name": "city",
          "name": "city",
          "claim_token_format": [
            "http://openid.net/specs/openid-connect-core-1_0.html#IDToken"
          ],
          "issuer": [
            "https://demo.gluu.org"
          ]
        }
      ],
      "error": "need_info"
    },
    "error": "need_info"
  }
}
```

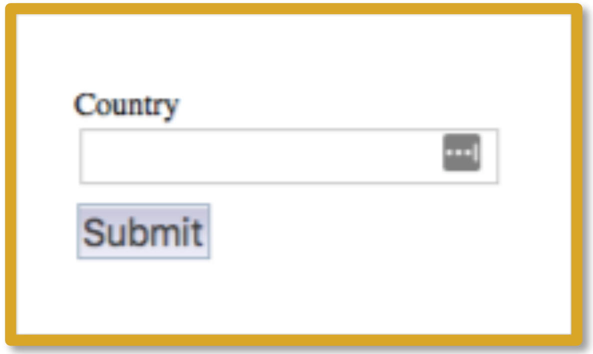
New

Go to /uma_authz

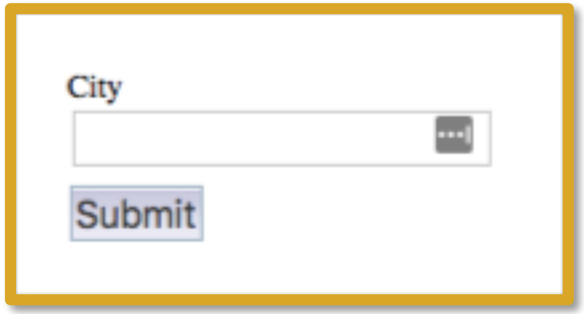
https://demo.gluu.org/oxauth/restv1/uma/gather_claims?customUserParam2=value2&customUserParam1=value1&client_id=@!7A1F.7A69.7E9A.EFBA!0001!AD32.2532!0008!EDC9.3F4A.698D.10AA&ticket=6f726d47-1891-4275-8f23-3250ec250dd0&claims_redirect_uri=http://demo.gluu.

443	= oxAuth
8000	= kong
8443	= oxd
8080	= client demo

STEP 1



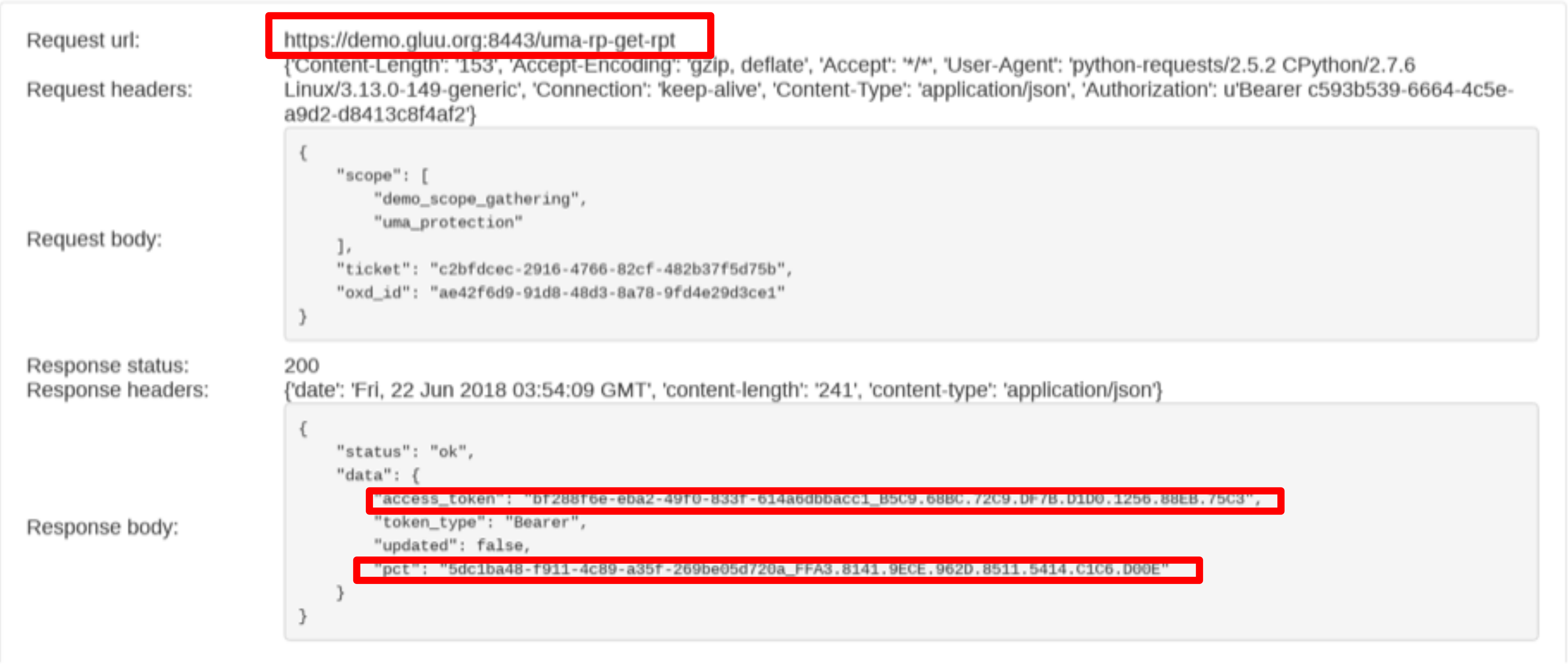
STEP 2



LIVE DEMO! (ish)

Requesting party is redirected to the AS for a multi-step consent workflow.

Claims gathering done! Here’s a PCT for next time!



Request url: https://demo.gluu.org:8443/uma-rp-get-rpt

Request headers: {'Content-Length': '153', 'Accept-Encoding': 'gzip, deflate', 'Accept': '*/*', 'User-Agent': 'python-requests/2.5.2 CPython/2.7.6 Linux/3.13.0-149-generic', 'Connection': 'keep-alive', 'Content-Type': 'application/json', 'Authorization': u'Bearer c593b539-6664-4c5e-a9d2-d8413c8f4af2'}

Request body:

```
{
    "scope": [
        "demo_scope_gathering",
        "uma_protection"
    ],
    "ticket": "c2bfdcec-2916-4766-82cf-482b37f5d75b",
