

What Is User-Managed Access And Why Do We Need It?

Presented by several UMANitarians
(participants in the Kantara UMA Work Group)
with WG chair Eve Maler as your emcee



Agenda

Introduction: What is UMA, why do we need it, and how far along is it?

Use cases illustrating UMA's unique strengths

Why would an organization want to UMA-enable its apps?

Existing UMA implementations

How UMA works to build technical and business trust

Q&A

Privacy is not about secrecy



The goal of a flexible, user-centric identity management infrastructure must be to allow the user to quickly determine what information will be revealed to which parties and for what purposes, how trustworthy those parties are and how they will handle the information, and what the consequences of sharing their information will be”

– Ann Cavoukian, Information and Privacy Commissioner of Ontario,
Privacy in the Clouds paper



It's about context, control, choice, and respect

The price for sharing access to our data is too high

The price for sharing access to our data is too high

Either we have to do all the work ourselves

Price for Using Our "Free" Website

*"Remember...
You're not the customer, you're the product!"*



GraphJam.com

*...often in the role of the "product,"
not the "customer"*

The price for sharing access to our data is too high

Either we have to do all the work ourselves

Or we have to agree to install large data pipelines

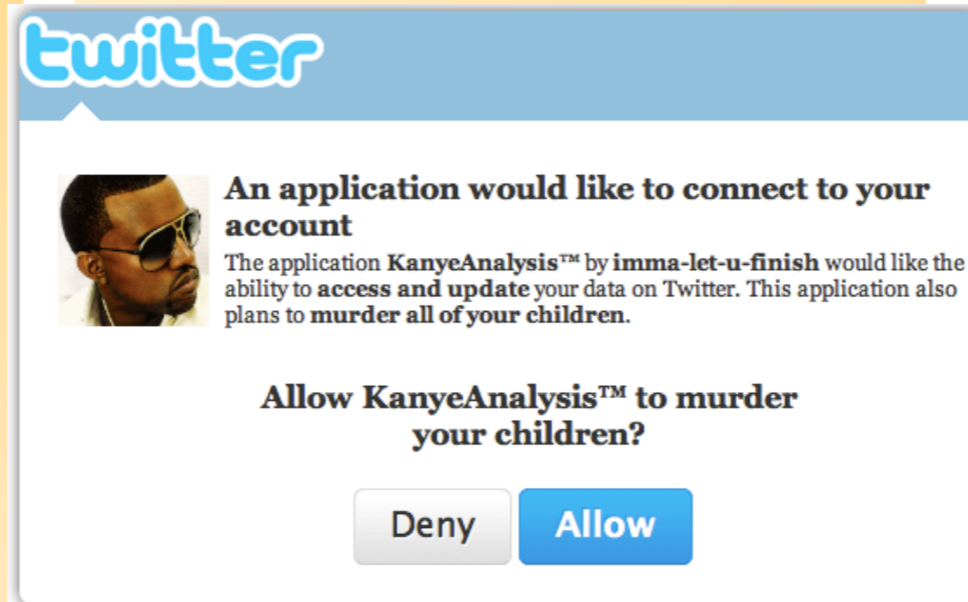
Price for Using Our "Free" Website

*"Remember...
You're not the customer, you're the product!"*



GraphJam.com

...often in the role of the "product," not the "customer"



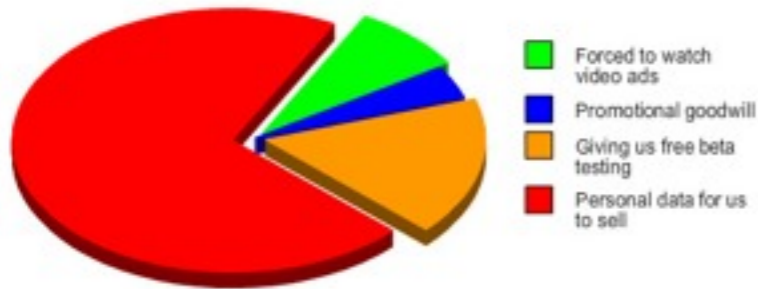
...resulting in oversharing of high-quality data and a "too many subscriptions" problem

The price for sharing access to our data is too high

Either we have to do all the work ourselves

Price for Using Our "Free" Website

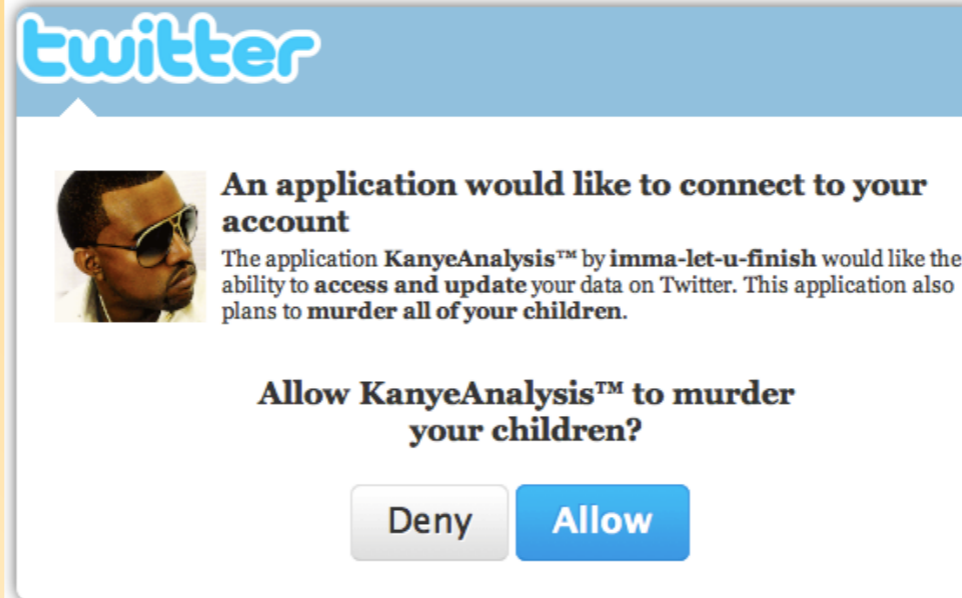
*"Remember...
You're not the customer, you're the product!"*



GraphJam.com

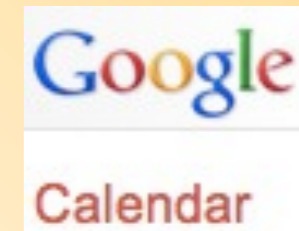
...often in the role of the "product," not the "customer"

Or we have to agree to install large data pipelines



...resulting in oversharing of high-quality data and a "too many subscriptions" problem

Or we share with friends through "secret links"



Your calendar's Private Address is designed for your use only. All of your calendar information is available via your private links, so don't share this address with others.

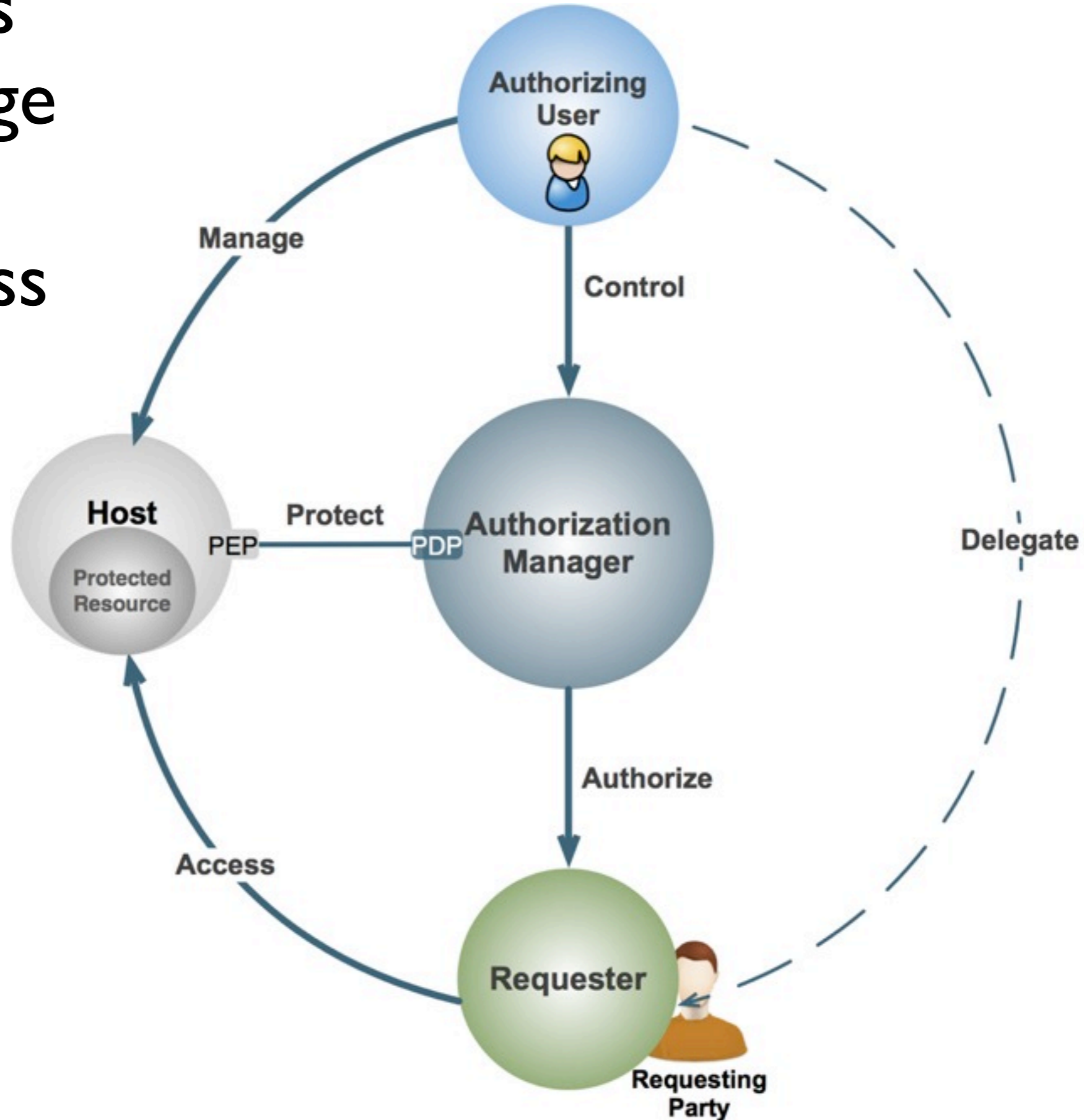
To change your Private Address and disable any previous access, click the **Reset Private URLs** link.

...rebuilding friend lists over and over – and hoping they won't give away the store

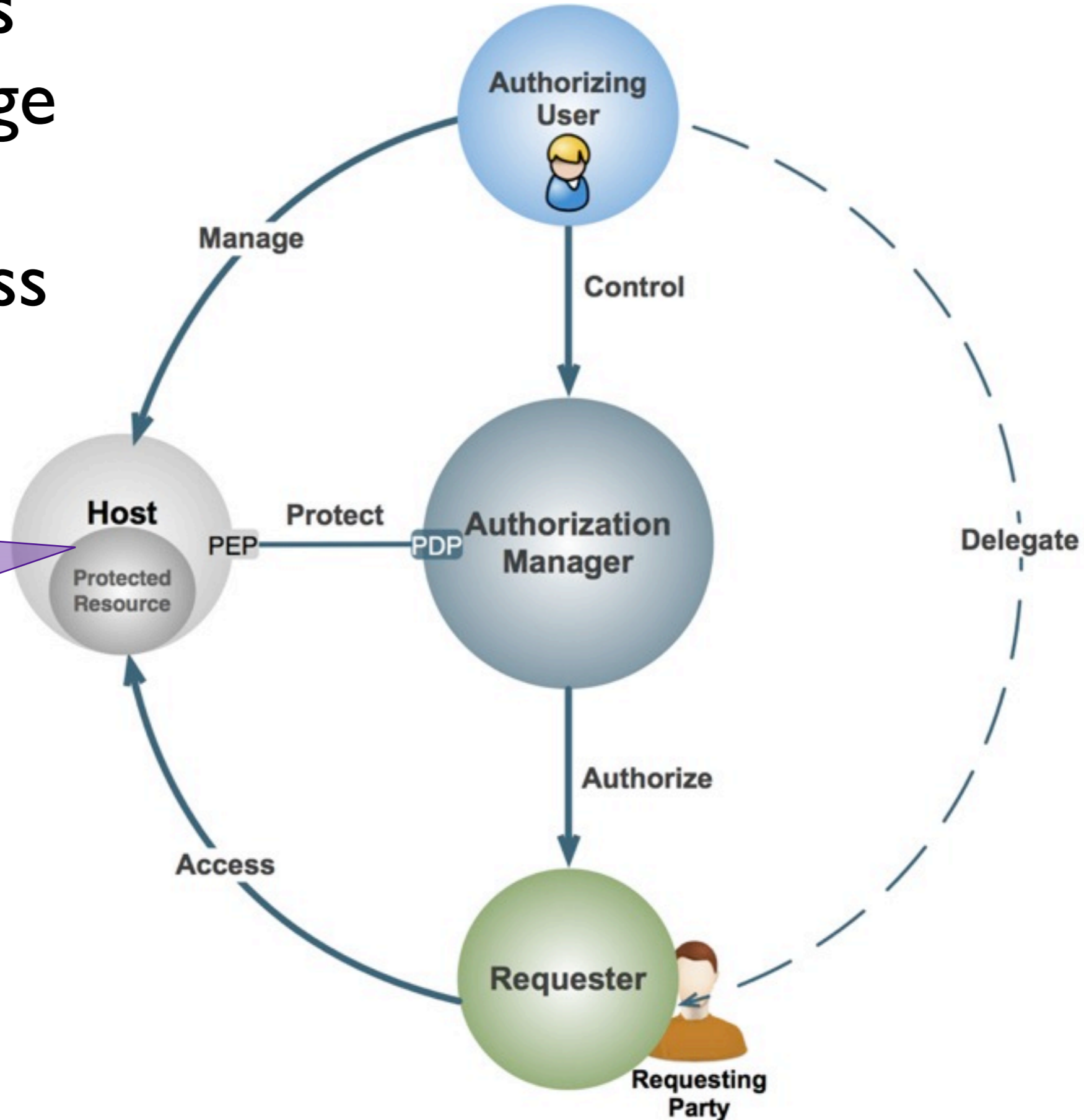
UMA is...