    "oxd_id": "ae42f6d9-91d8-48d3-8a78-9fd4e29d3ce1"
}
```

Response status: 200

Response headers: {'date': 'Fri, 22 Jun 2018 03:54:09 GMT', 'content-length': '241', 'content-type': 'application/json'}

Response body:

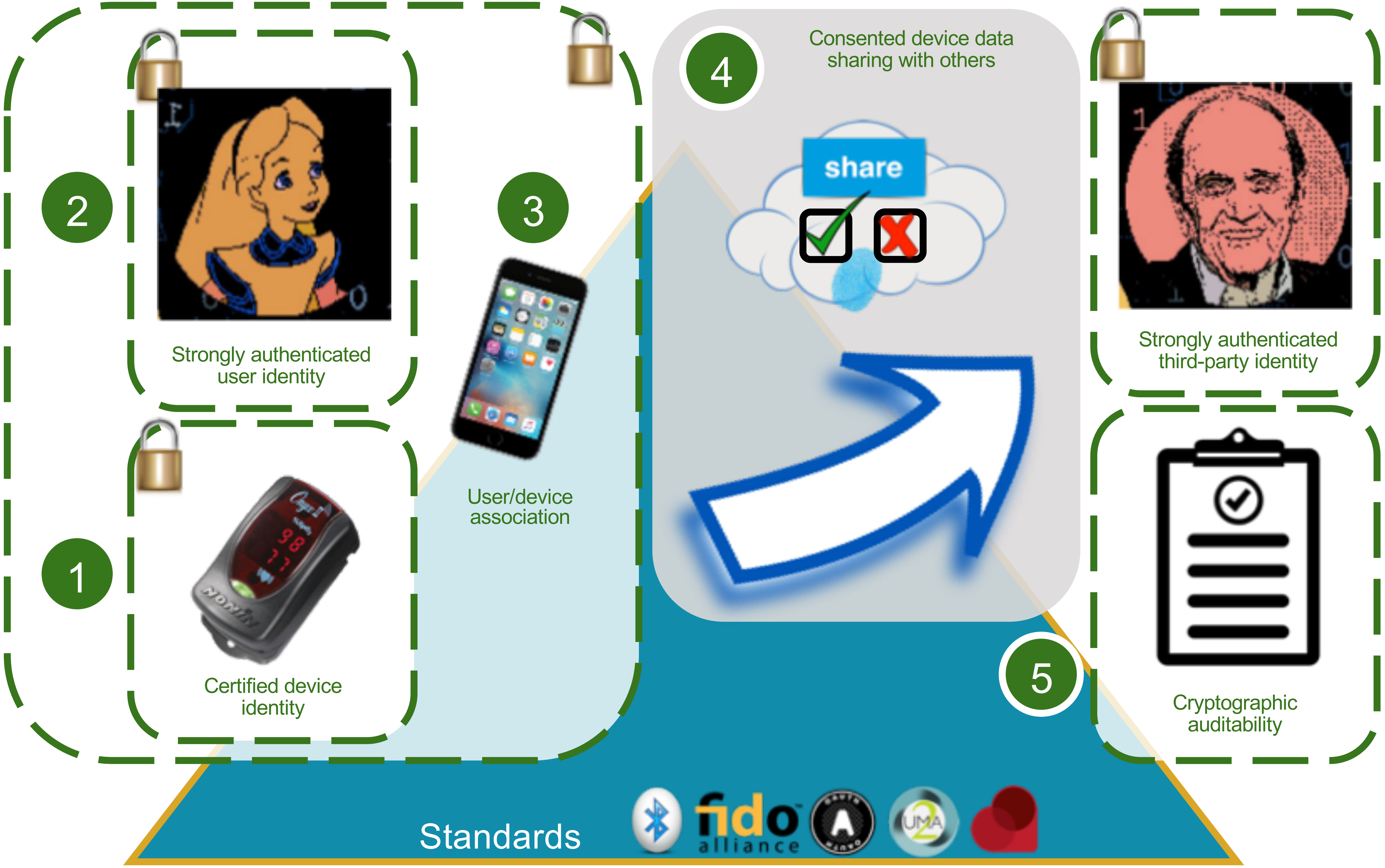
```
{
    "status": "ok",
    "data": {
        "access_token": "bf288f6e-eba2-49f0-833f-614a6dbbacc1_B5C9.68BC.72C9.DF7B.D1D0.1256.88EB.75C3",
        "token_type": "Bearer",
        "updated": false,
        "pct": "5dc1ba48-f911-4c89-a35f-269be05d720a_FFA3.8141.9ECE.962D.8511.5414.C1C6.D00E"
    }
}
```

Walkthrough by Eve:

Sharing pulse oximeter data in a trusted and consented way with third parties through loosely coupled cloud services

OMR
Open Med Ready

NONIN
Pulse Oximeter
Bluetooth® Smart / Nonin 3231 USB
98 %
82
97 %
87
NONIN
Bluetooth® Smart Wireless Technology
USB Connection
NONIN PureSAT SpO2



Certified device identity

Strongly authenticated user identity

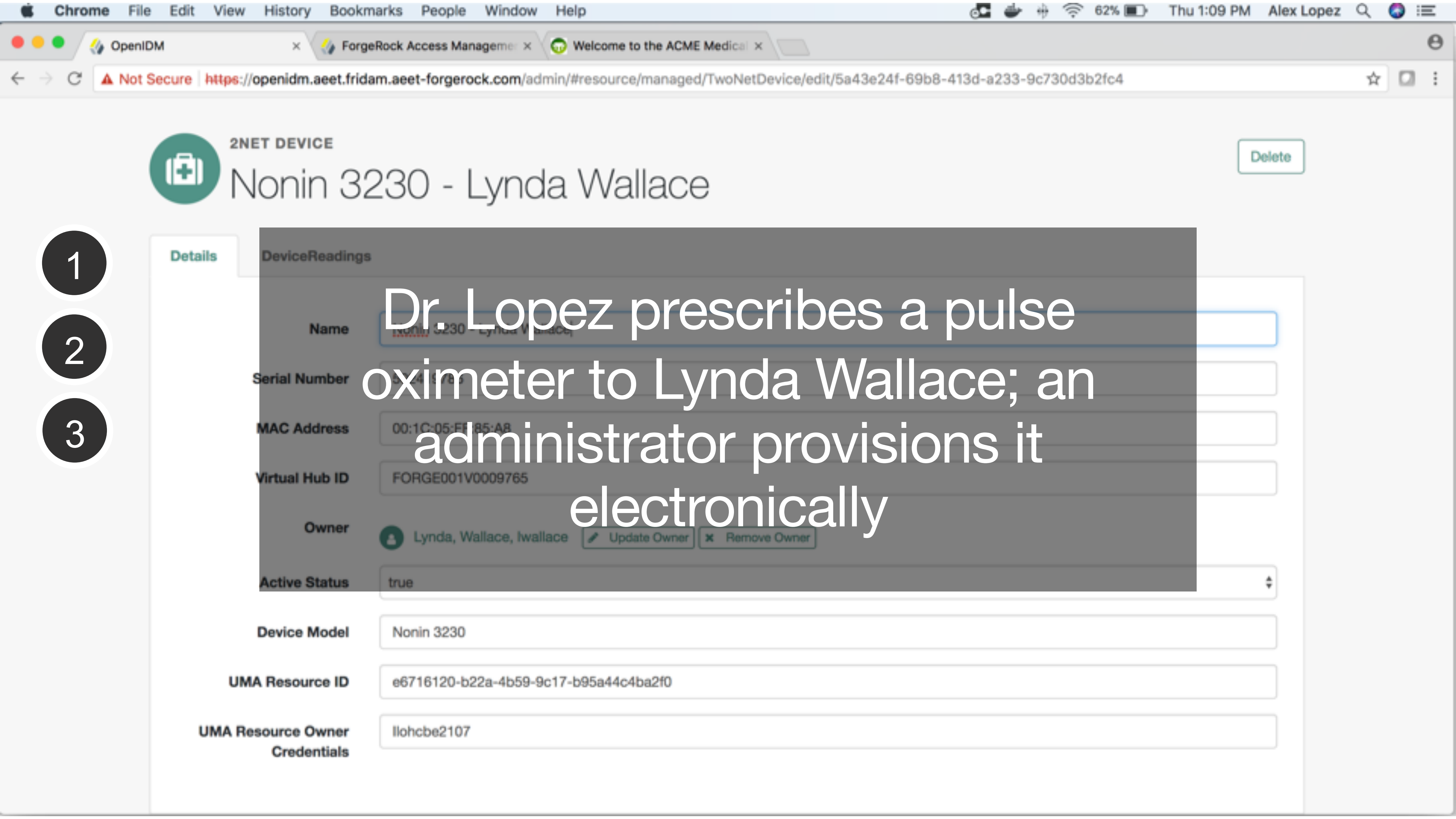
User/device association

Consented device data sharing with others

Strongly authenticated third-party identity

Cryptographic auditability

Standards


Chrome File Edit View History Bookmarks People Window Help
62% Thu 1:09 PM Alex Lopez
OpenIDM
ForgeRock Access Manageme
Welcome to the ACME Medical
Not Secure https://openidm.aeet.fridam.aeet-forgerock.com/admin/#resource/managed/TwoNetDevice/edit/5a43e24f-69b8-413d-a233-9c730d3b2fc4
Delete
Details
DeviceReadings
Update Owner
Remove Owner

1

2

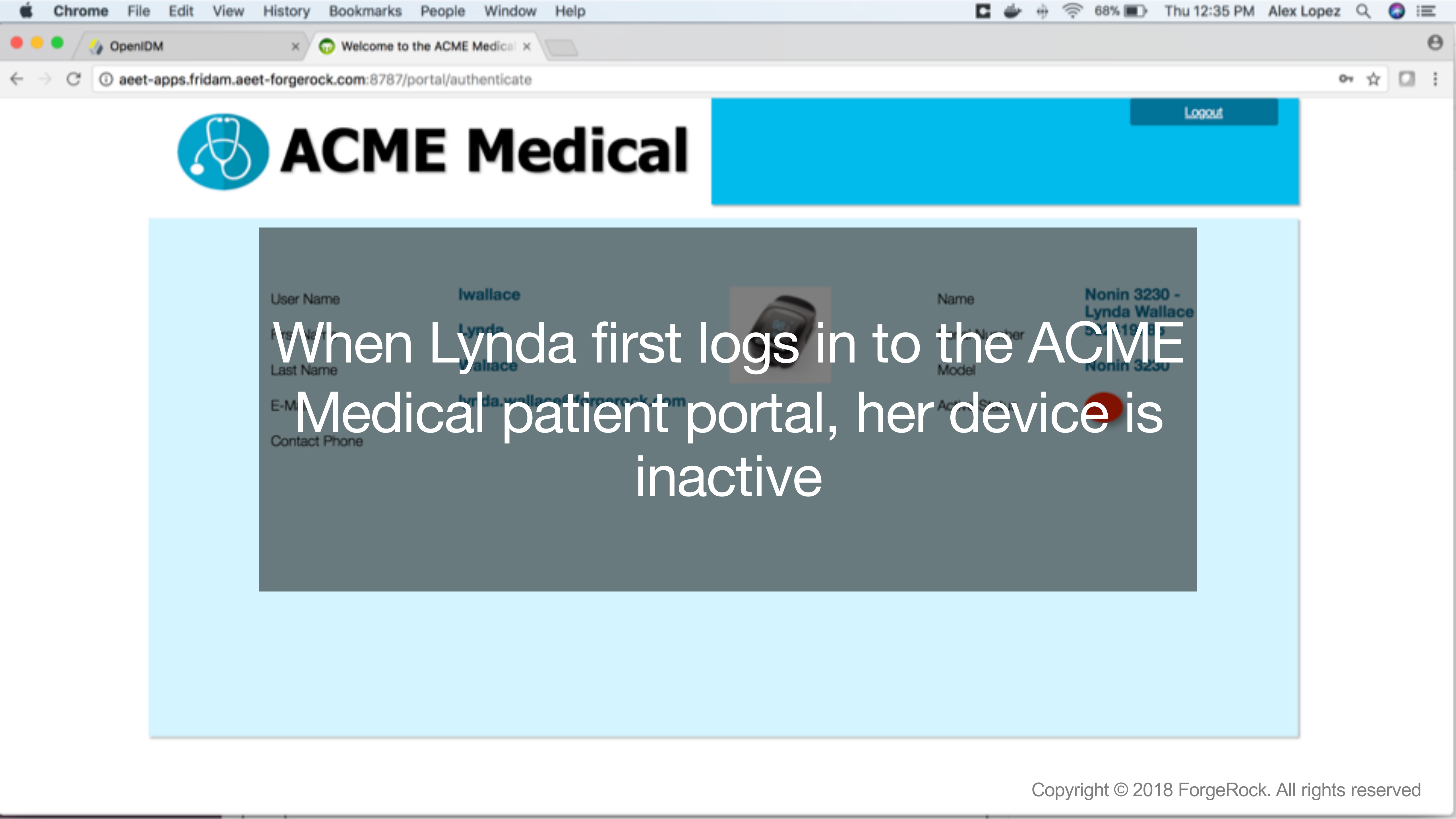
3

Dr. Lopez prescribes a pulse oximeter to Lynda Wallace; an administrator provisions it electronically

2NET DEVICE

Nonin 3230 - Lynda Wallace

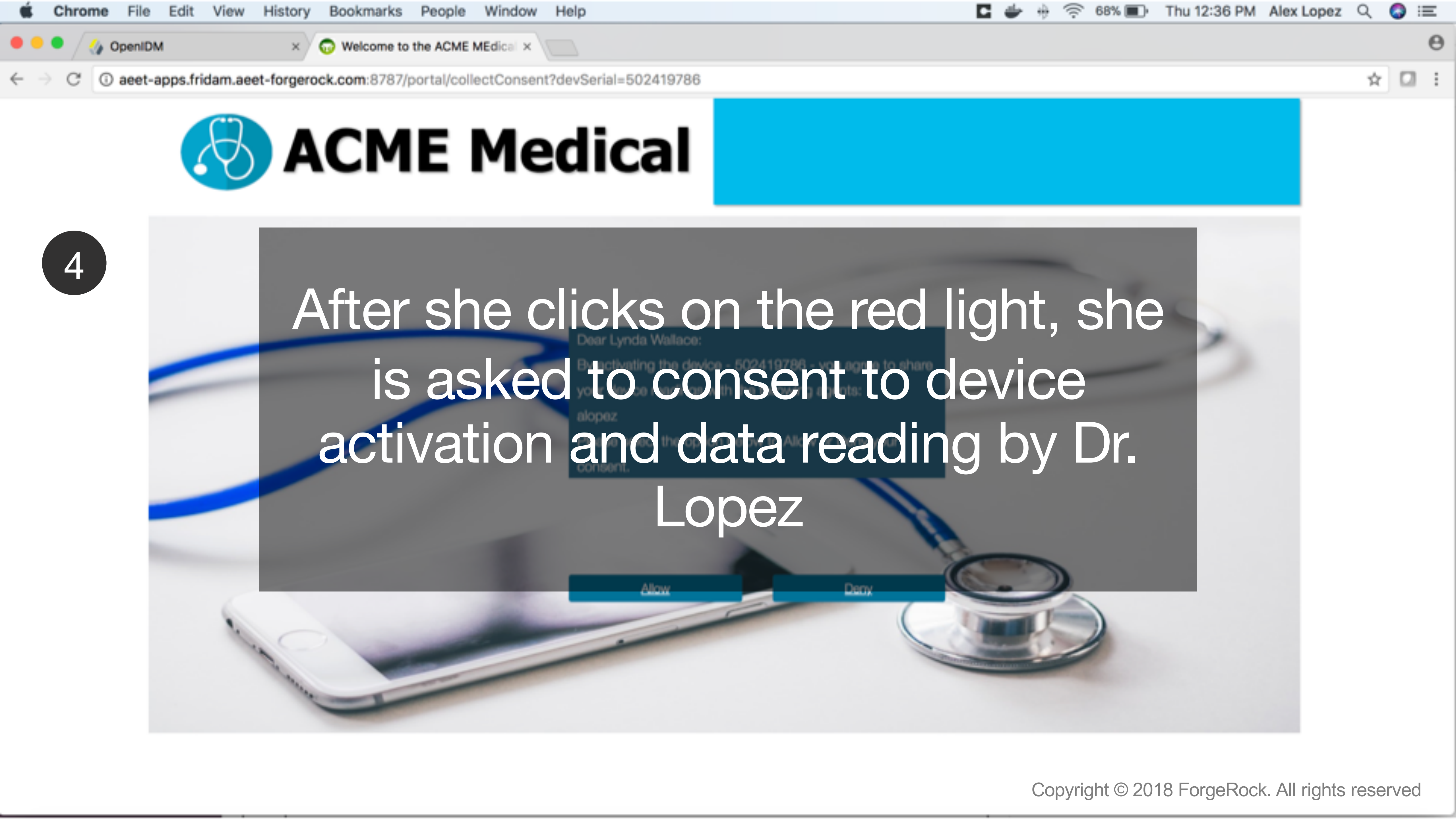
Name	Nonin 3230 - Lynda Wallace
Serial Number	502419785
MAC Address	00:1C:05:FF:85:A8
Virtual Hub ID	FORGE001V0009765
Owner	Lynda, Wallace, lwallace
Active Status	true
Device Model	Nonin 3230
UMA Resource ID	e6716120-b22a-4b59-9c17-b95a44c4ba2f0
UMA Resource Owner Credentials	llohcbe2107



Chrome File Edit View History Bookmarks People Window Help
Thu 12:35 PM Alex Lopez
OpenIDM
Welcome to the ACME Medical
aeet-apps.fridam.aeet-forgerock.com:8787/portal/authenticate
ACME Medical
Logout

User Name	lwallace	Name	Nonin 3230 - Lynda Wallace
First Name	Lynda	Serial Number	[illegible]
Last Name	Wallace	Model	Nonin 3230
E-Mail	lynda.wallace@forgerock.com	Activation Status	●
Contact Phone			

When Lynda first logs in to the ACME Medical patient portal, her device is inactive


Chrome File Edit View History Bookmarks People Window Help
68% Thu 12:36 PM Alex Lopez
OpenIDM
Welcome to the ACME MEdical
aeet-apps.fridam.aeet-forgerock.com:8787/portal/collectConsent?devSerial=502419786
ACME Medical

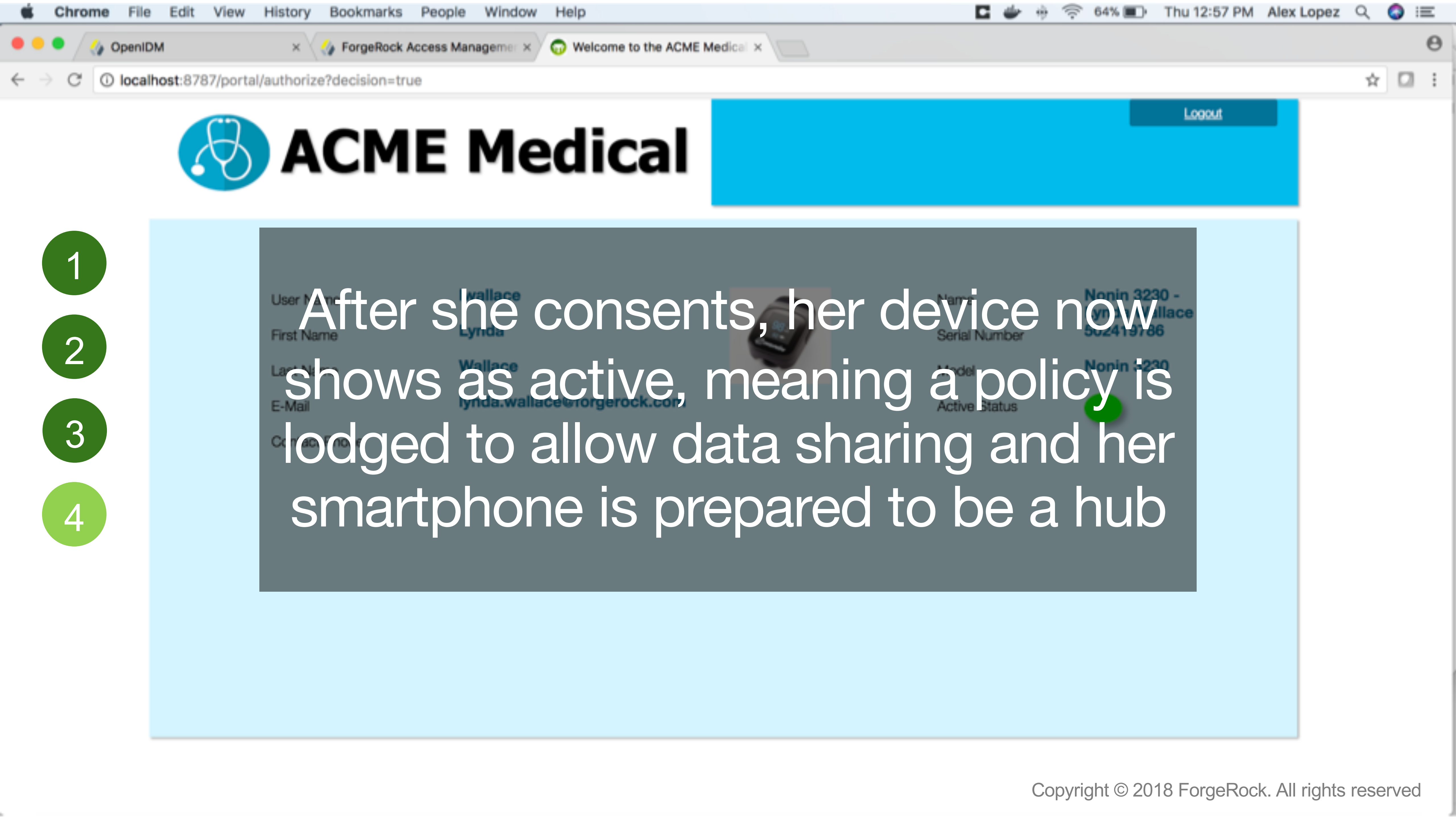
Dear Lynda Wallace:

By activating the device - 502419786 - you agree to share your [illegible] alopez [illegible] consent.