- A web protocol that lets you control authorization of data sharing and service access made on your behalf
- A set of draft specifications that is free for anyone to implement
- Undergoing multiple implementation efforts
- A Work Group of the Kantara Initiative that is free for anyone to **join** and contribute to
- Striving to be simple, OAuth-based, identifier-agnostic, RESTful, modular, generative, and developed rapidly
- Contributed to the IETF for consideration as an Internet-Draft (rev 02)
- Heading towards interoperability testing and increased OpenID Connect integration in early 2012

UMA enables you to manage sharing and protect access from a single hub



UMA enables you to manage sharing and protect access from a single hub

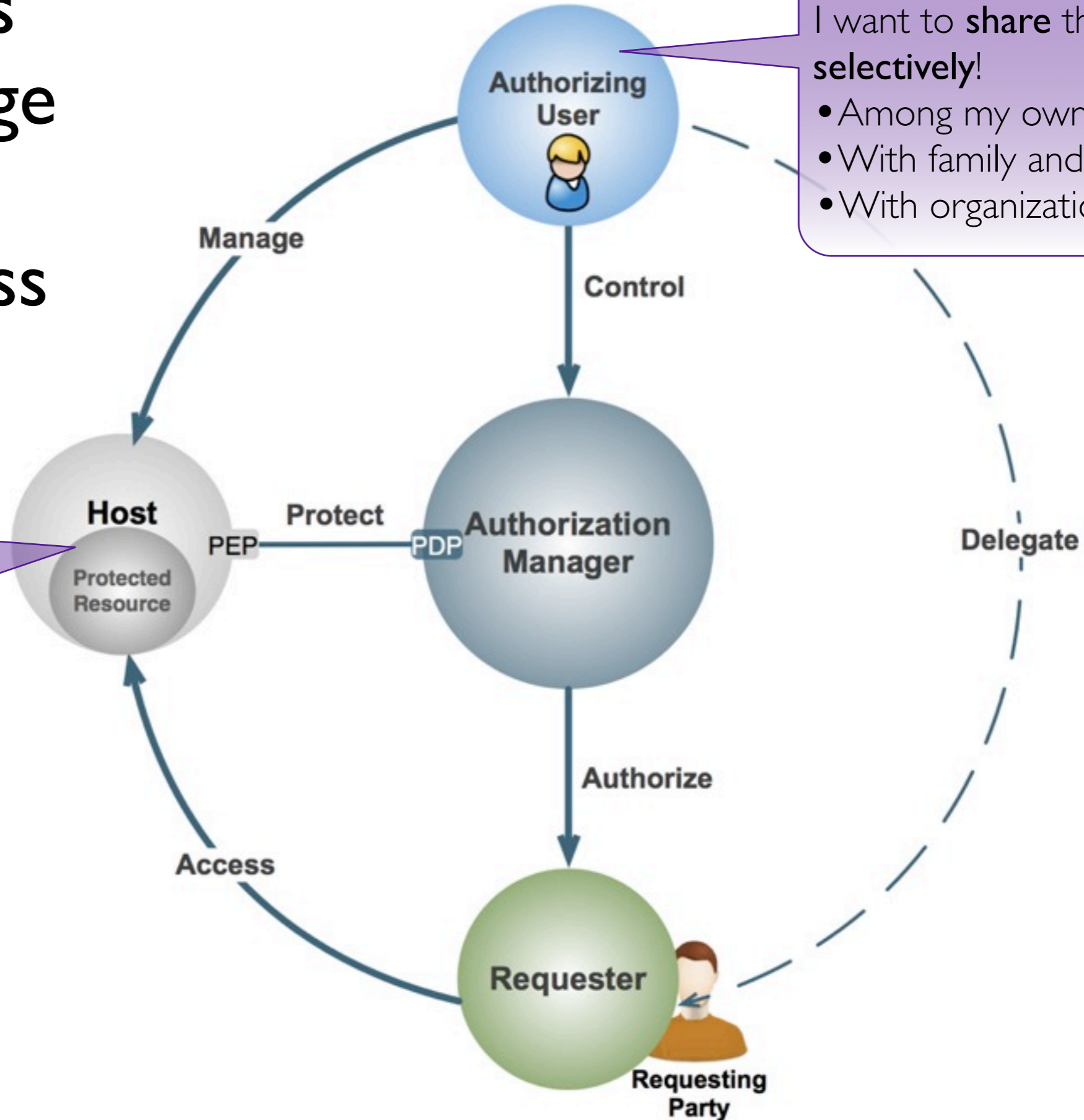


- Historical
- Biographical
- Reputation
- Vocational
- Artistic/user-generated
- Social
- Location/geolocation
- Computational
- Genealogical
- Biological/medical
- Legal
- ...

tinyurl.com/umawg



UMA enables you to manage sharing and protect access from a single hub



I want to **share** this stuff **selectively!**

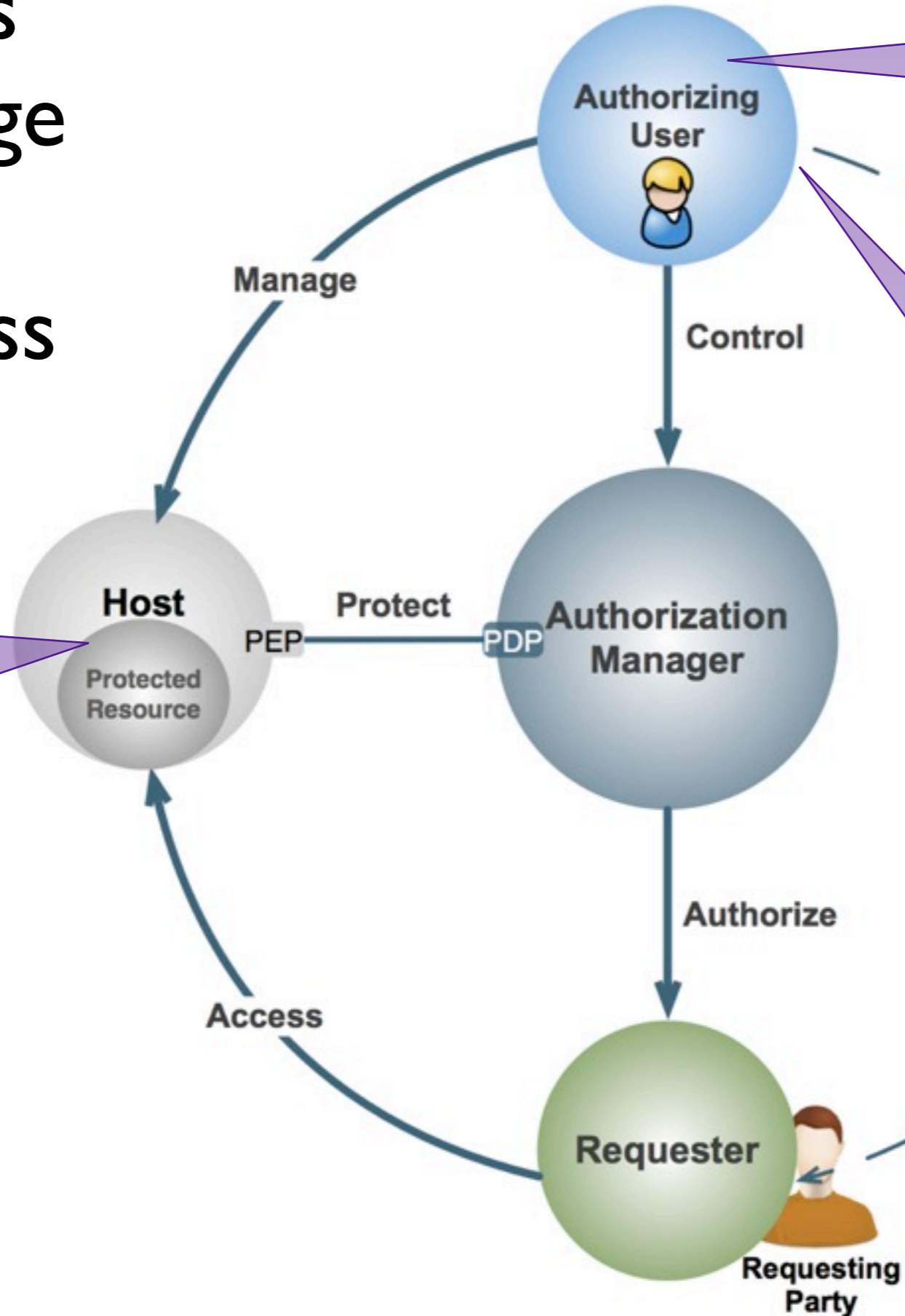
- Among my own apps
- With family and friends
- With organizations

- Historical
- Biographical
- Reputation
- Vocational
- Artistic/user-generated
- Social
- Location/geolocation
- Computational
- Genealogical
- Biological/medical
- Legal
- ...

tinyurl.com/umawg



UMA enables you to manage sharing and protect access from a single hub



I want to **share** this stuff **selectively!**

- Among my own apps
- With family and friends
- With organizations

I want to **protect** this stuff from being seen by everyone in the world!

- Historical
- Biographical
- Reputation
- Vocational
- Artistic/user-generated
- Social
- Location/geolocation
- Computational
- Genealogical
- Biological/medical
- Legal
- ...



UMA gives users a digital footprint dashboard



UMA gives users a digital footprint dashboard

*Web 2.0 access control
today is inconsistent and
unsophisticated*



UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You can unify access control under one AM



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You can unify access control under one AM



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

You can unify access control under one AM

Your AM can test for claims like “over 18”

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You must be online in order to authorize access by others



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

You can unify access control under one AM

Your AM can test for claims like “over 18”

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You must be online in order to authorize access by others



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

You can unify access control under one AM

Your AM can test for claims like “over 18”

You can set up policies that work while you’re away

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You must be online in order to authorize access by others

You can't "advertise" your content without giving it away



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

You can unify access control under one AM

Your AM can test for claims like "over 18"

You can set up policies that work while you're away

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You must be online in order to authorize access by others

You can't "advertise" your content without giving it away



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

You can unify access control under one AM

Your AM can test for claims like "over 18"

You can set up policies that work while you're away

You can control access to stuff with public URLs

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You must be online in order to authorize access by others

You can't "advertise" your content without giving it away

You can't get a global view of all your sharing relationships



Source: <http://www.flickr.com/photos/paraflyer/2749336420/>

You can unify access control under one AM

Your AM can test for claims like "over 18"

You can set up policies that work while you're away

You can control access to stuff with public URLs

UMA gives users a digital footprint dashboard

Web 2.0 access control today is inconsistent and unsophisticated

You have to name known people in order to share with others

You must be online in order to authorize access by others

You can't "advertise" your content without giving it away

You can't get a global view of all your sharing relationships



You can unify access control under one AM

Your AM can test for claims like "over 18"

You can set up policies that work while you're away

You can control access to stuff with public URLs

You can manage and revoke access from one place

Agenda

Introduction: What is UMA, why do we need it, and how far along is it?

Use cases illustrating UMA's unique strengths

Why would an organization want to UMA-enable its apps?

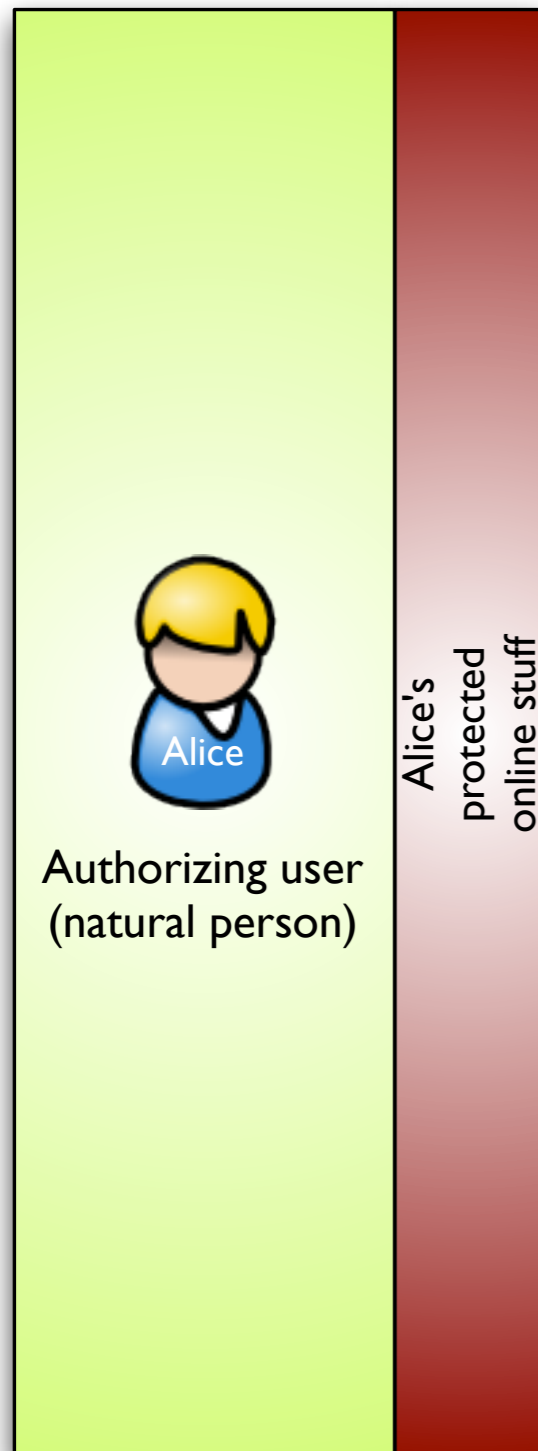
Existing UMA implementations

How UMA works to build technical and business trust

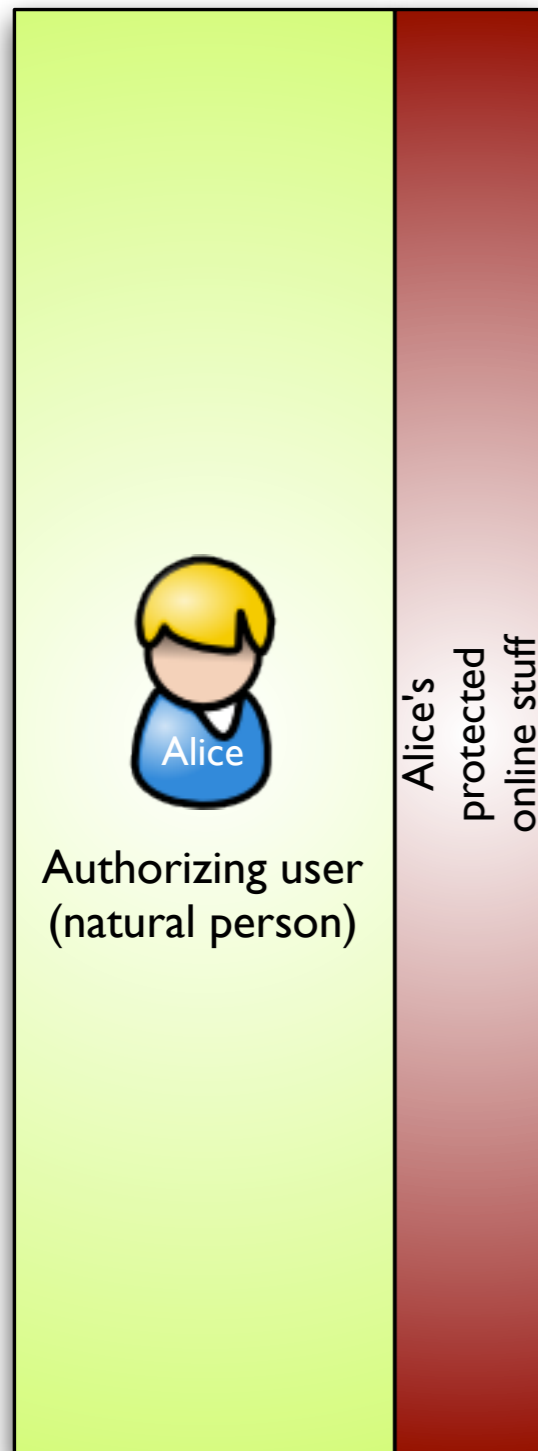
Q&A

UMA data sharing constellations @@how to put OpenID Connect as well?