Allow Deny

4

After she clicks on the red light, she is asked to consent to device activation and data reading by Dr. Lopez


Chrome File Edit View History Bookmarks People Window Help
64% Thu 12:57 PM Alex Lopez
OpenIDM
ForgeRock Access Manageme
Welcome to the ACME Medica
localhost:8787/portal/authorize?decision=true
ACME Medical
Logout
1
2
3
4

User Name: lwallace

First Name: Lynda

Last Name: Wallace

E-Mail: lynda.wallace@forgerock.com

Contact no

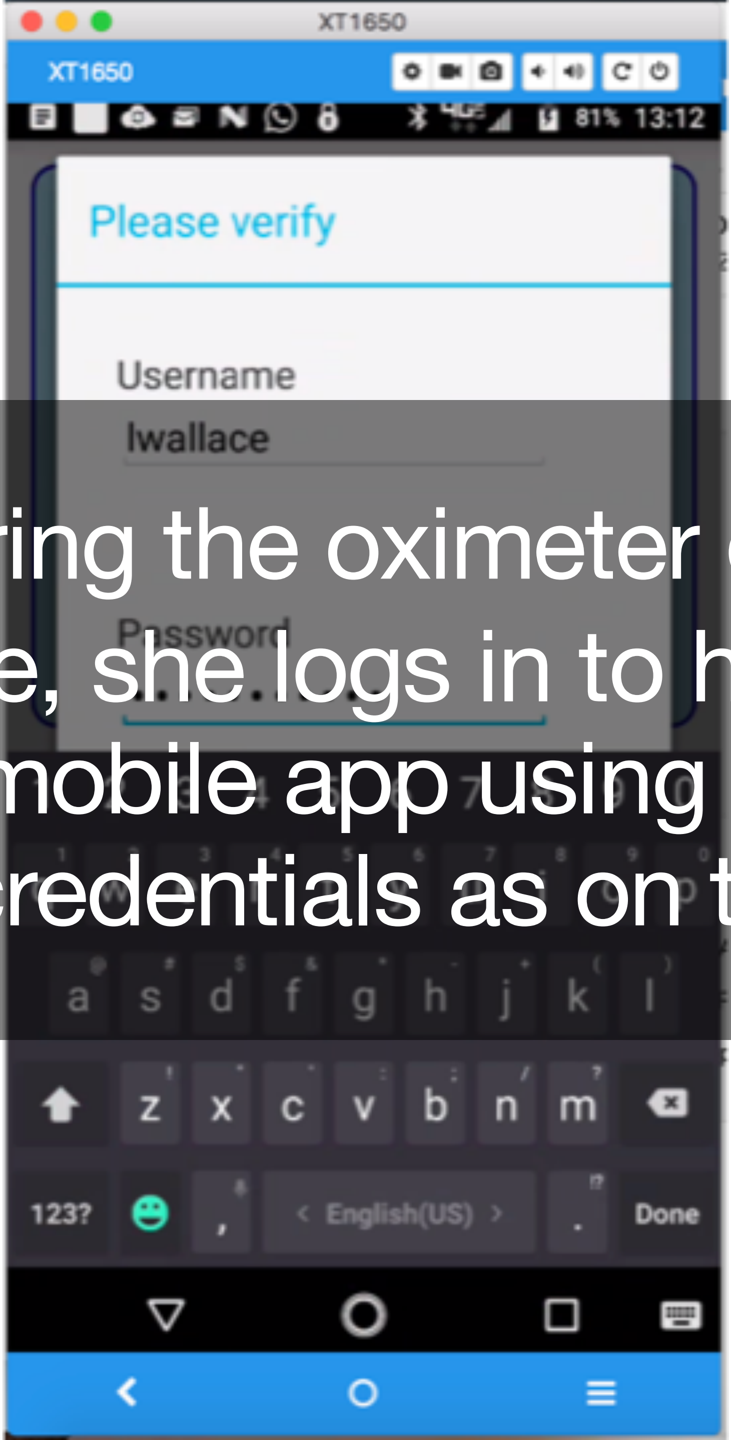
Name: Nonin 3230 - Lynda Wallace

Serial Number: 502419786

Model: Nonin 3230

Active Status: ●

After she consents, her device now shows as active, meaning a policy is lodged to allow data sharing and her smartphone is prepared to be a hub



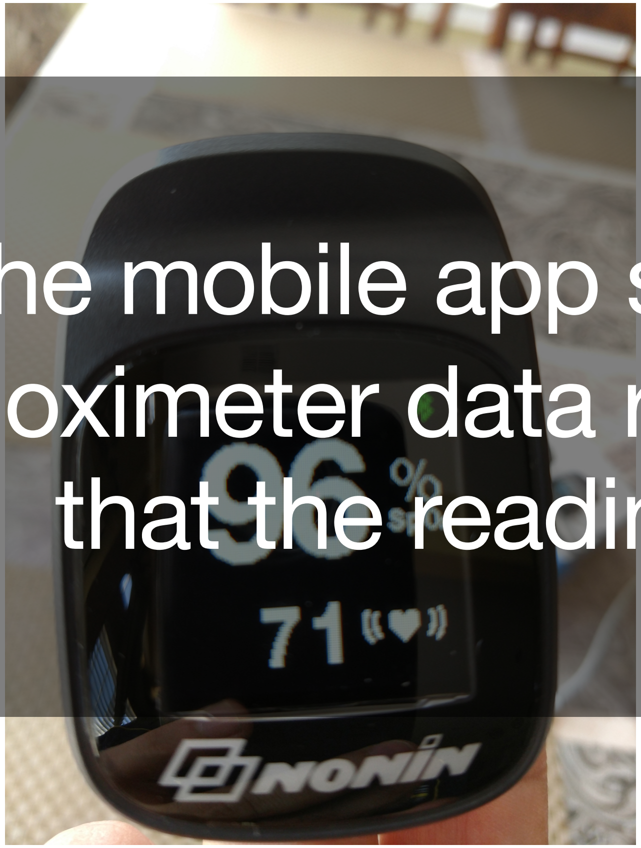
Please verify

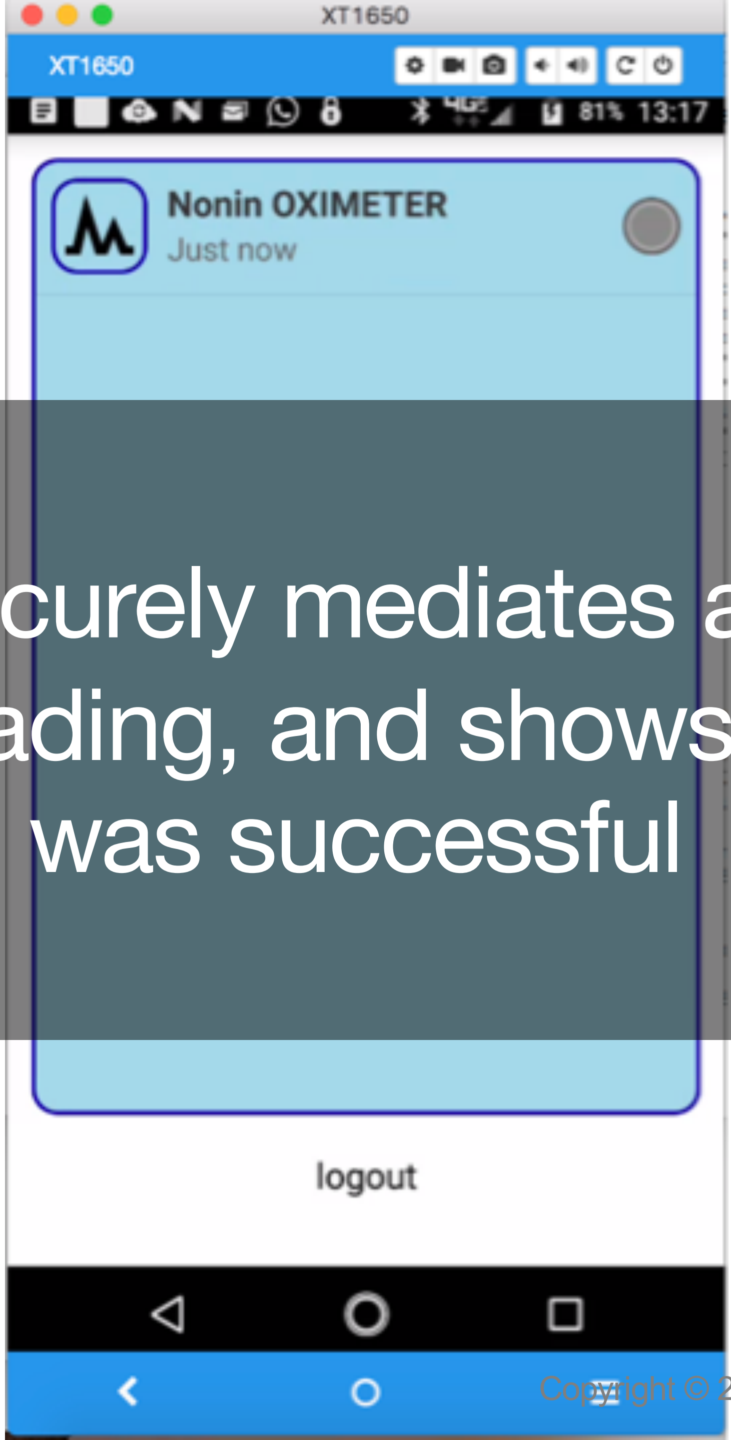
Username

lwallace

Password

After pairing the oximeter device to her phone, she logs in to her ACME Medical mobile app using the same identity credentials as on the portal




XT1650

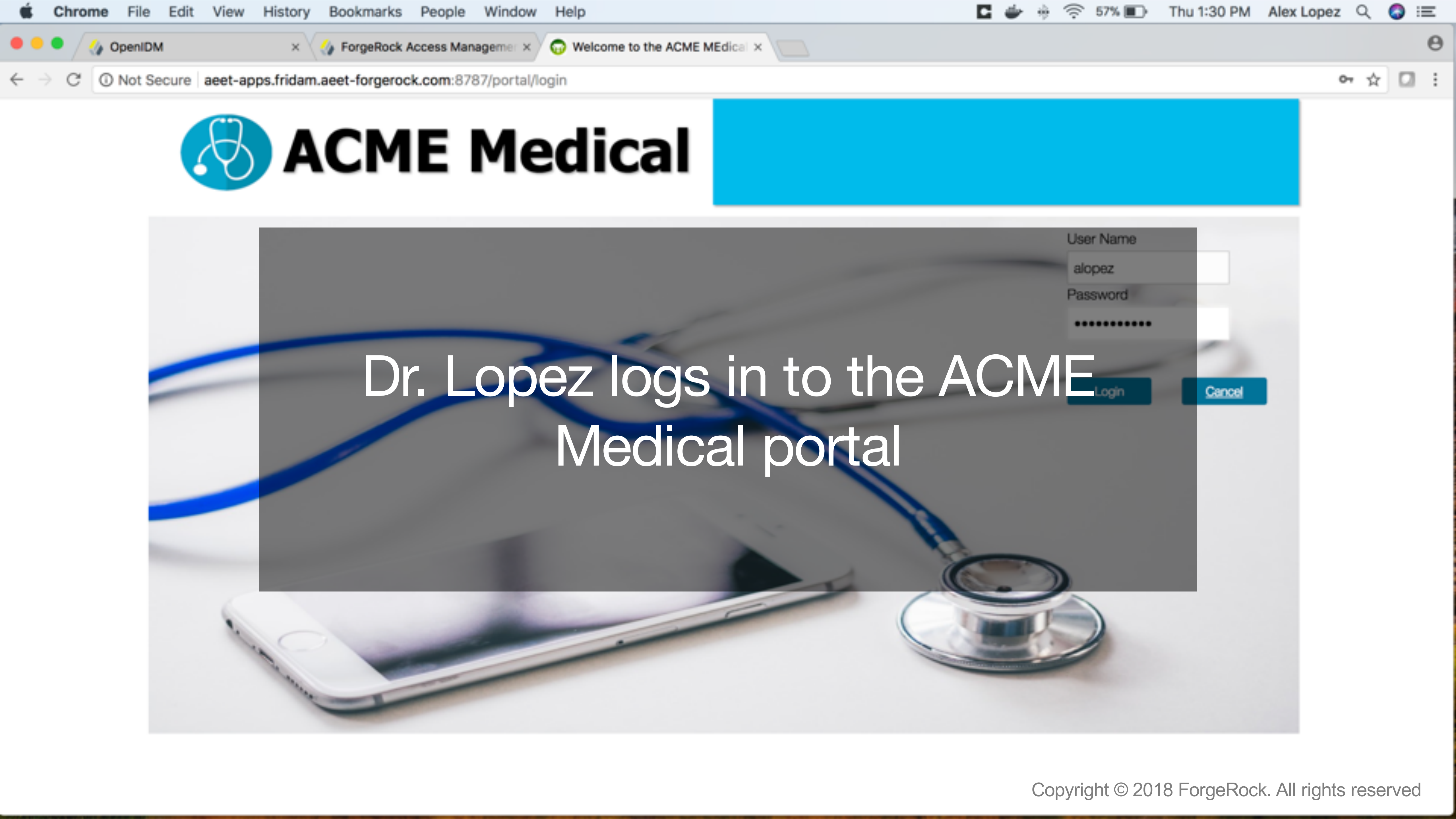
XT1650

81% 13:17

Nonin OXIMETER
Just now

logout