UMA data sharing constellations @@how to put OpenID Connect as well?



UMA data sharing constellations @@how to put OpenID Connect as well?

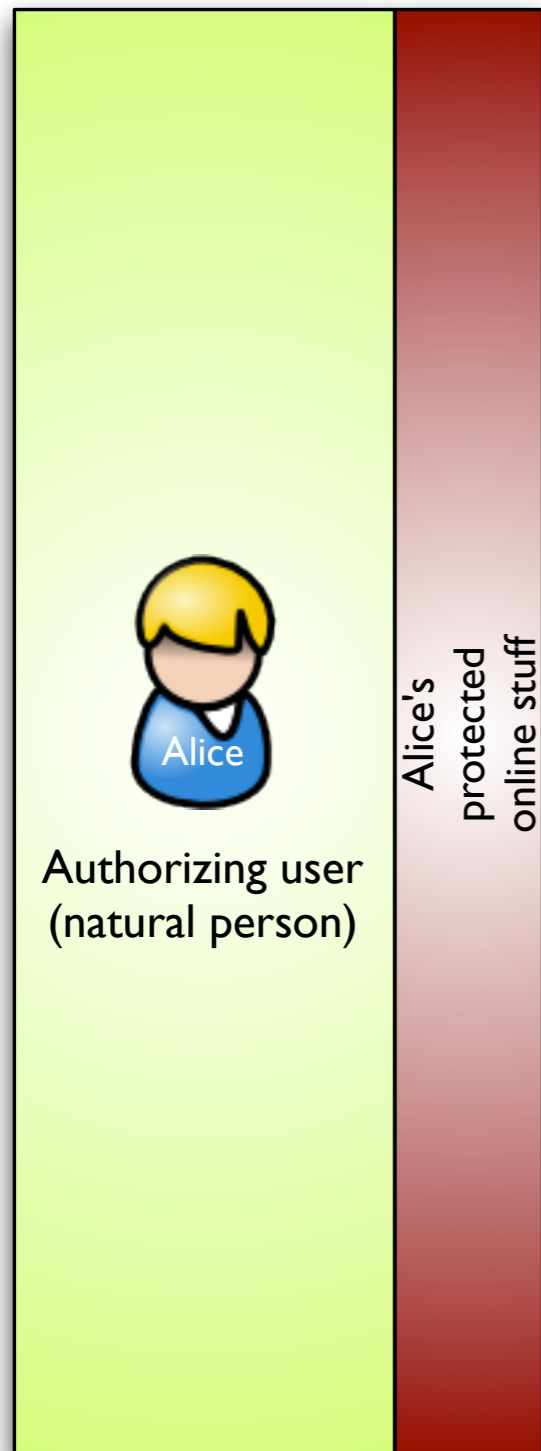


Similar to OAuth-mediated access:
Alice wants a client app she uses to
regularly access a service she uses



Requesting user
(person-to-self
sharing)

UMA data sharing constellations @@how to put OpenID Connect as well?

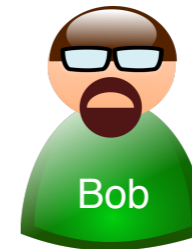


Similar to OAuth-mediated access:
Alice wants a client app she uses to regularly access a service she uses



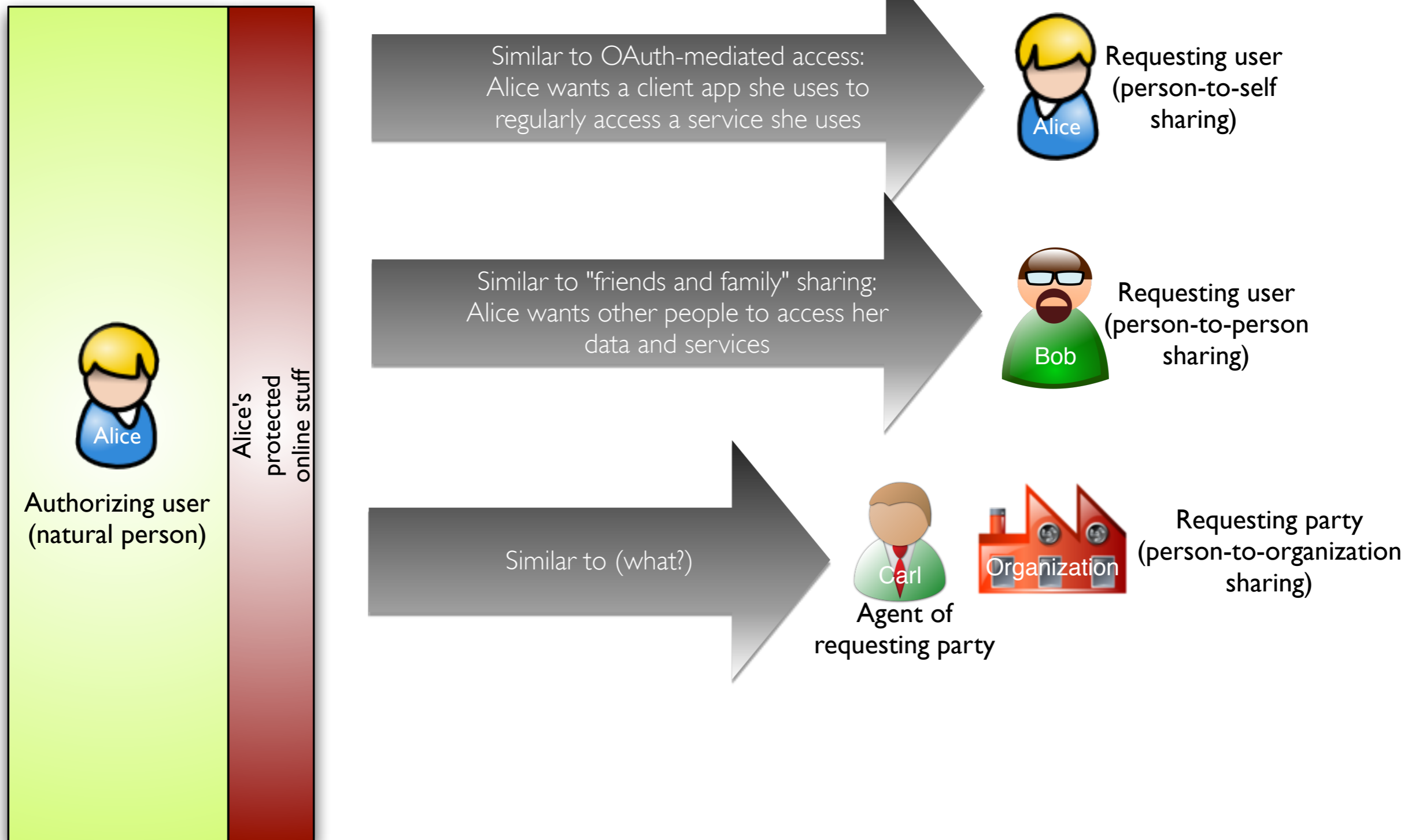
Requesting user
(person-to-self sharing)

Similar to "friends and family" sharing:
Alice wants other people to access her data and services

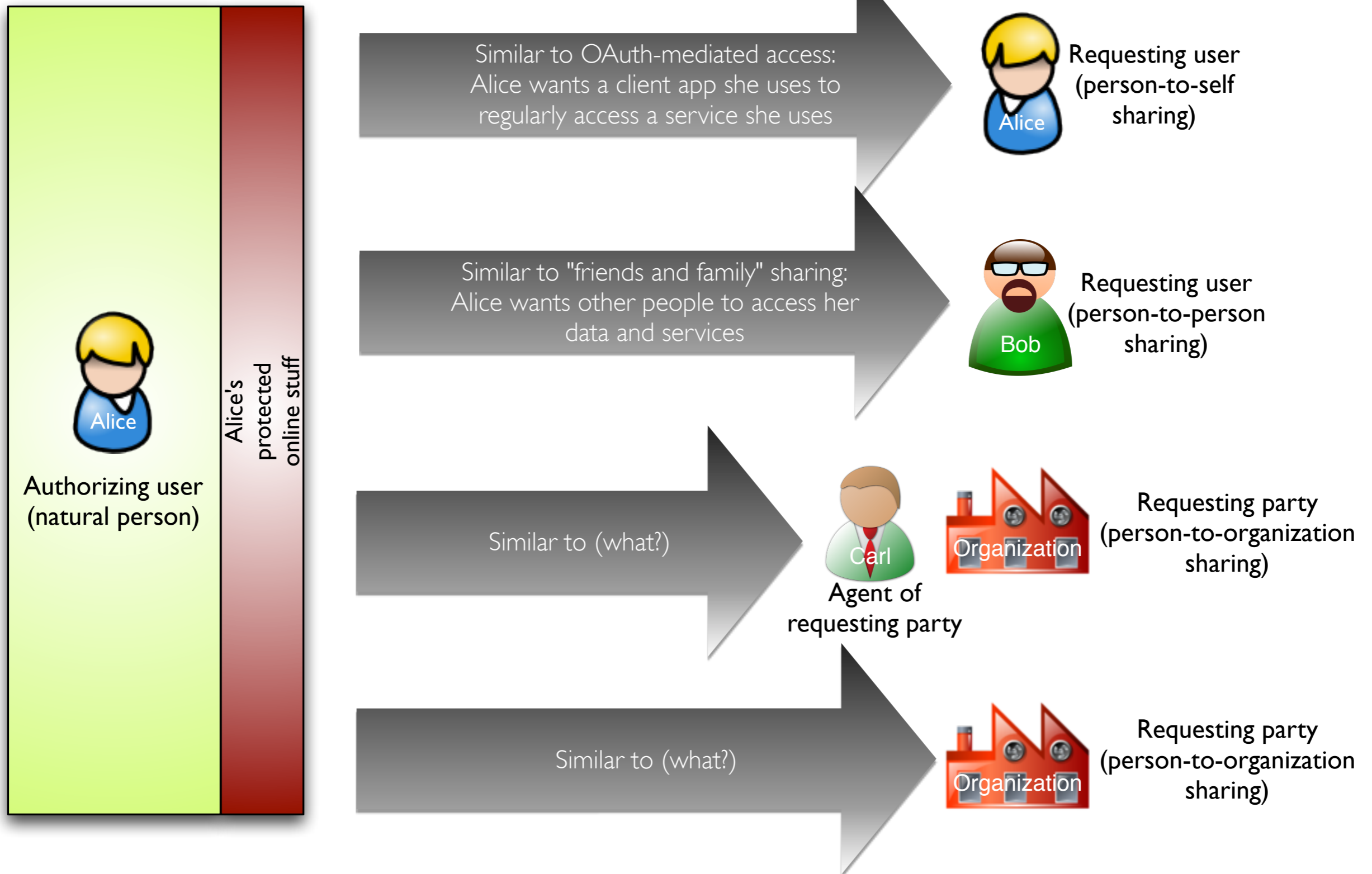


Requesting user
(person-to-person sharing)

UMA data sharing constellations @@how to put OpenID Connect as well?



UMA data sharing constellations @@how to put OpenID Connect as well?



Use case: Protecting hData electronic health records

- @@Discuss need for high security, compliance, and dynamic introduction of parties

Use case: Sharing trusted identity attributes à la NSTIC

- @@Discuss “personal data store”/”personal data locker” implications
- @@Discuss street identity example?
- @@Discuss roles for trust frameworks and LOAs (more to come on UMA’s trust model)
- @@Set up SMART and TAS3 segments

Use case: Selectively sharing photos and albums

- @@Discuss advantages of standardized scopes and resource sets
- @@Set up Fraunhofer segment

Agenda

Introduction: What is UMA, why do we need it, and how far along is it?

Use cases illustrating UMA's unique strengths

Why would an organization want to UMA-enable its apps?

Existing UMA implementations

How UMA works to build technical and business trust

Q&A

Web apps that become UMA hosts can easily offer “context, control, choice, and respect”



Web apps that become UMA hosts can easily offer “context, control, choice, and respect”

- You can provide sophisticated protection and sharing of any user content or data that isn't meant to be fully public

Web apps that become UMA hosts can easily offer “context, control, choice, and respect”

- You can provide sophisticated protection and sharing of any user content or data that isn't meant to be fully public
- You can outsource the entire job to third parties (AMs)

Web apps that become UMA hosts can easily offer “context, control, choice, and respect”

- You can provide sophisticated protection and sharing of any user content or data that isn't meant to be fully public
- You can outsource the entire job to third parties (AMs)
- You can ensure that the protection of sensitive resources is stronger than the “private URL trick”

Web apps that become UMA hosts can easily offer “context, control, choice, and respect”

- You can provide sophisticated protection and sharing of any user content or data that isn't meant to be fully public
- You can outsource the entire job to third parties (AMs)
- You can ensure that the protection of sensitive resources is stronger than the “private URL trick”
- You can build trust more readily with users who are “privacy fundamentalists”

Web apps that become UMA hosts can easily offer “context, control, choice, and respect”

- You can provide sophisticated protection and sharing of any user content or data that isn't meant to be fully public
- You can outsource the entire job to third parties (AMs)
- You can ensure that the protection of sensitive resources is stronger than the “private URL trick”
- You can build trust more readily with users who are “privacy fundamentalists”
- You can integrate these features using lightweight OAuth, JSON, HTTP, and REST paradigms and a freely implementable protocol

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

- The separation between IdPs and other attribute providers has long been acknowledged – you can never be the sole trusted source of all interesting user data

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

- The separation between IdPs and other attribute providers has long been acknowledged – you can never be the sole trusted source of all interesting user data
- OpenID Connect is solving how you as an IdP can act as a discovery hub for OAuth-mediated access to attributes

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

- The separation between IdPs and other attribute providers has long been acknowledged – you can never be the sole trusted source of all interesting user data
- OpenID Connect is solving how you as an IdP can act as a discovery hub for OAuth-mediated access to attributes
- UMA complements it by solving how you as an IdP can now act as an authorization hub for access by “others besides Alice” to:

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

- The separation between IdPs and other attribute providers has long been acknowledged – you can never be the sole trusted source of all interesting user data
- OpenID Connect is solving how you as an IdP can act as a discovery hub for OAuth-mediated access to attributes
- UMA complements it by solving how you as an IdP can now act as an authorization hub for access by “others besides Alice” to:
 - Trusted attributes

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

- The separation between IdPs and other attribute providers has long been acknowledged – you can never be the sole trusted source of all interesting user data
- OpenID Connect is solving how you as an IdP can act as a discovery hub for OAuth-mediated access to attributes
- UMA complements it by solving how you as an IdP can now act as an authorization hub for access by “others besides Alice” to:
 - Trusted attributes
 - User-generated content

Identity providers that become UMA AMs can centrally coordinate sharing of anything to anyone

- The separation between IdPs and other attribute providers has long been acknowledged – you can never be the sole trusted source of all interesting user data
- OpenID Connect is solving how you as an IdP can act as a discovery hub for OAuth-mediated access to attributes
- UMA complements it by solving how you as an IdP can now act as an authorization hub for access by “others besides Alice” to:
 - Trusted attributes
 - User-generated content
 - APIs

Agenda

Introduction: What is UMA, why do we need it, and how far along is it?

Use cases illustrating UMA's unique strengths

Why would an organization want to UMA-enable its apps?

Existing UMA implementations

How UMA works to build technical and business trust

Q&A

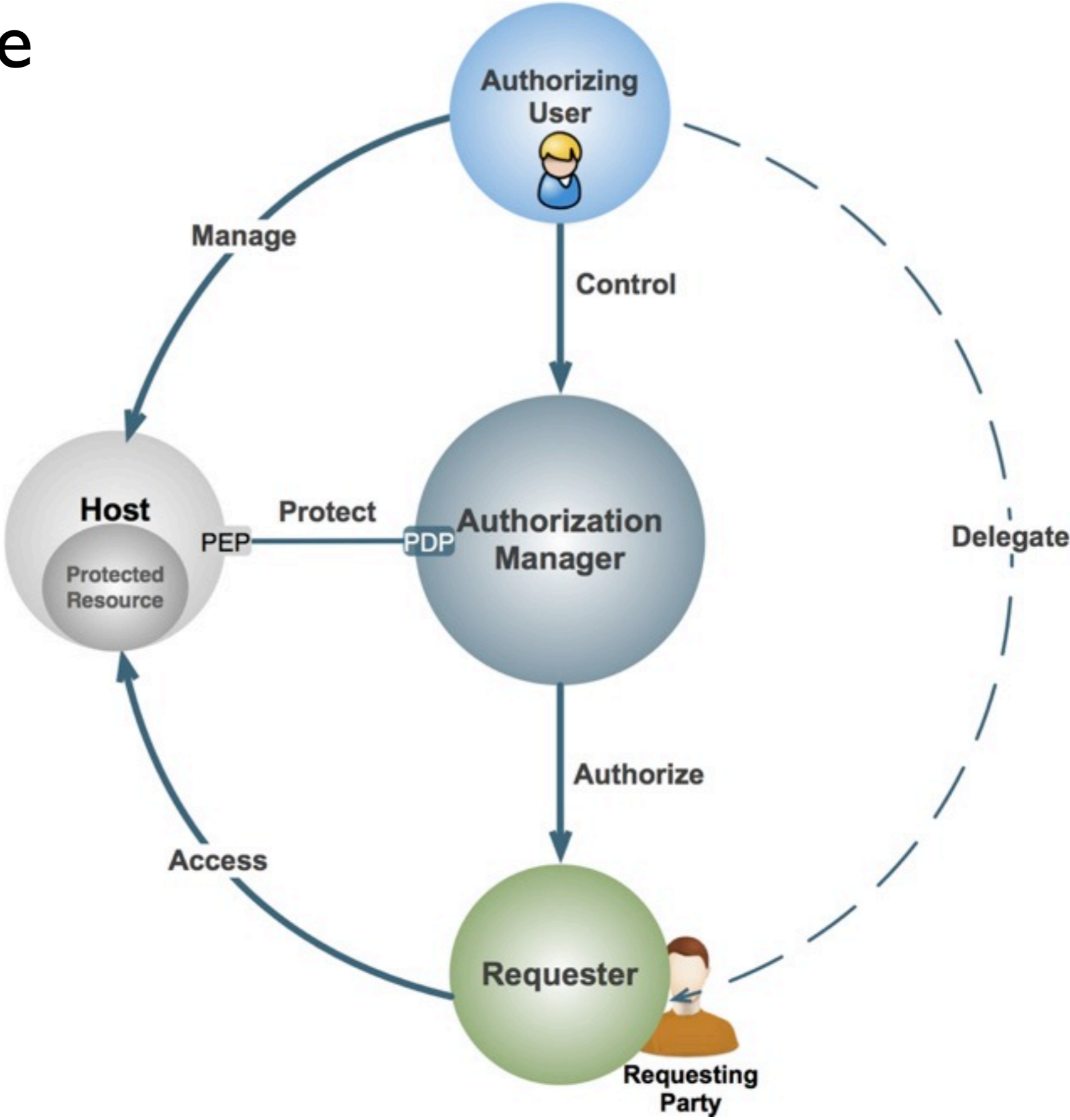
Implementation work to date

- The SMART project at Newcastle University
- Fraunhofer AISEC project
- Synergetics TAS3/UMA integration

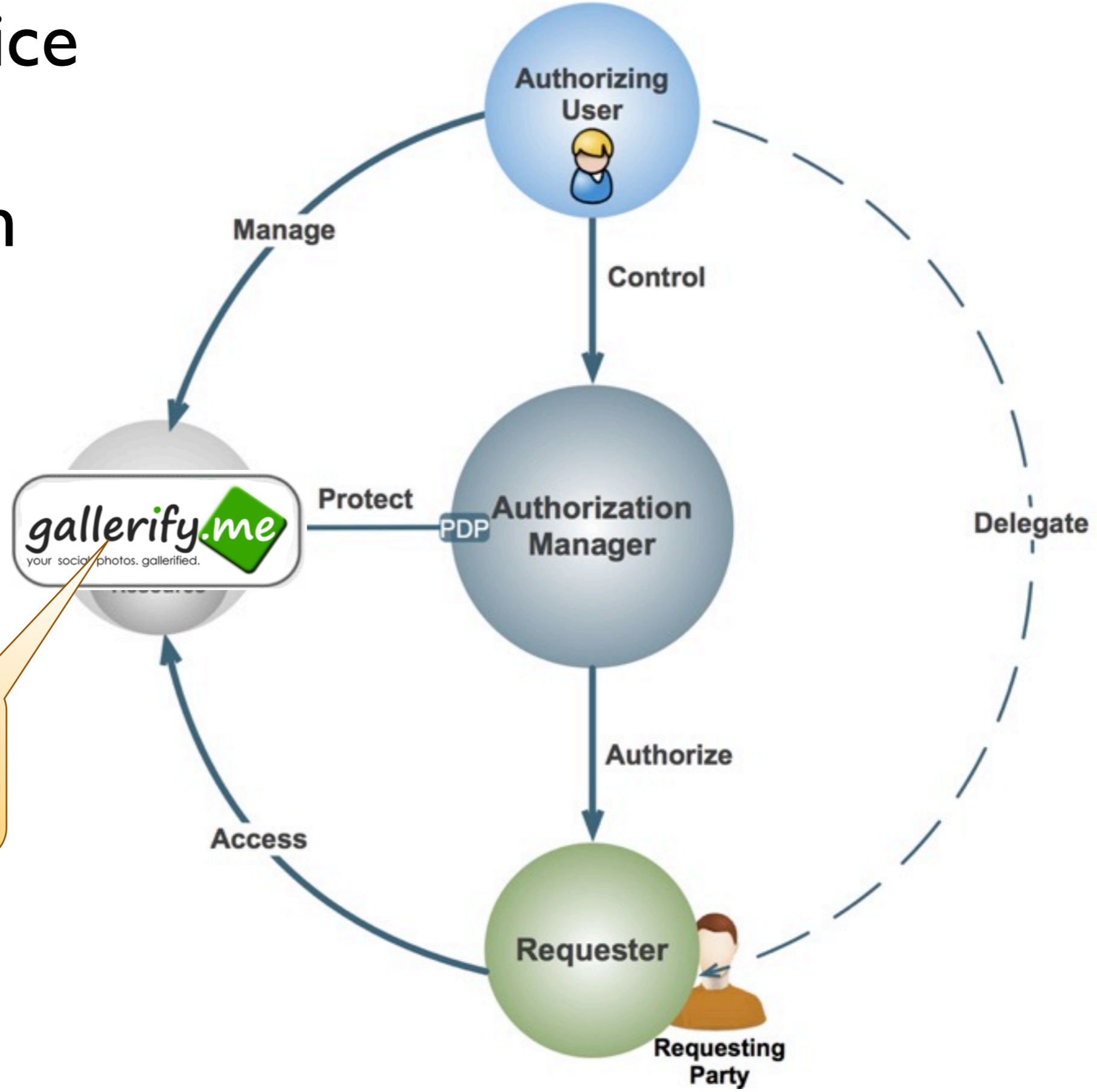
The SMART project is...

- About “Student-Managed Access to Online Resources”
- Taking place at the School of Computing Science, Newcastle University
 - Affiliated with Centre for Cybercrime and Computer Security
 - Team members include Prof. Aad Van Moorsel, Maciej Machulak, Łukasz Moreń, Maciej Wolniak, Chris Franks, and Jacek Szpot
 - JISC-funded
- Planning to open-source its “UMA/j” implementation and sample apps
- See: research.ncl.ac.uk/smart, smartjisc.wordpress.com, and [@smartproject](https://twitter.com/smartproject)

SMART lets Alice share photos selectively with Bob (@@replace)

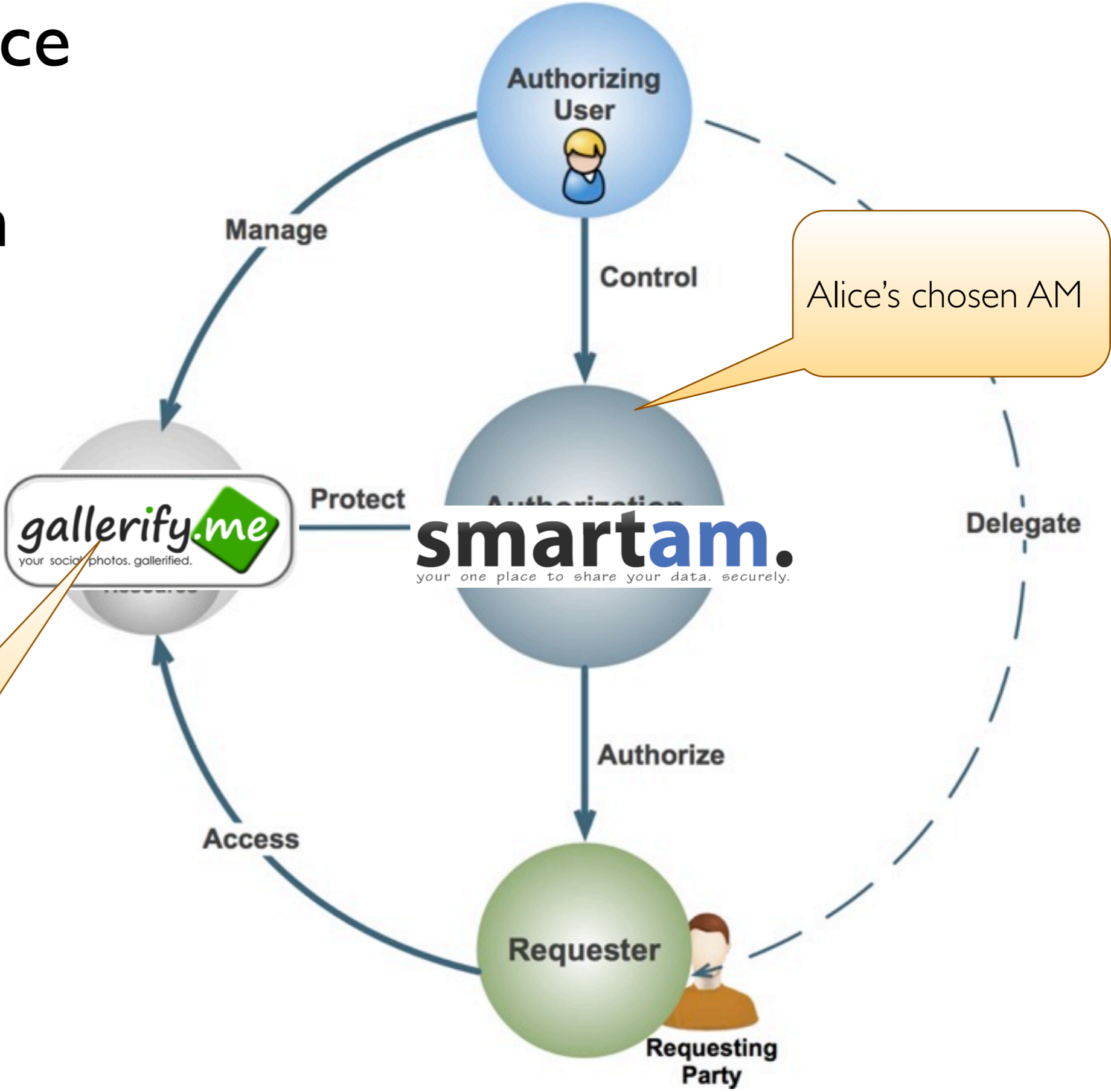


SMART lets Alice share photos selectively with Bob (@@replace)