The mobile app securely mediates an oximeter data reading, and shows that the reading was successful


Chrome File Edit View History Bookmarks People Window Help
57%
Thu 1:30 PM
Alex Lopez
OpenIDM
ForgeRock Access Manageme
Welcome to the ACME MEdical
Not Secure
aeet-apps.fridam.aeet-forgerock.com:8787/portal/login
ACME Medical
Login
Cancel

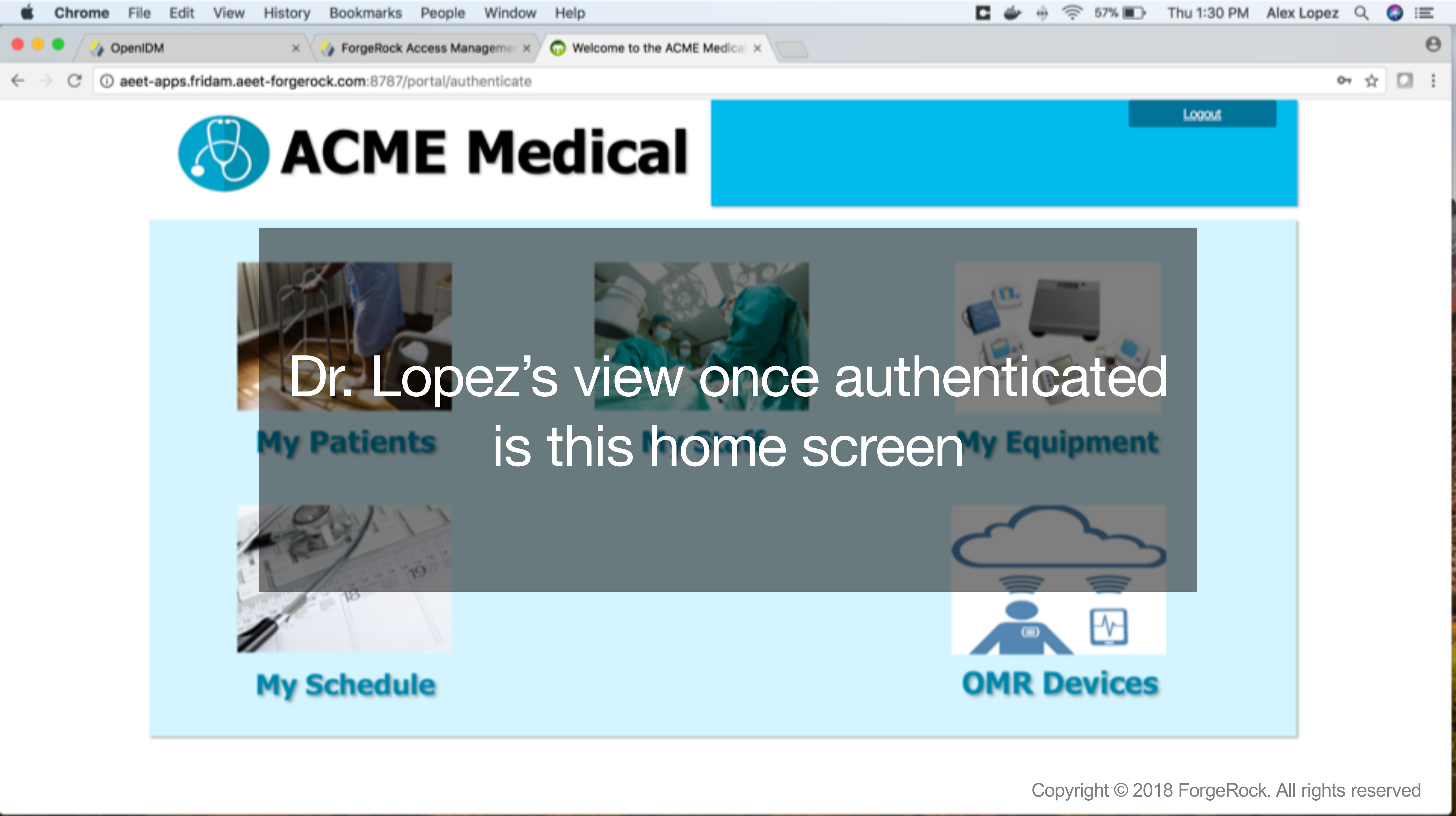
User Name

alopez

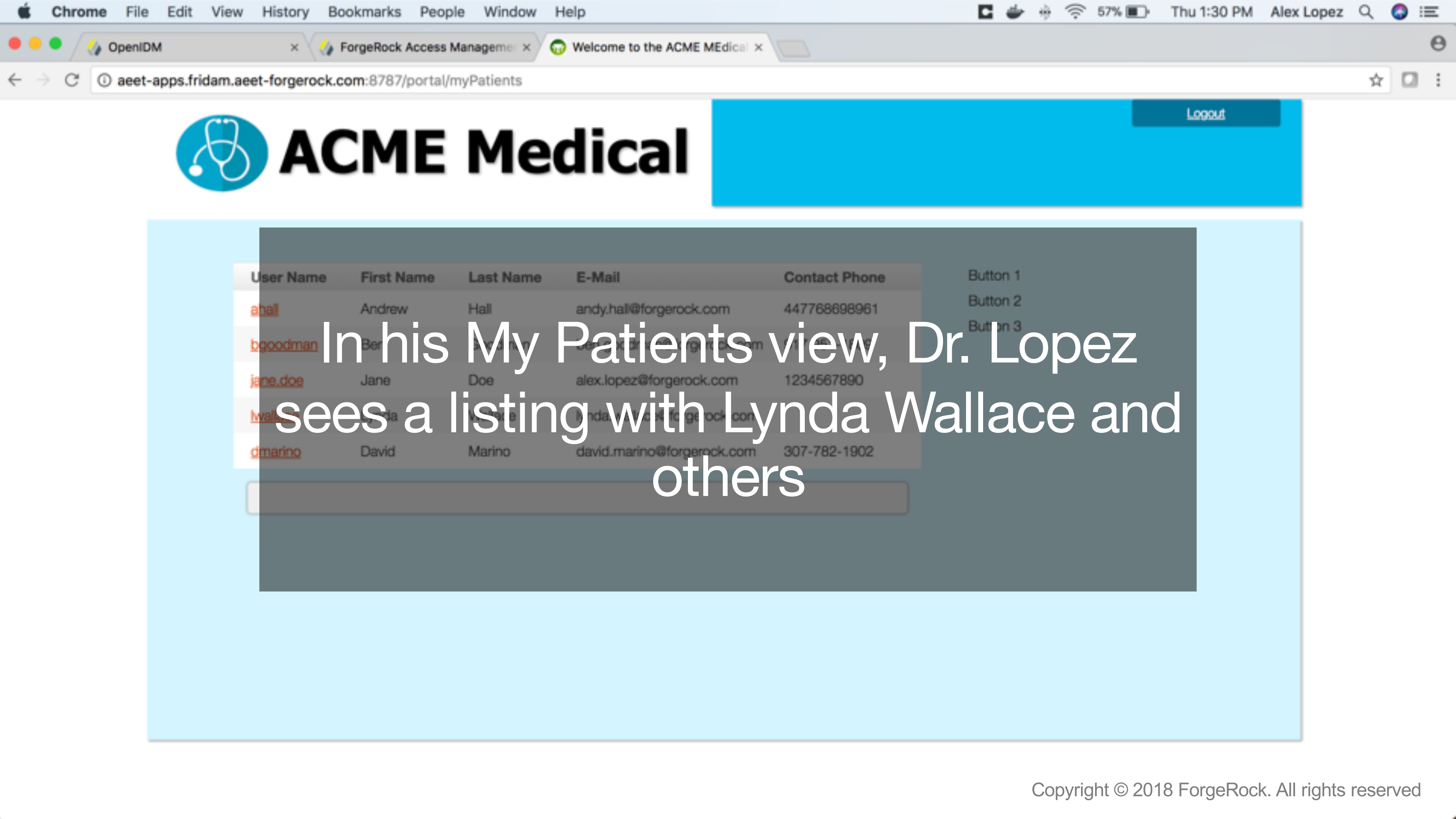
Password

••••••••••••

Dr. Lopez logs in to the ACME Medical portal

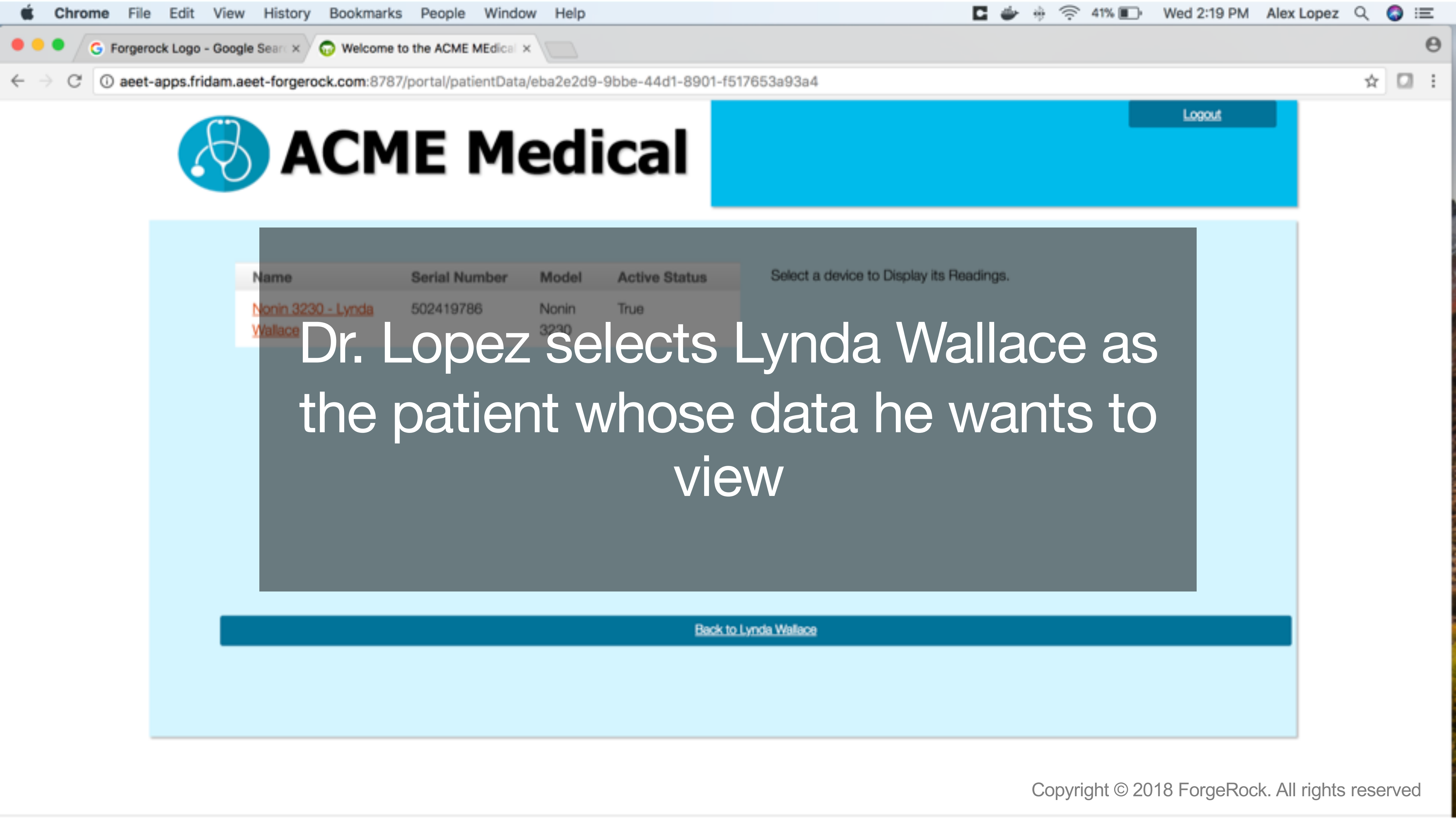

Chrome File Edit View History Bookmarks People Window Help
Thu 1:30 PM Alex Lopez
OpenIDM
ForgeRock Access Manageme
Welcome to the ACME Medical
aeet-apps.fridam.aeet-forgerock.com:8787/portal/authenticate
ACME Medical
Logout
My Patients
My Staff
My Equipment
My Schedule
OMR Devices

Dr. Lopez's view once authenticated is this home screen


Chrome File Edit View History Bookmarks People Window Help
Thu 1:30 PM Alex Lopez
OpenIDM
ForgeRock Access Manageme
Welcome to the ACME MEdical
aeet-apps.fridam.aeet-forgerock.com:8787/portal/myPatients
ACME Medical
Logout
Button 1
Button 2
Button 3

User Name	First Name	Last Name	E-Mail	Contact Phone