The photo service Alice uses, with protected albums

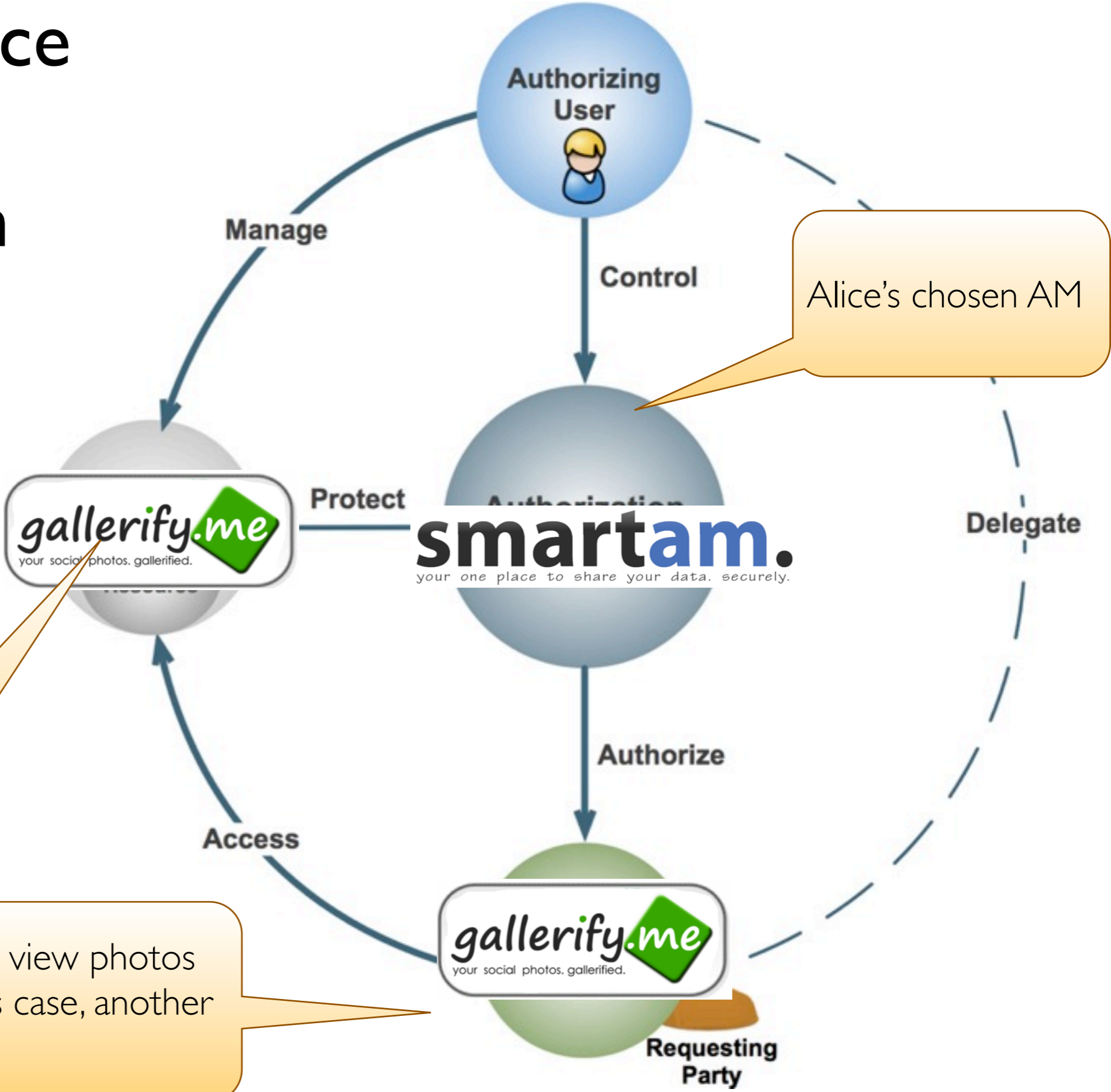
SMART lets Alice share photos selectively with Bob (@@replace)



The photo service Alice uses, with protected albums



SMART lets Alice share photos selectively with Bob (@@replace)

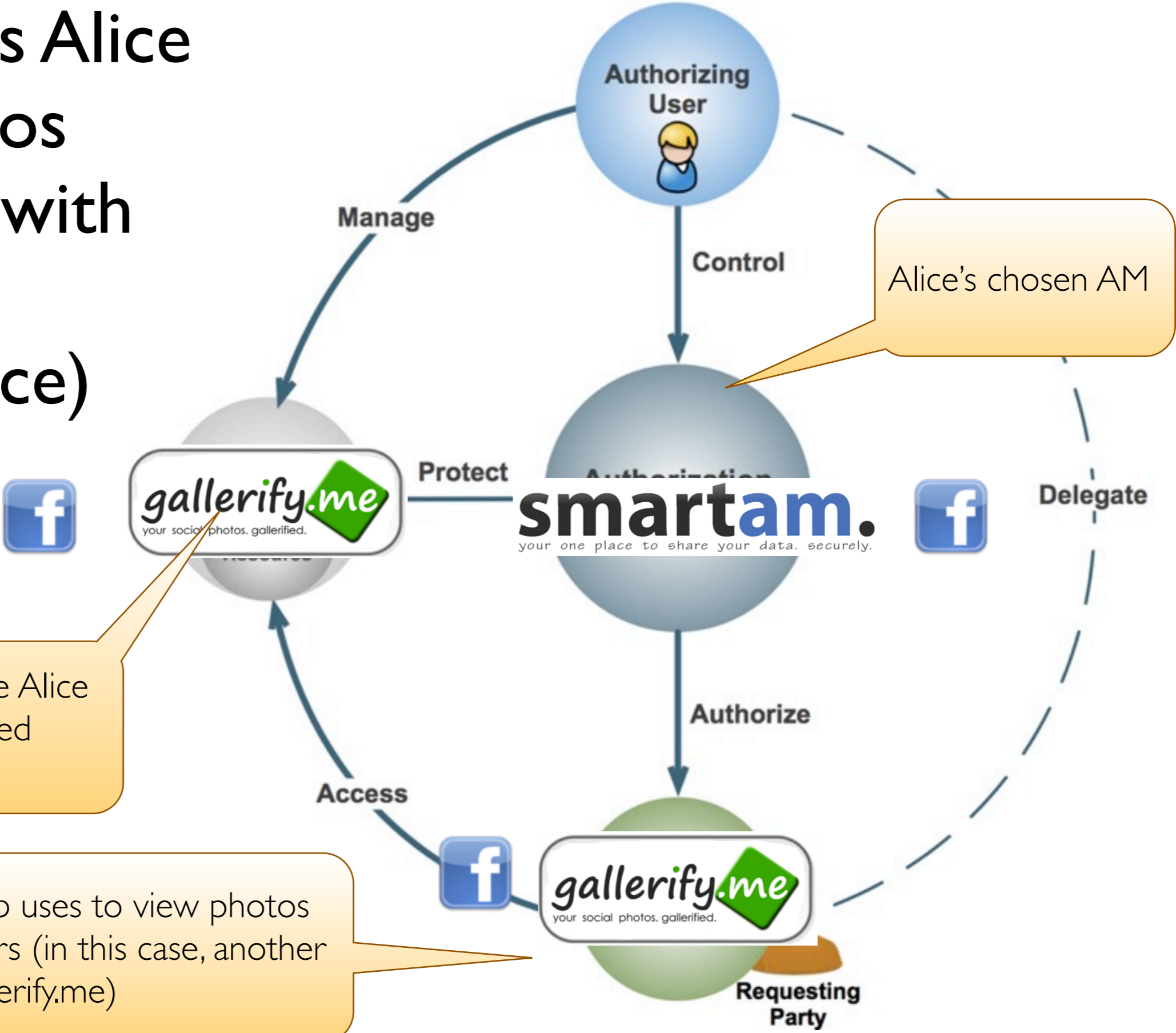


The photo service Alice uses, with protected albums

The service Bob uses to view photos owned by others (in this case, another instance of Gallerify.me)



SMART lets Alice share photos selectively with Bob (@@replace)



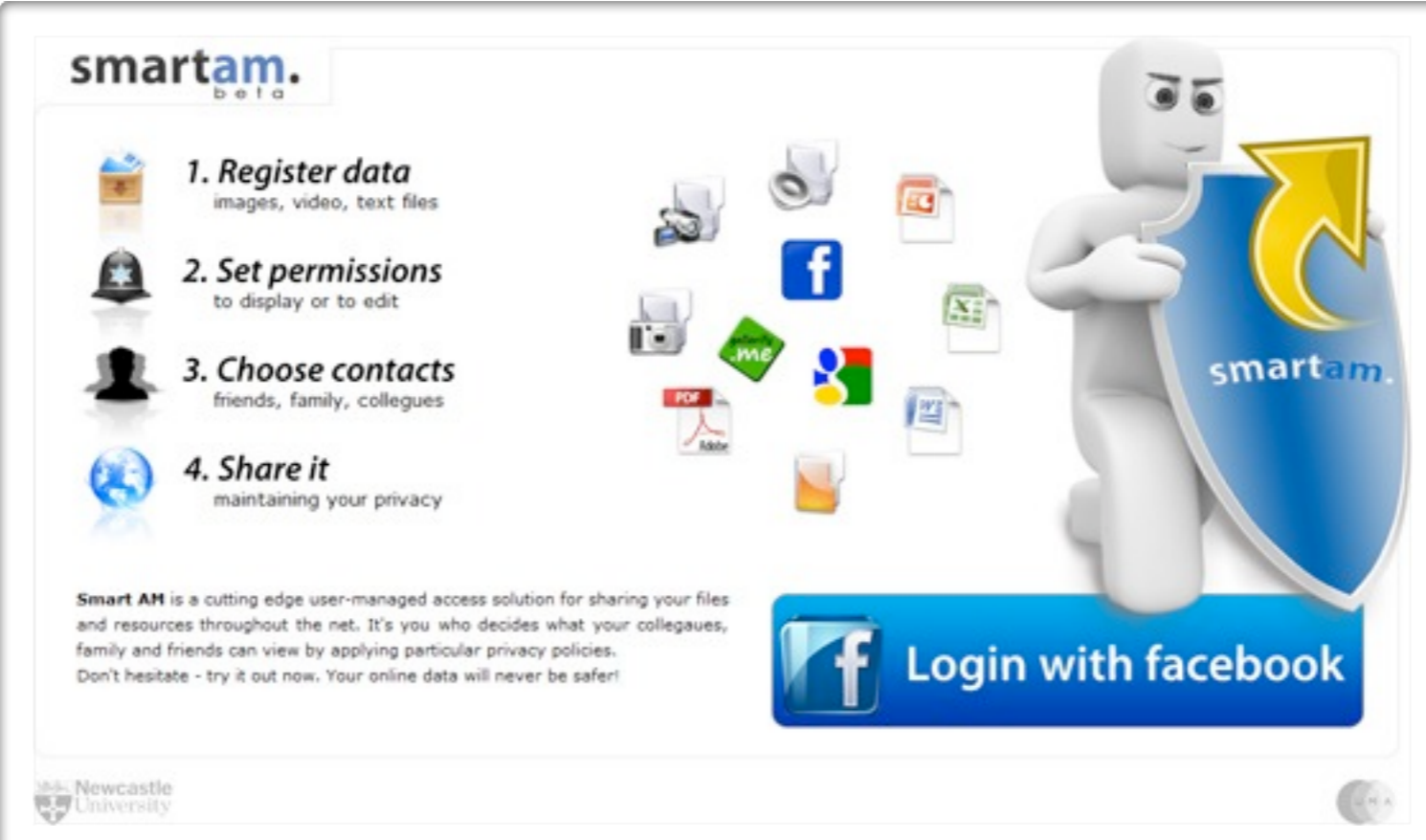
The photo service Alice uses, with protected albums

The service Bob uses to view photos owned by others (in this case, another instance of Gallerify.me)



SMARTAM 2.0 is in public beta: try it for yourself!

- Instructions are on the [blog](#)
- Visit [gallerify.me](#) and [smartam.net](#) to get started



The image shows a screenshot of the SmartAM Beta interface. At the top left, the logo 'smartam. beta' is displayed. Below it, a 4-step process is outlined:

- 1. Register data**
images, video, text files
- 2. Set permissions**
to display or to edit
- 3. Choose contacts**
friends, family, colleagues
- 4. Share it**
maintaining your privacy

To the right of the steps, a 3D white figure holds a large blue shield with a yellow arrow pointing up and the text 'smartam.' on it. Surrounding the figure are various file icons (PDF, Word, Excel, etc.) and social media icons (Facebook, Google+, etc.).

Below the steps, a paragraph reads: "Smart AM is a cutting edge user-managed access solution for sharing your files and resources throughout the net. It's you who decides what your colleagues, family and friends can view by applying particular privacy policies. Don't hesitate - try it out now. Your online data will never be safer!"

At the bottom right, there is a blue button with the Facebook 'f' logo and the text "Login with facebook".

In the bottom left corner of the interface, the Newcastle University logo is visible.

Fraunhofer AISEC project

- @@fill in
- @@links to relevant docs and materials
- @@status and next steps
- @@mention further EU plans?

Synergetics/TAS3 project

- @@fill in
- @@links to relevant docs and materials
- @@status and next steps

Agenda

Introduction: What is UMA, why do we need it, and how far along is it?

Use cases illustrating UMA's unique strengths

Why would an organization want to UMA-enable its apps?

Existing UMA implementations

How UMA works to build technical and business trust

Q&A

Here is UMA's history with OAuth

we're right about here

ProtectServe



1.0



1.0



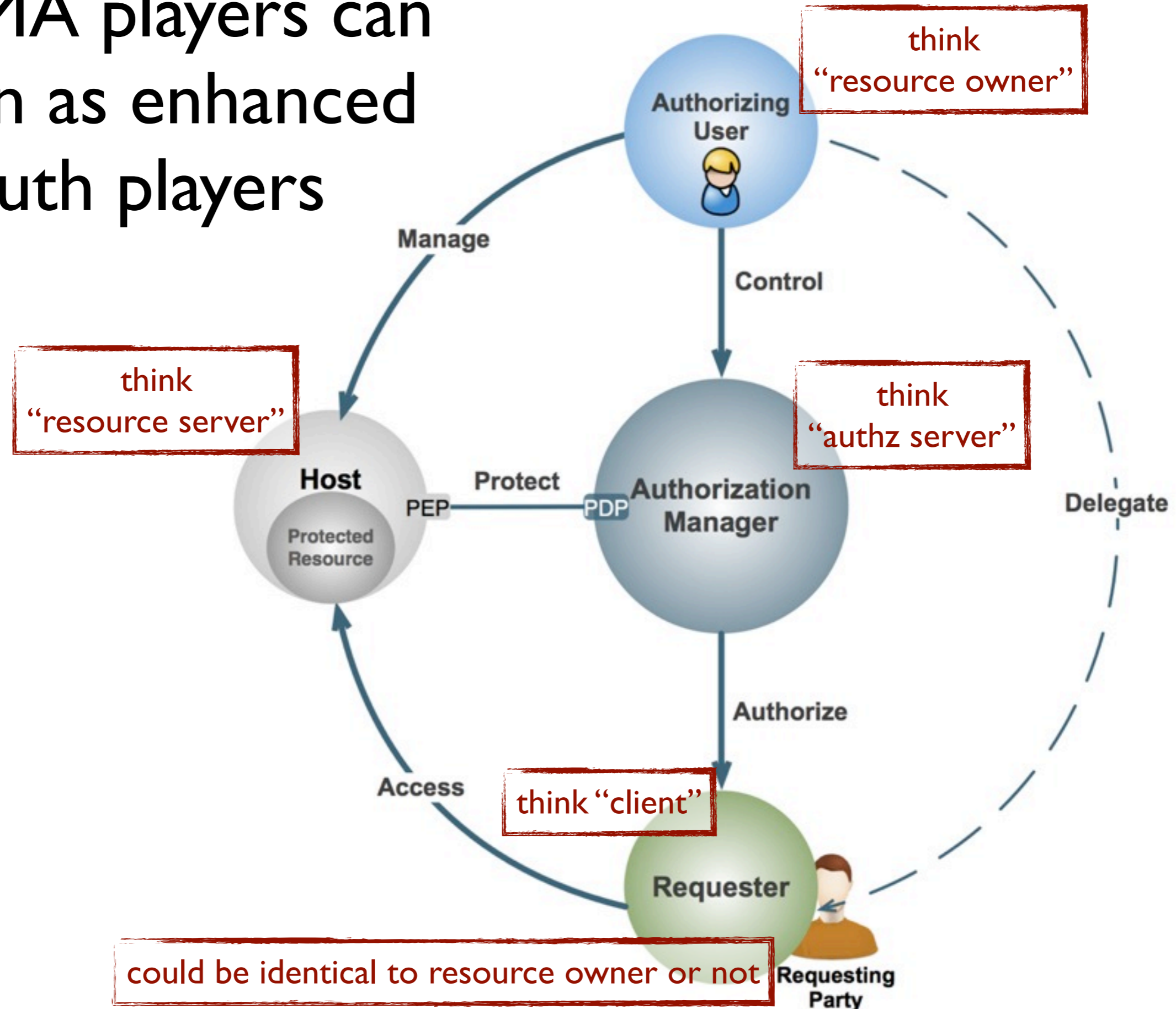
...