ahall	Andrew	Hall	andy.hall@forgerock.com	447768698961
bgoodman	[illegible]	[illegible]	[illegible]	[illegible]
jane.doe	Jane	Doe	alex.lopez@forgerock.com	1234567890
lwallace	Lynda	Wallace	lynda.wallace@forgerock.com	[illegible]
dmarino	David	Marino	david.marino@forgerock.com	307-782-1902

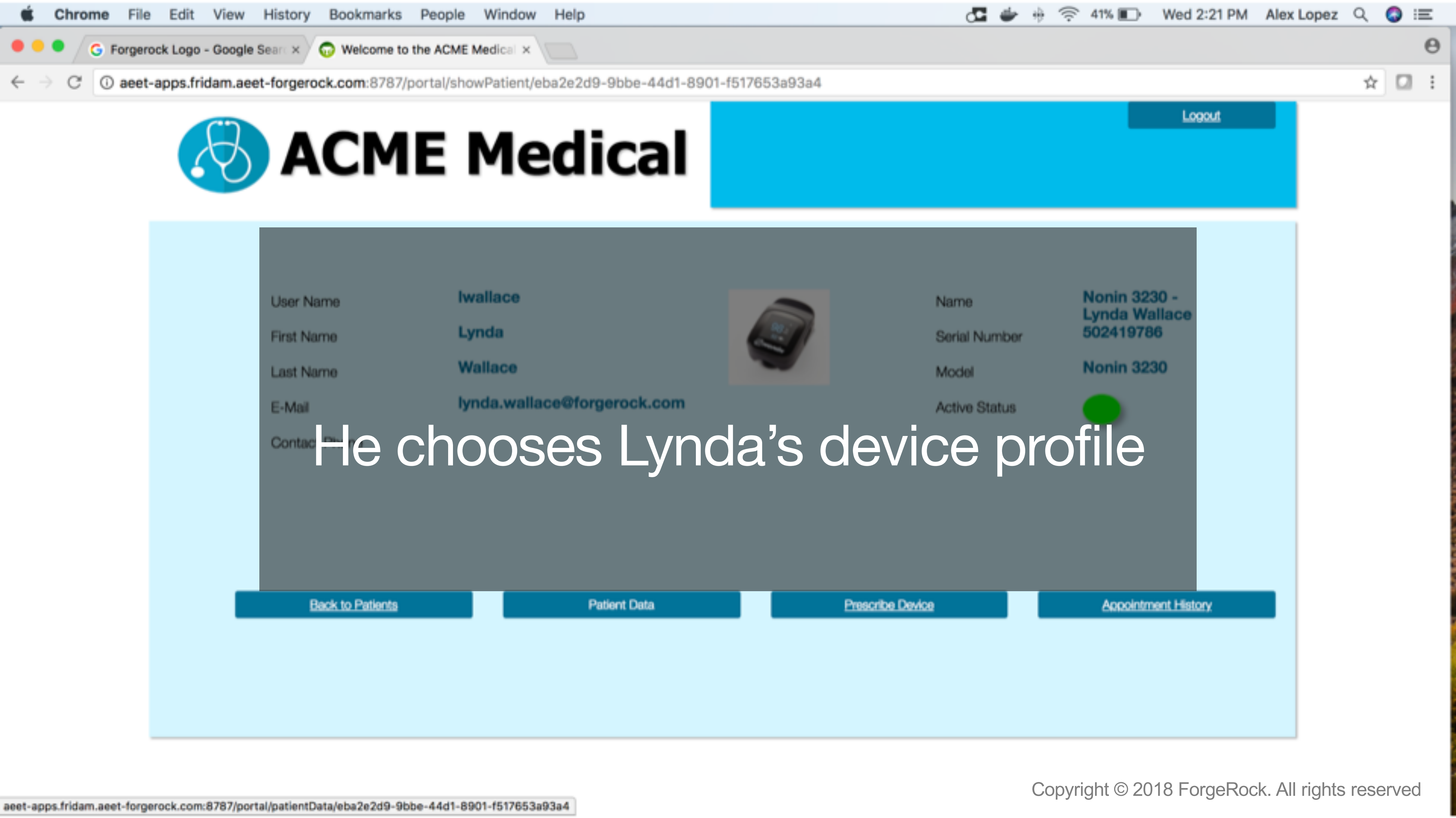
In his My Patients view, Dr. Lopez sees a listing with Lynda Wallace and others


Chrome File Edit View History Bookmarks People Window Help
41%
Wed 2:19 PM
Alex Lopez
Forgerock Logo - Google Sear
Welcome to the ACME MEdical
aeet-apps.fridam.aeet-forgerock.com:8787/portal/patientData/eba2e2d9-9bbe-44d1-8901-f517653a93a4
ACME Medical
Logout
Back to Lynda Wallace

Name	Serial Number	Model	Active Status
Nonin 3230 - Lynda Wallace	502419786	Nonin 3230	True

Select a device to Display its Readings.

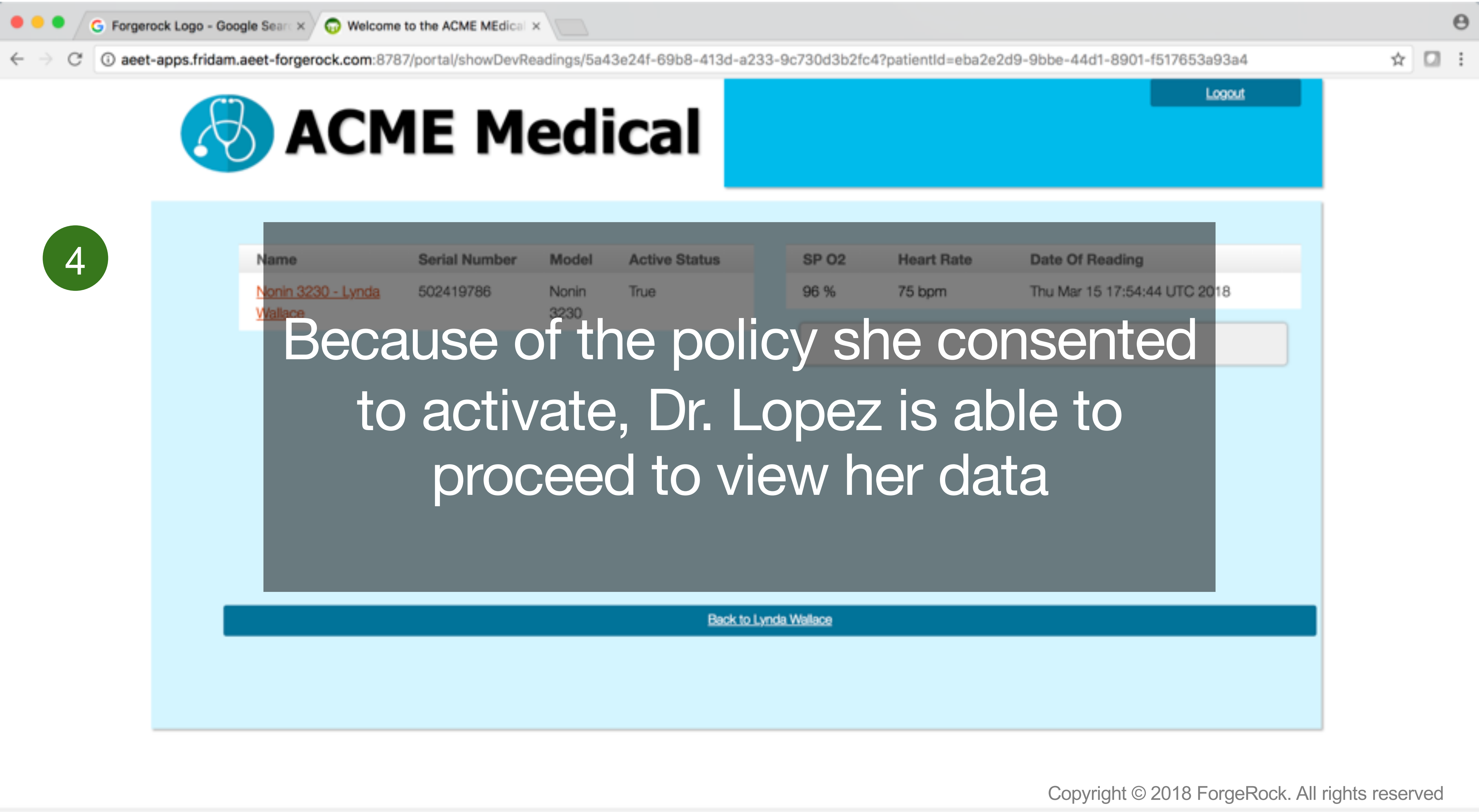
Dr. Lopez selects Lynda Wallace as the patient whose data he wants to view


Chrome File Edit View History Bookmarks People Window Help
41%
Wed 2:21 PM
Alex Lopez
Forgerock Logo - Google Sear ×
Welcome to the ACME Medical ×
aeet-apps.fridam.aeet-forgerock.com:8787/portal/showPatient/eba2e2d9-9bbe-44d1-8901-f517653a93a4
ACME Medical
Logout
Back to Patients
Patient Data
Prescribe Device
Appointment History
aeet-apps.fridam.aeet-forgerock.com:8787/portal/patientData/eba2e2d9-9bbe-44d1-8901-f517653a93a4

User Name	lwallace
First Name	Lynda
Last Name	Wallace
E-Mail	lynda.wallace@forgerock.com
Contact Ph	

Name	Nonin 3230 - Lynda Wallace
Serial Number	502419786
Model	Nonin 3230
Active Status	🟢

He chooses Lynda’s device profile



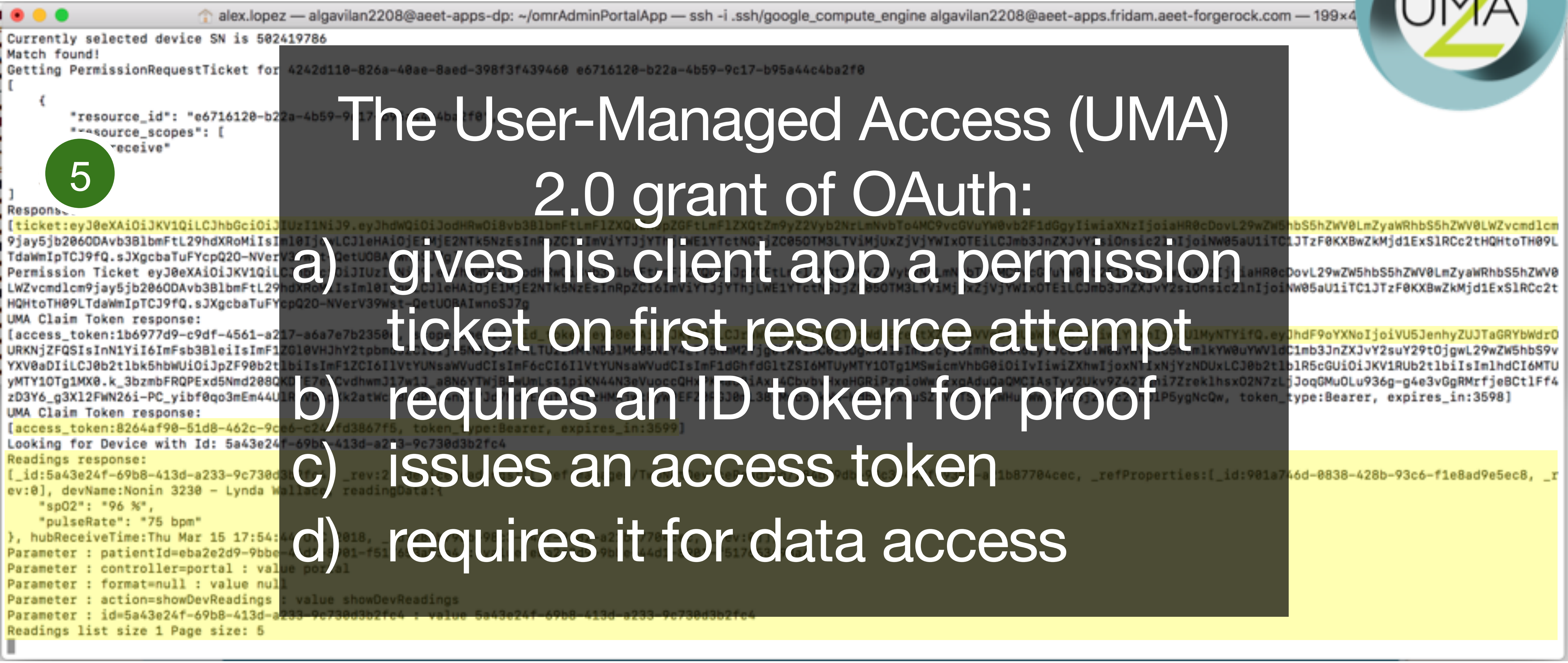
aeet-apps.fridam.aeet-forgerock.com:8787/portal/showDevReadings/5a43e24f-69b8-413d-a233-9c730d3b2fc4?patientId=eba2e2d9-9bbe-44d1-8901-f517653a93a4

4

Name	Serial Number	Model	Active Status
Nonin 3230 - Lynda Wallace	502419786	Nonin 3230	True

SP O2	Heart Rate	Date Of Reading
96 %	75 bpm	Thu Mar 15 17:54:44 UTC 2018

Because of the policy she consented to activate, Dr. Lopez is able to proceed to view her data



alex.lopez — algavilan2208@aeet-apps-dp: ~/omrAdminPortalApp — ssh -i .ssh/google_compute_engine algavilan2208@aeet-apps.fridam.aeet-forgerock.com — 199×4

```
Currently selected device SN is 502419786
Match found!