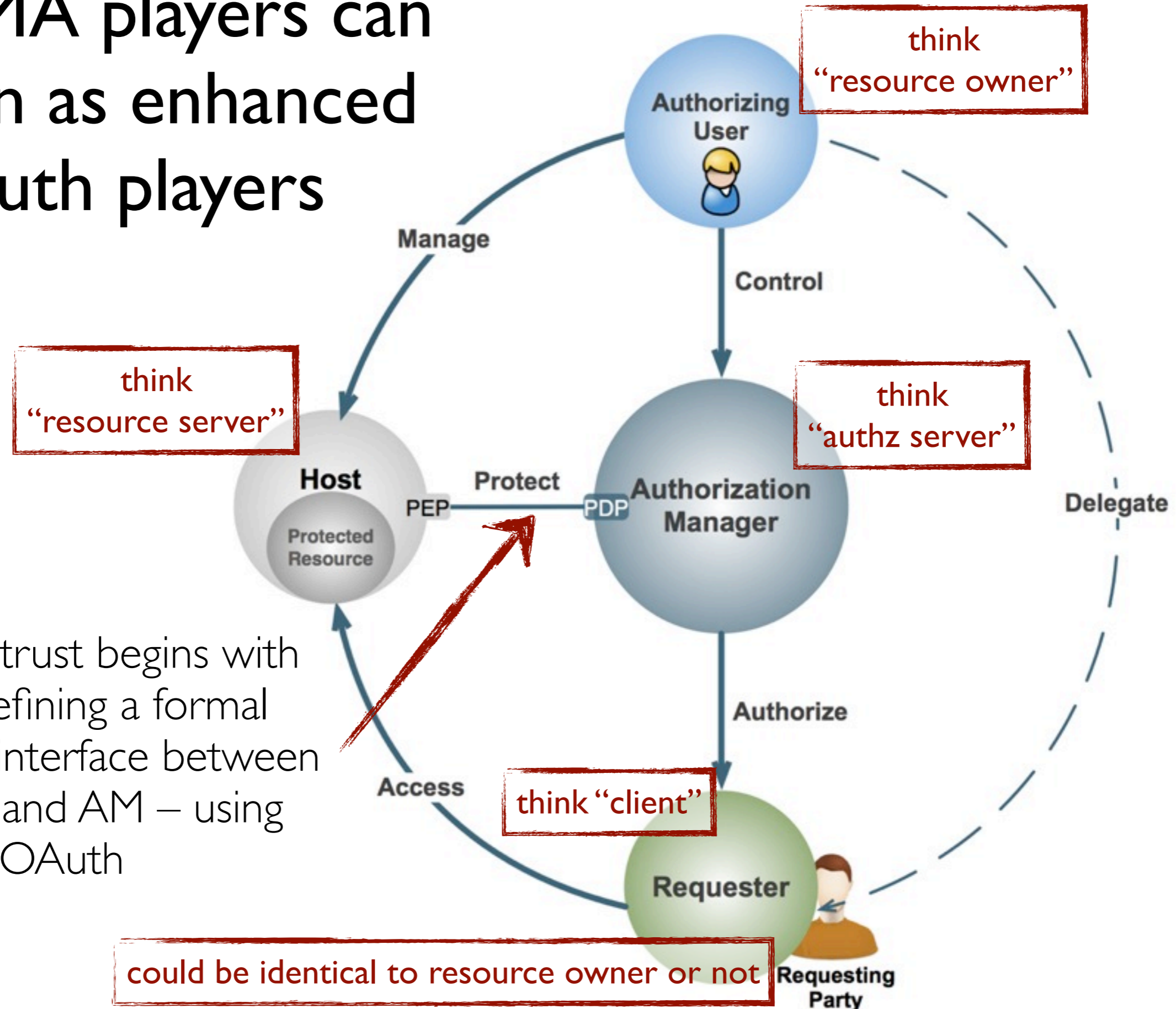
2.0



The UMA players can be seen as enhanced OAuth players



The UMA players can be seen as enhanced OAuth players



Technical trust begins with UMA defining a formal protected interface between the host and AM – using OAuth

could be identical to resource owner or not

By contrast, here is UMA's history with OpenID

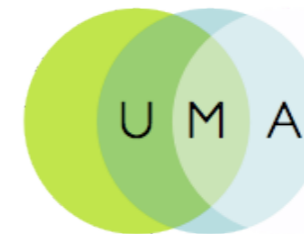
we're right about here



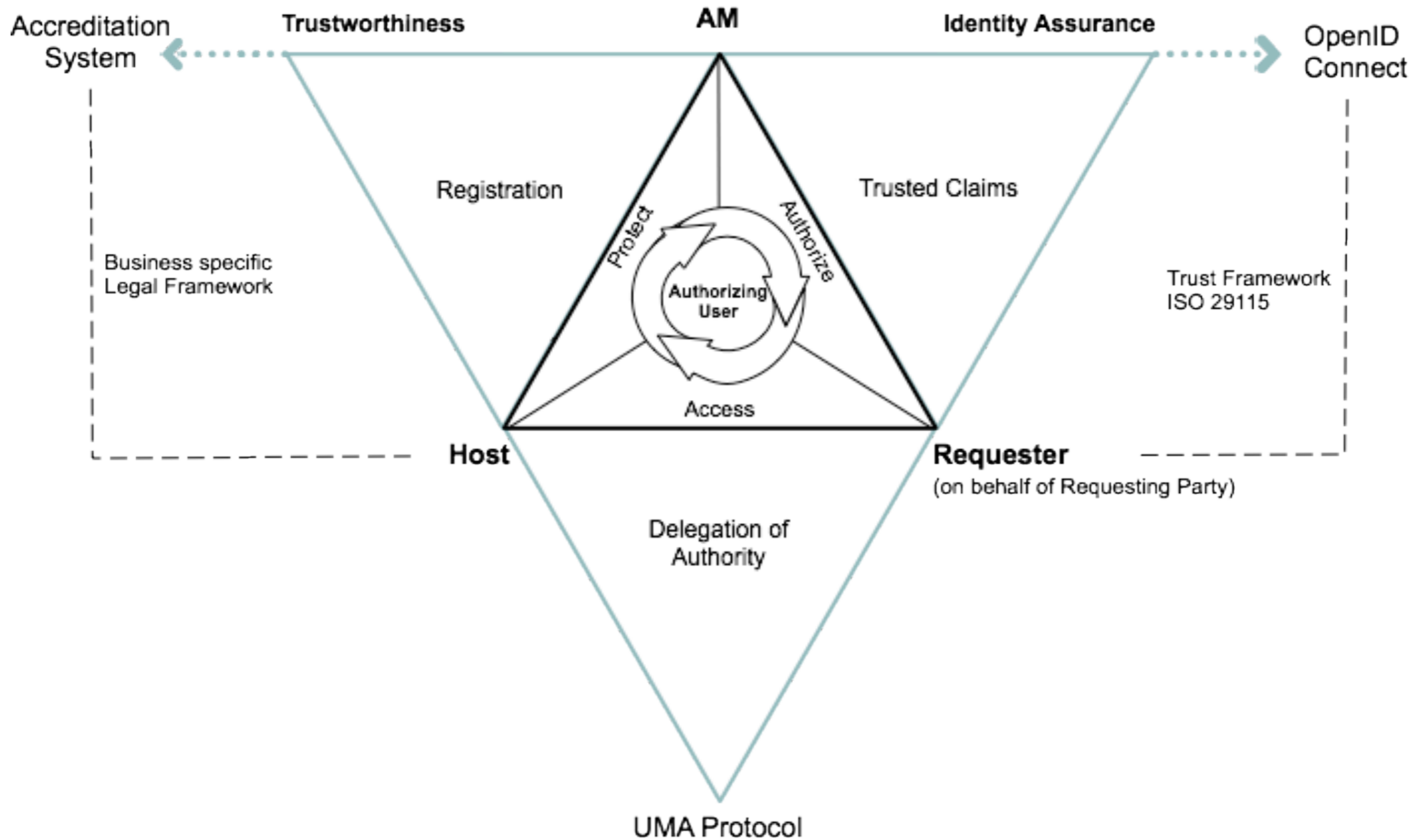
ProtectServe



...