Getting PermissionRequestTicket for 4242d110-826a-40ae-8aed-398f3f439460 e6716120-b22a-4b59-9c17-b95a44c4ba2f0
[
    {
        "resource_id": "e6716120-b2
        "resource_scopes": [
            "receive"

]
Response:
[ticket:eyJ0eXAiOiJKV1QiLCJhbGciOiJ                                                          9wZW5hbS5hZWV0LmZyaWRhbS5hZWV0LWZvcmdlcm
9jay5jb206ODAvb3BlbmFtL29hdXRoMiIsI                                                          U1iTC1JTzF0KXBwZkMjd1ExSlRCc2tHQHtoTH09L
TdaWmIpTCJ9fQ.sJXgcbaTuFYcpQ2O-NVer
Permission Ticket eyJ0eXAiOiJKV1QiL                                                          aHR0cDovL29wZW5hbS5hZWV0LmZyaWRhbS5hZWV0
LWZvcmdlcm9jay5jb206ODAvb3BlbmFtL29                                                          nIjoiNW05aU1iTC1JTzF0KXBwZkMjd1ExSlRCc2t
HQHtoTH09LTdaWmIpTCJ9fQ.sJXgcbaTuFY
UMA Claim Token response:
[access_token:1b6977d9-c9df-4561-a2                                                          fQ.eyJhdF9oYXNoIjoiVU5JenhyZUJTaGRYbWdrO
URKNjZFQSIsInN1YiI6ImFsb3BleiIsImF1                                                          YWVldC1mb3JnZXJvY2suY29tOjgwL29wZW5hbS9v
YXV0aDIiLCJ0b2tlbk5hbWUiOiJpZF90b2t                                                          0b2tlblR5cGUiOiJKV1RUb2tlbiIsImlhdCI6MTU
yMTY1OTg1MX0.k_3bzmbFRQPExd5Nmd208Q                                                          2N7zLjJoqGMuOLu936g-g4e3vGgRMrfjeBCtlFf4
zD3Y6_g3Xl2FWN26i-PC_yibf0qo3mEm44U                                                          oken_type:Bearer, expires_in:3598]
UMA Claim Token response:
[access_token:8264af90-51d8-462c-9c
Looking for Device with Id: 5a43e24
Readings response:
[_id:5a43e24f-69b8-413d-a233-9c730d                                                          901a746d-0838-428b-93c6-f1e8ad9e5ec8, _r
ev:0], devName:Nonin 3230 - Lynda W
    "spO2": "96 %",
    "pulseRate": "75 bpm"
}, hubReceiveTime:Thu Mar 15 17:54:
Parameter : patientId=eba2e2d9-9bbe
Parameter : controller=portal : value portal
Parameter : format=null : value null
Parameter : action=showDevReadings : value showDevReadings
Parameter : id=5a43e24f-69b8-413d-a233-9c730d3b2fc4 : value 5a43e24f-69b8-413d-a233-9c730d3b2fc4
Readings list size 1 Page size: 5
```

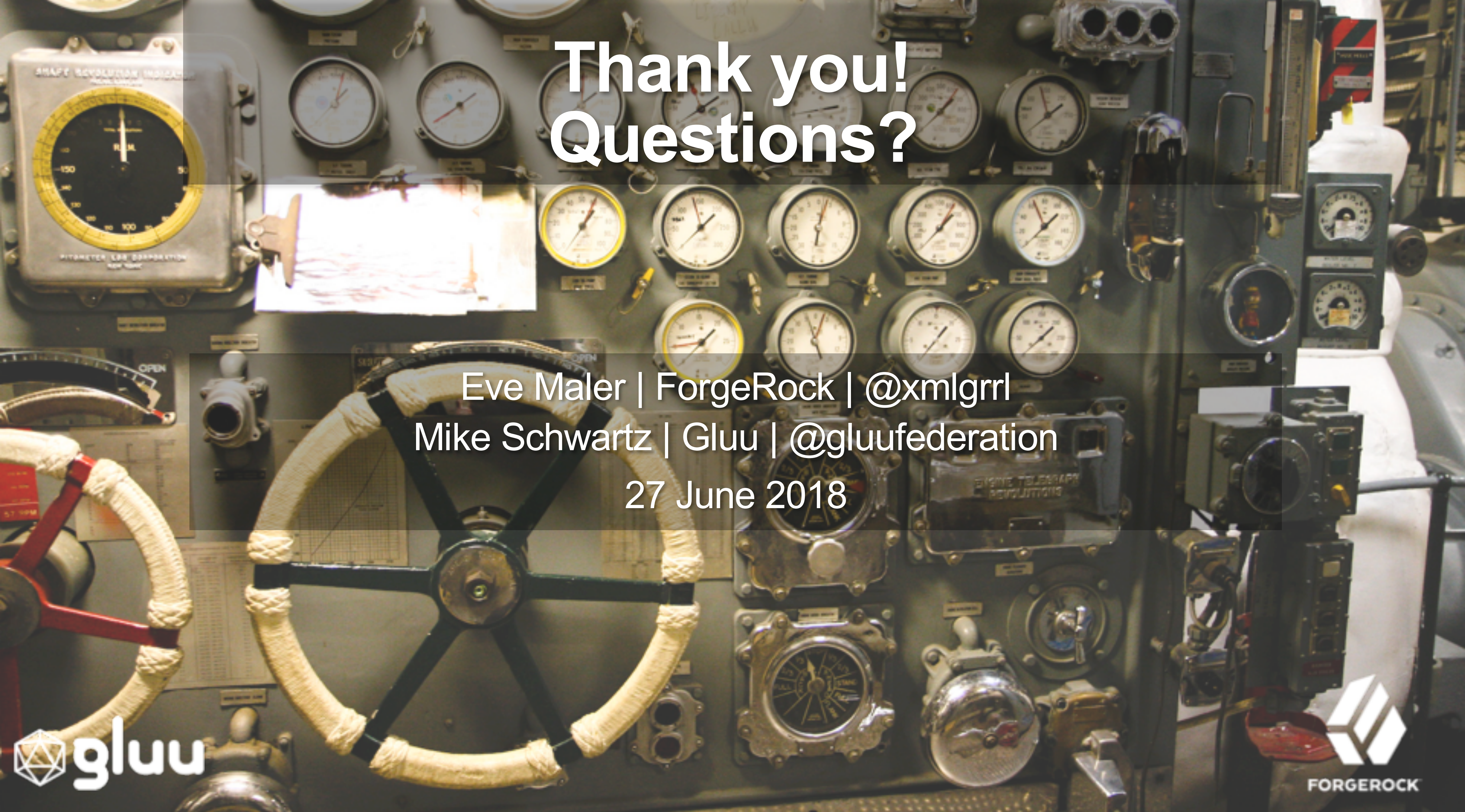
The User-Managed Access (UMA) 2.0 grant of OAuth:

a) gives his client app a permission ticket on first resource attempt

b) requires an ID token for proof

c) issues an access token

d) requires it for data access



Thank you! Questions?

Eve Maler | ForgeRock | @xmlgrrl

Mike Schwartz | Gluu | @gluufederation

27 June 2018

https://www.flickr.com/photos/mrcrash/1980165121/ | CC BY-NC 2.0