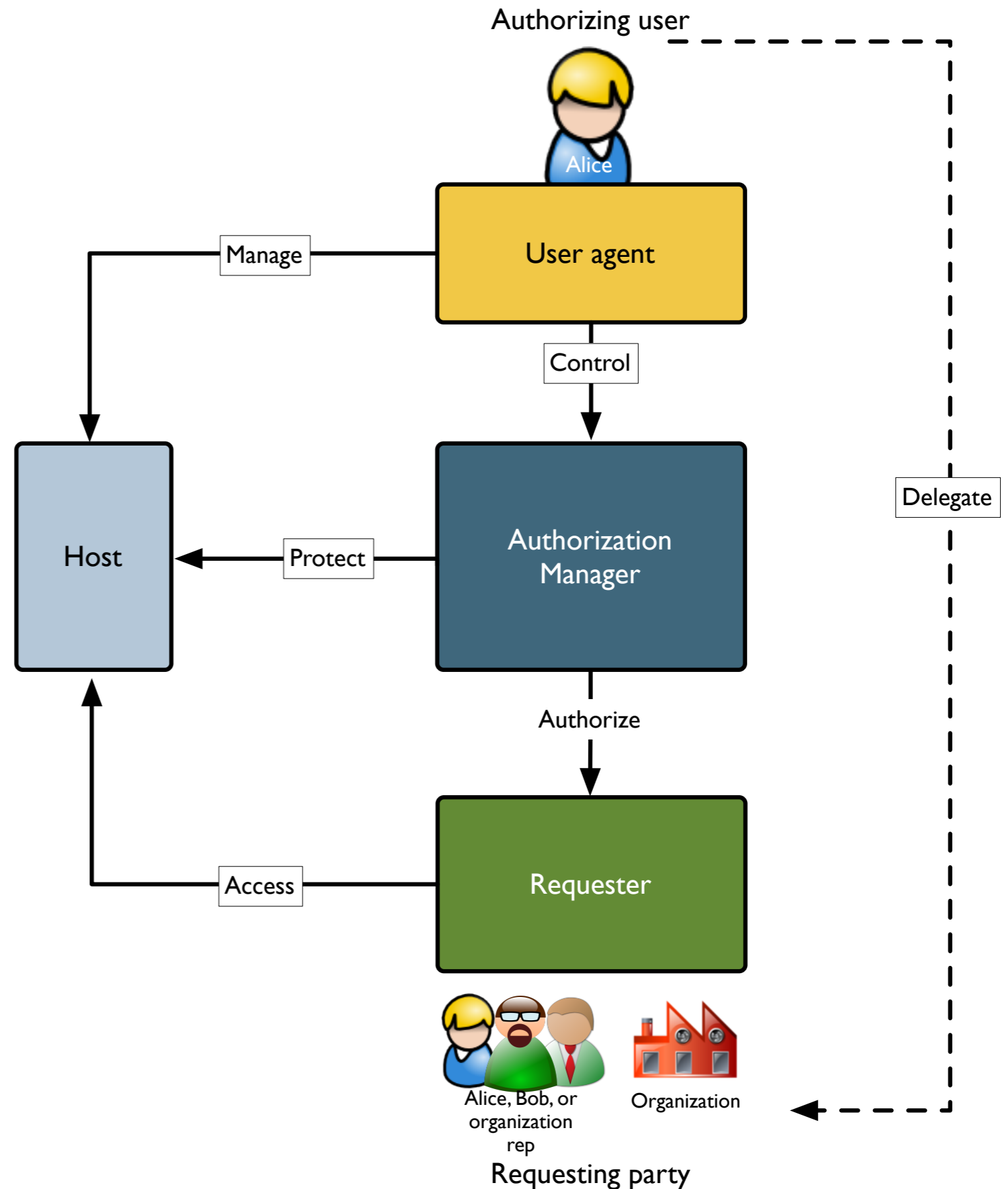


Business trust has many moving parts; claims-based authorization is one key



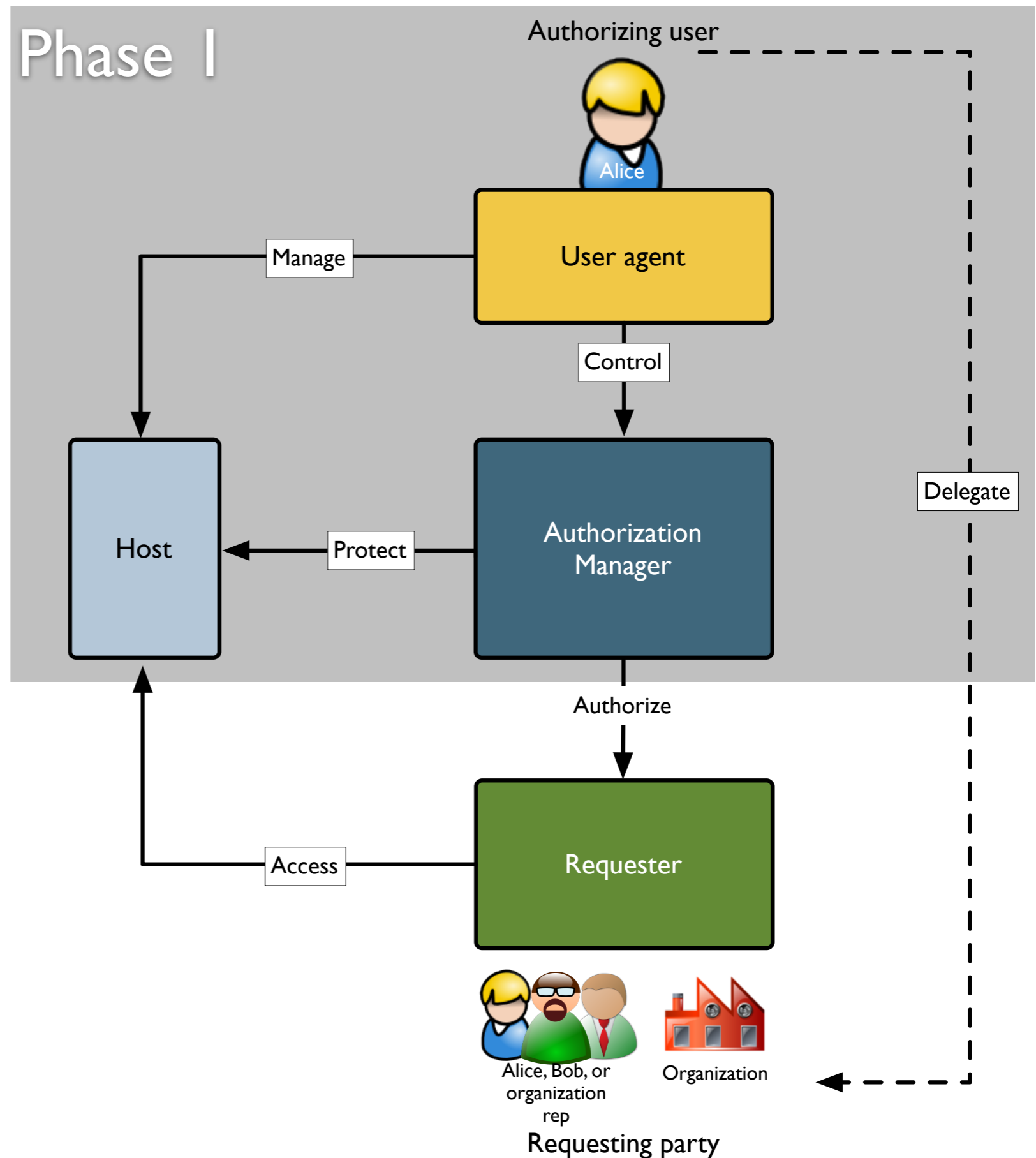
UMA has three phases

1. Protect a resource
2. Get authorization
3. Access a resource



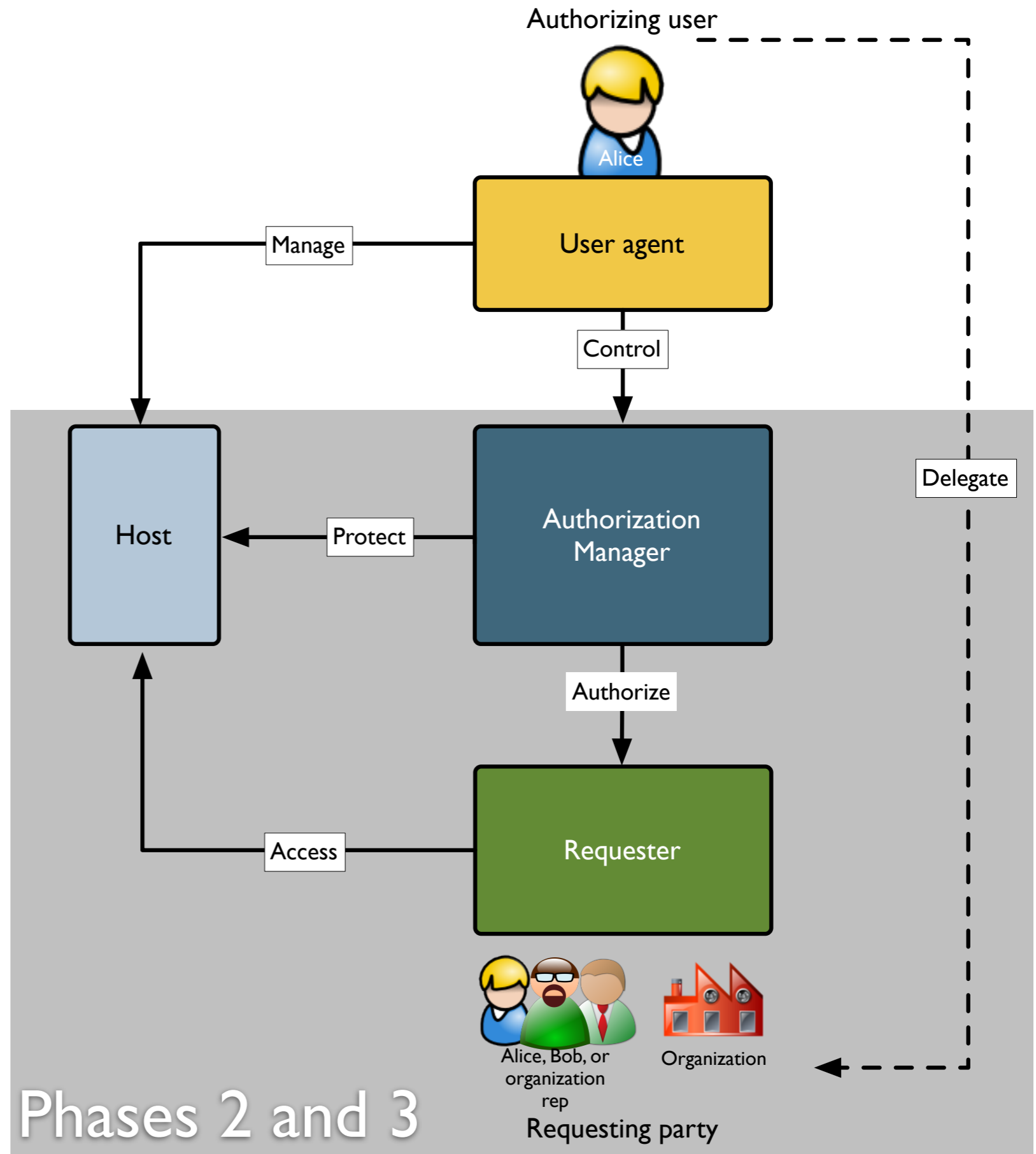
UMA has three phases

1. Protect a resource
2. Get authorization
3. Access a resource



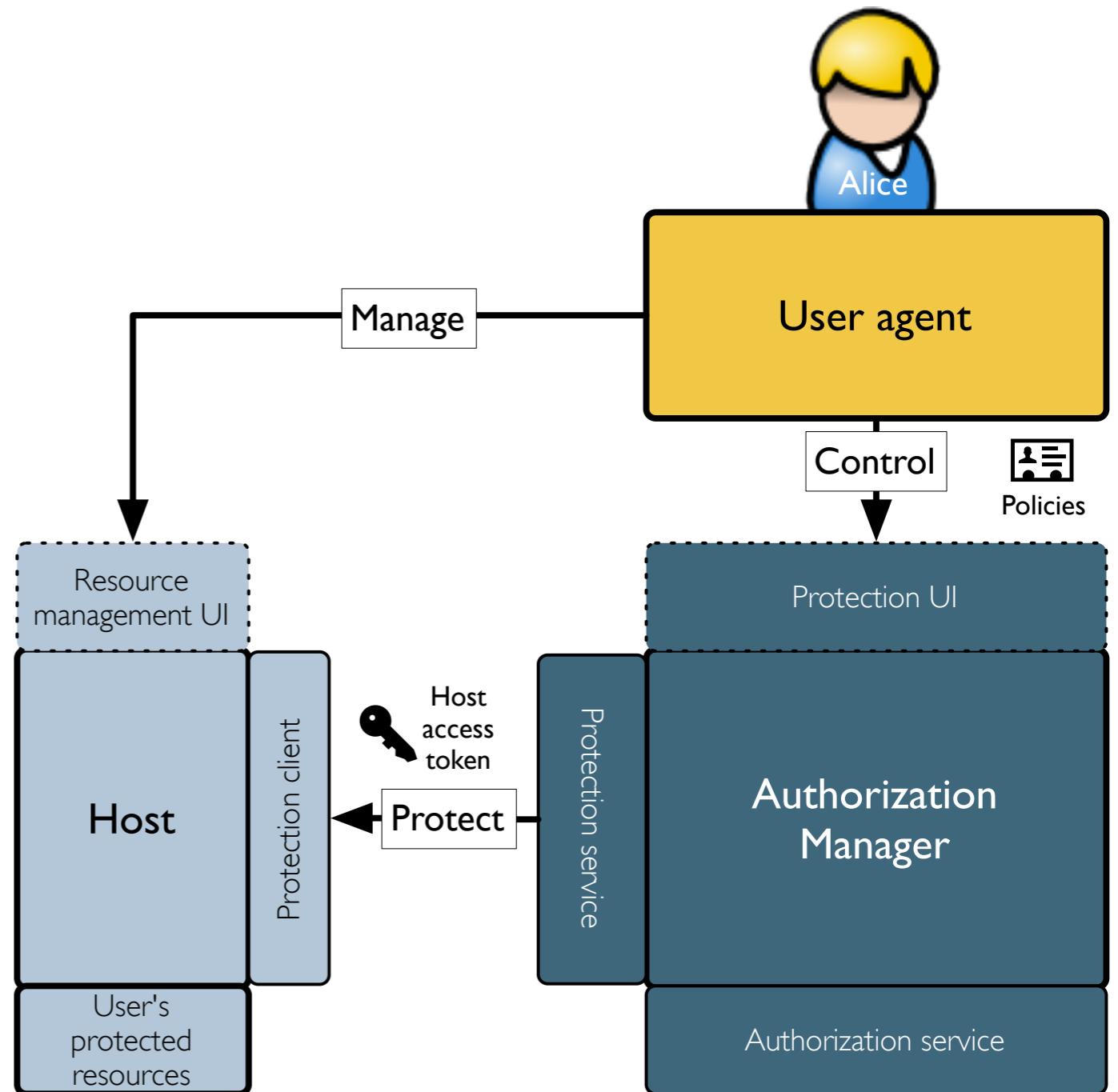
UMA has three phases

1. Protect a resource
2. Get authorization
3. Access a resource



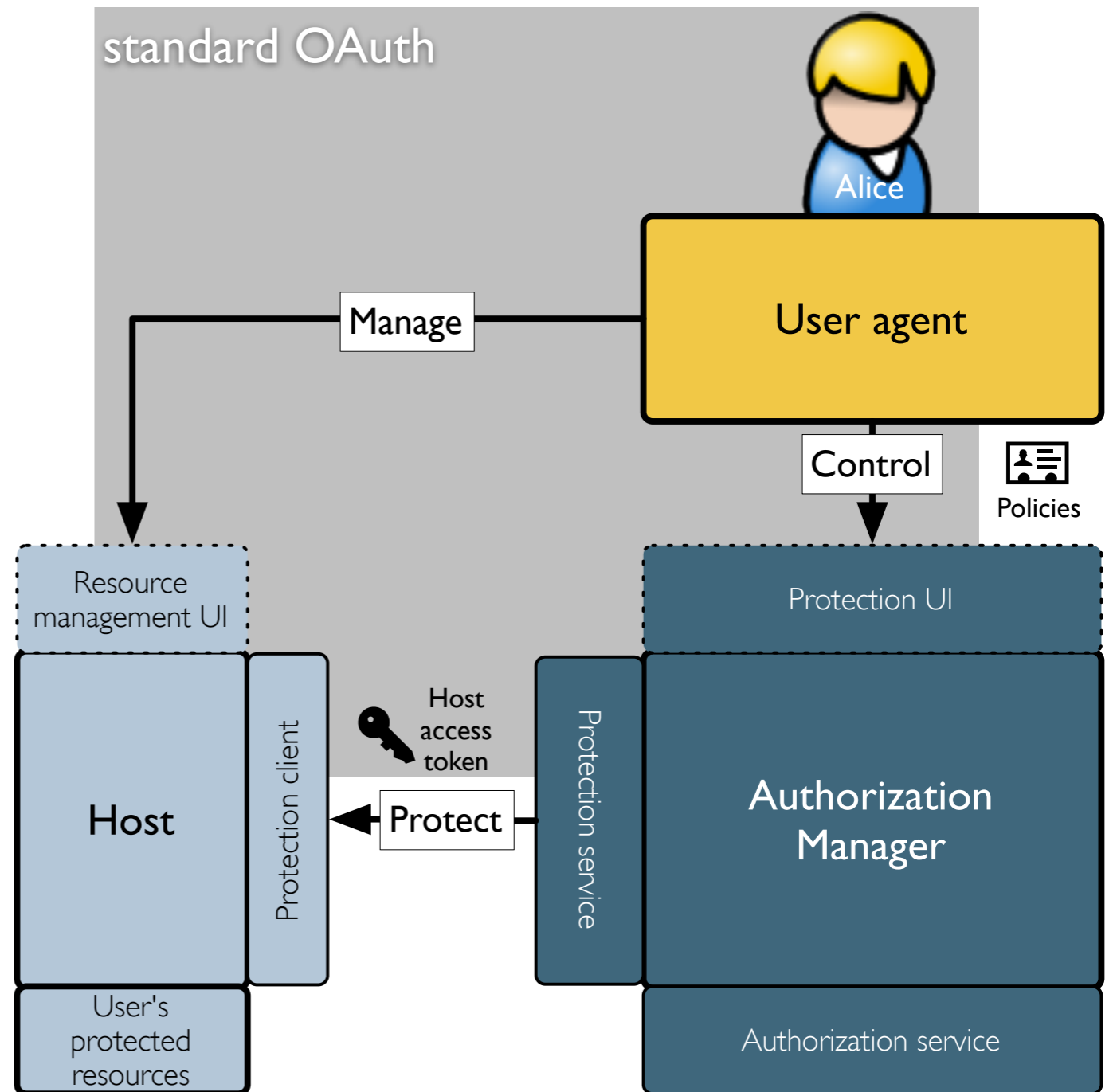
Phase I: Protect a resource

- Alice introduces host and AM using OAuth
 - Possibly with dynamic registration
- Host registers **sets of resources** to be protected and **available scopes** at AM *host resource set registration endpoint*
- Alice ensures AM knows her policies for sharing them



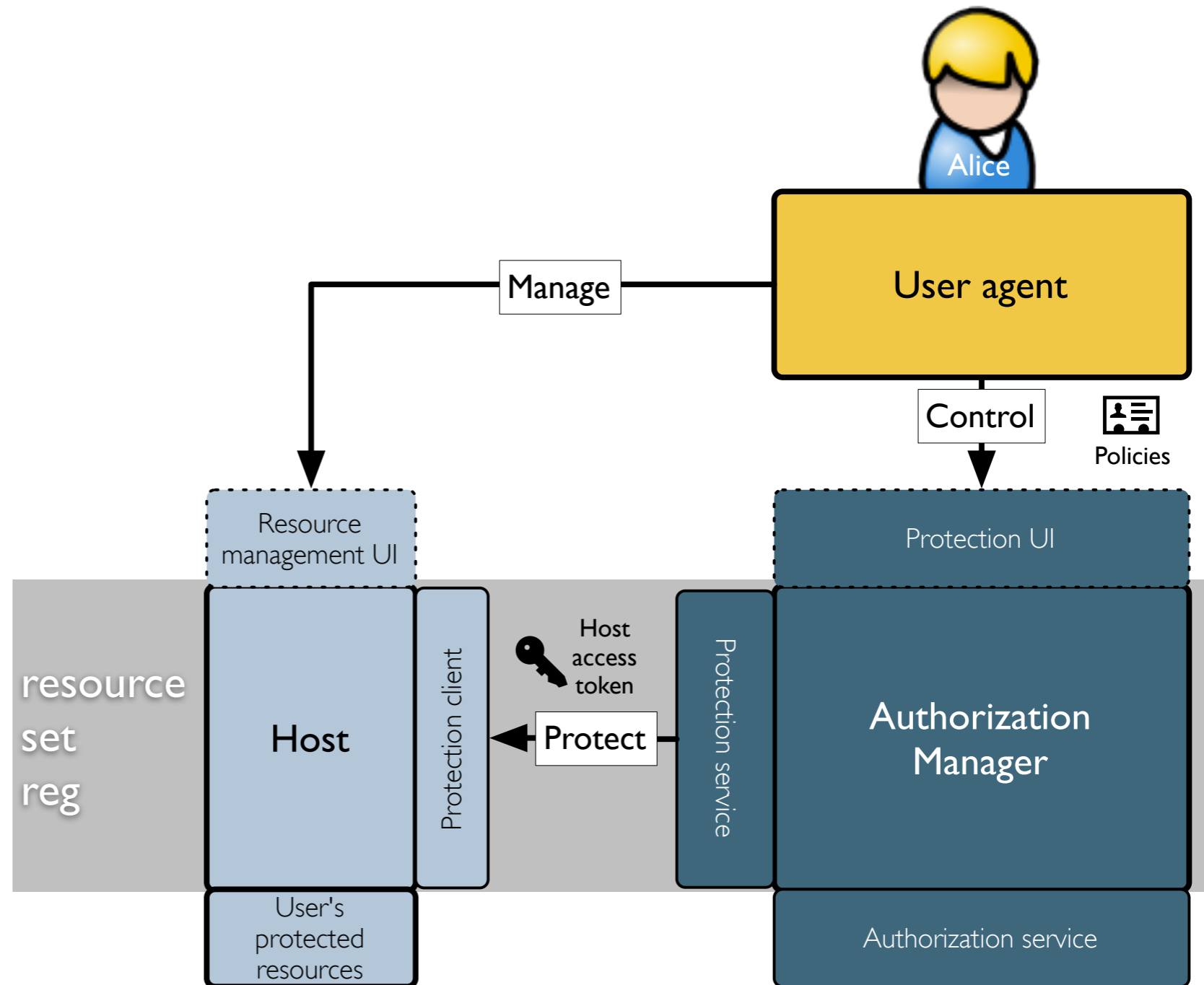
Phase I: Protect a resource

- Alice introduces host and AM using OAuth
- Possibly with dynamic registration
- Host registers **sets of resources** to be protected and **available scopes** at AM *host resource set registration endpoint*
- Alice ensures AM knows her policies for sharing them



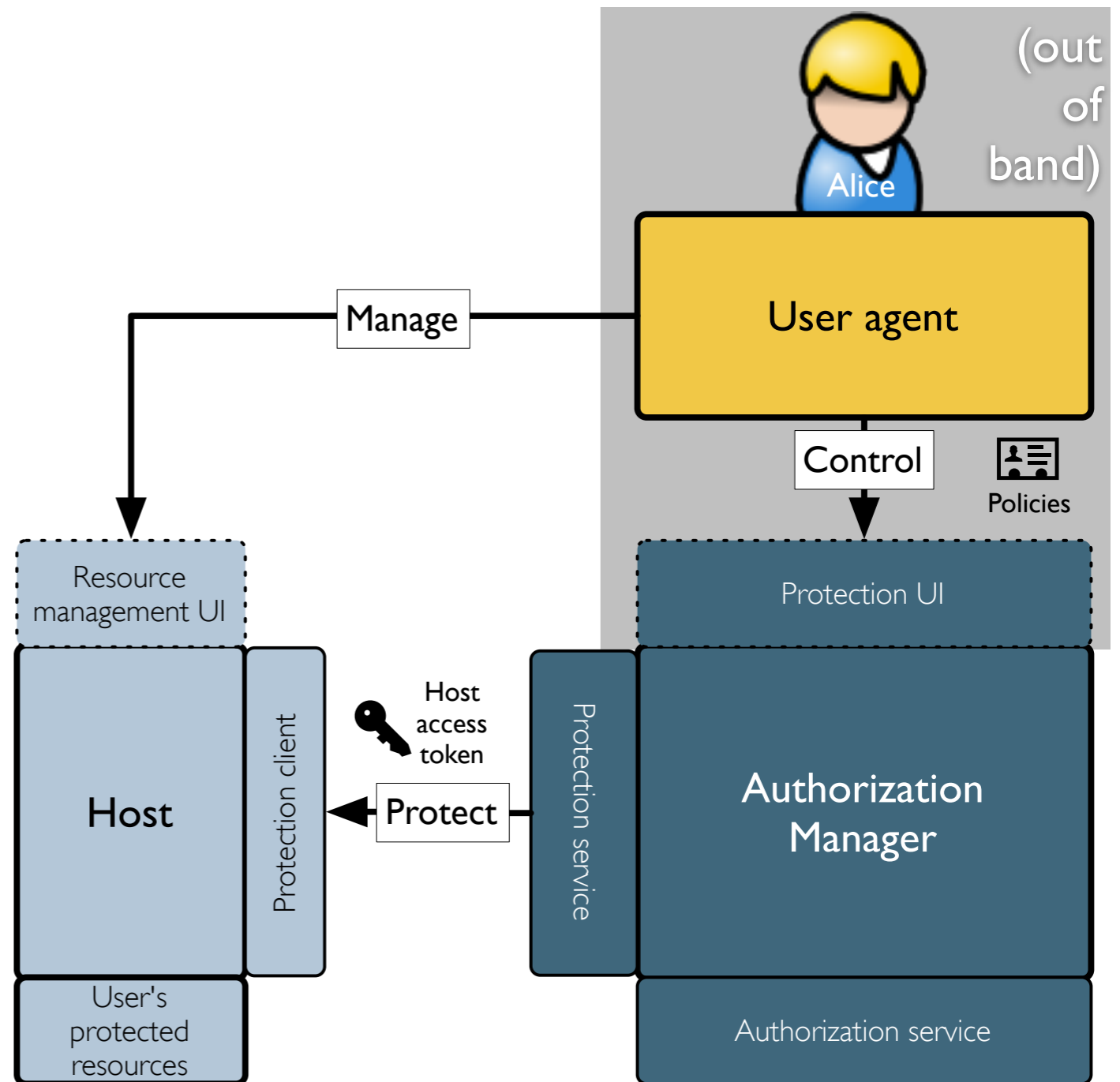
Phase I: Protect a resource

- Alice introduces host and AM using OAuth
 - Possibly with dynamic registration
- Host registers **sets of resources** to be protected and **available scopes** at AM *host resource set registration endpoint*
- Alice ensures AM knows her policies for sharing them



Phase I: Protect a resource

- Alice introduces host and AM using OAuth
 - Possibly with dynamic registration
- Host registers **sets of resources** to be protected and **available scopes** at AM *host resource set registration endpoint*
- Alice ensures AM knows her policies for sharing them



Working with resource set registration and scopes @@update

- Scope URIs resolve to scope descriptions
 - They can live anywhere
- Host registers resource sets and maps to available scopes
 - Using RESTful API

```
{
  "scope":
  {
    "_id": "view"
    "name": "View Photo and Related Info",
    "icon_uri": "http://www.example.com/icons/reading-glasses.png"
  }
}
```

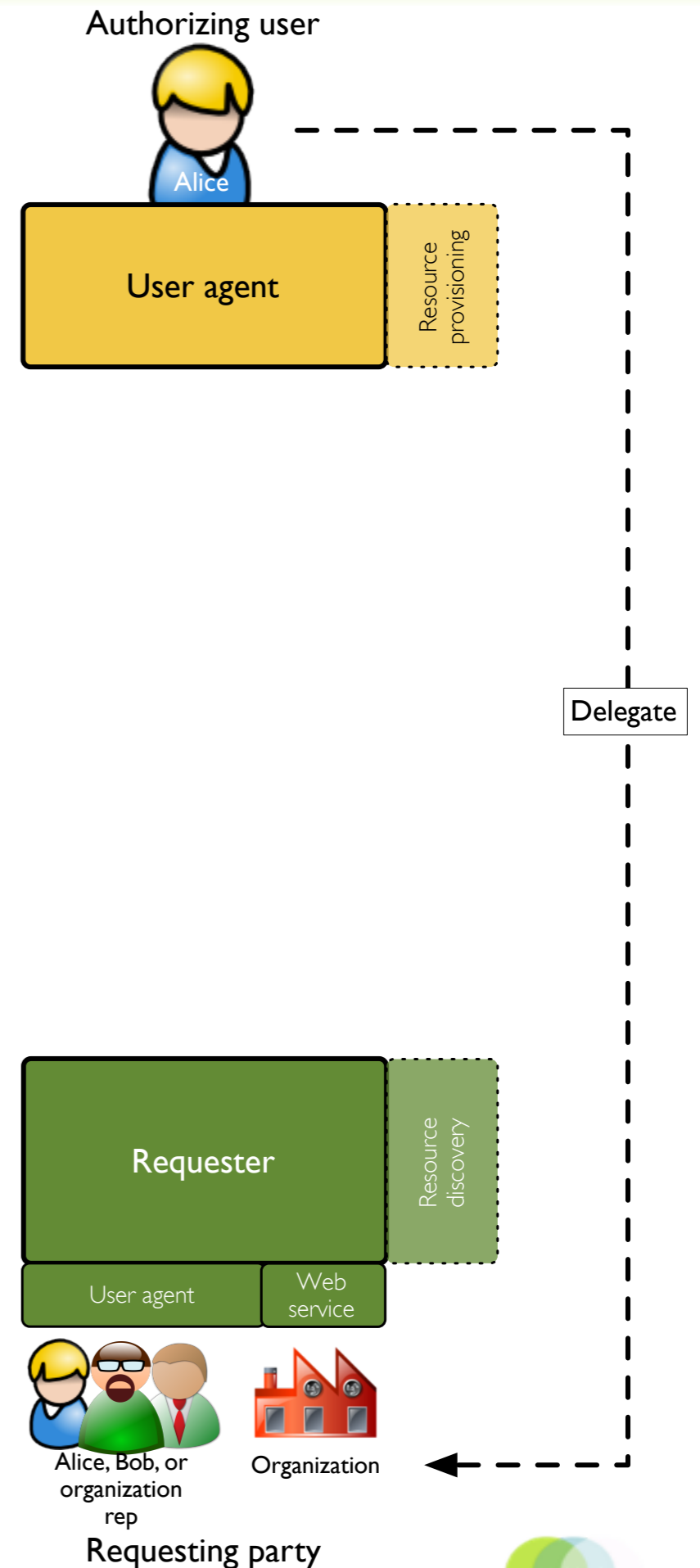
```
{
  "scope":
  {
    "_id": "all"
    "name": "All Actions",
    "icon_uri": "http://www.example.com/icons/galaxy.png"
  }
}
```

```
{
  "resource_set":
  {
    "_id": "112210f47de98100"
    "name": "Steve the puppy!",
    "icon_uri": "http://www.example.com/icons/flower",
    "scopes":
      ["http://photoz.example.com/dev/scopes/view",
       "http://photoz.example.com/dev/scopes/all"]
  }
}
```

```
PUT /host/photoz.example.com/resource_set/112210f47de98100 HTTP/1.1
Content-Type: application/json
...
```

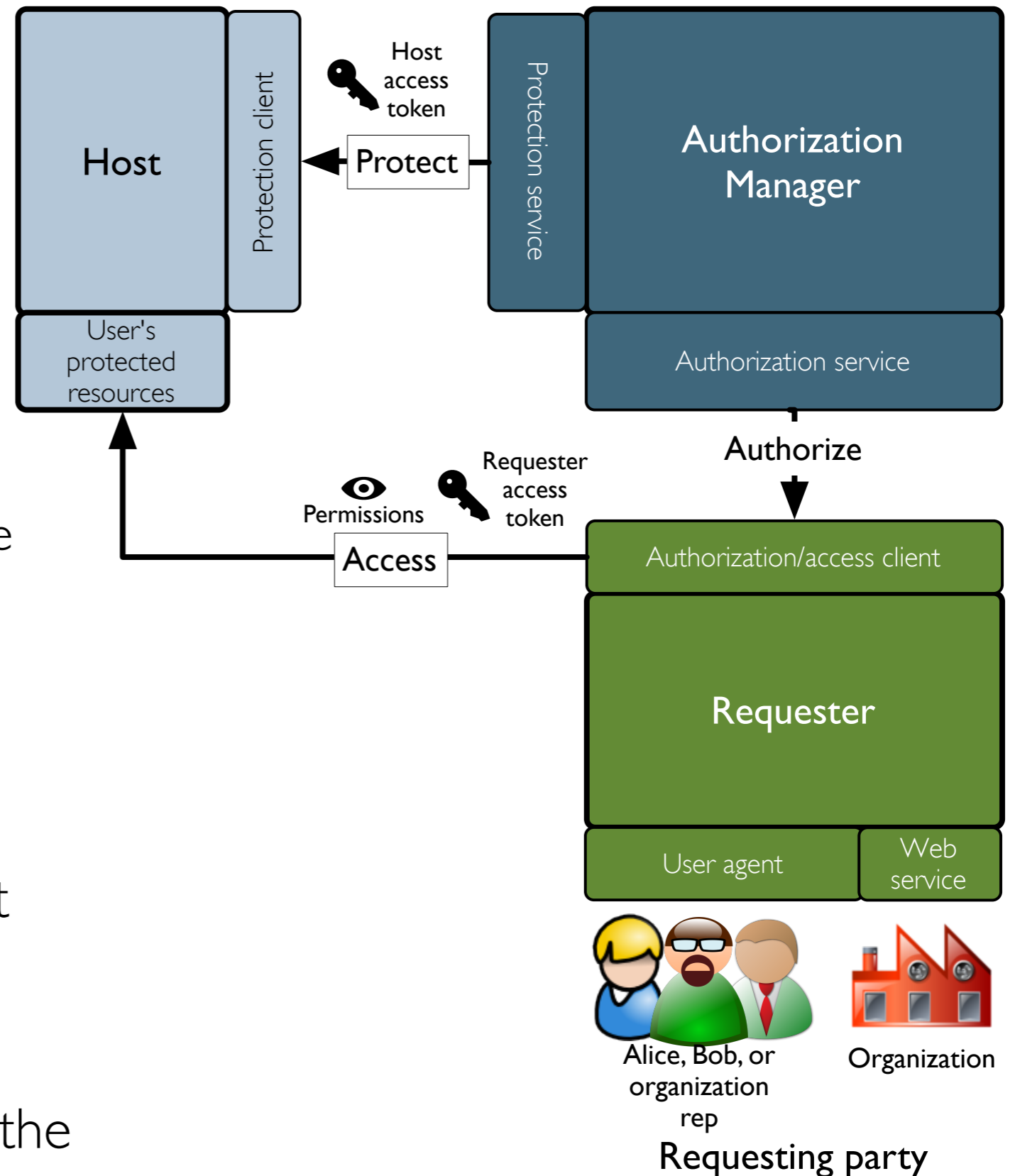

(intermission)

- The requesting party learns about the resource... *somehow*
 - Emailed link?
 - Discovery service?
 - Microformat data on Alice's blog?
- And it knows how to use the API and scopes at the host...*somehow*
 - Developer documentation?
 - Standardized scopes?



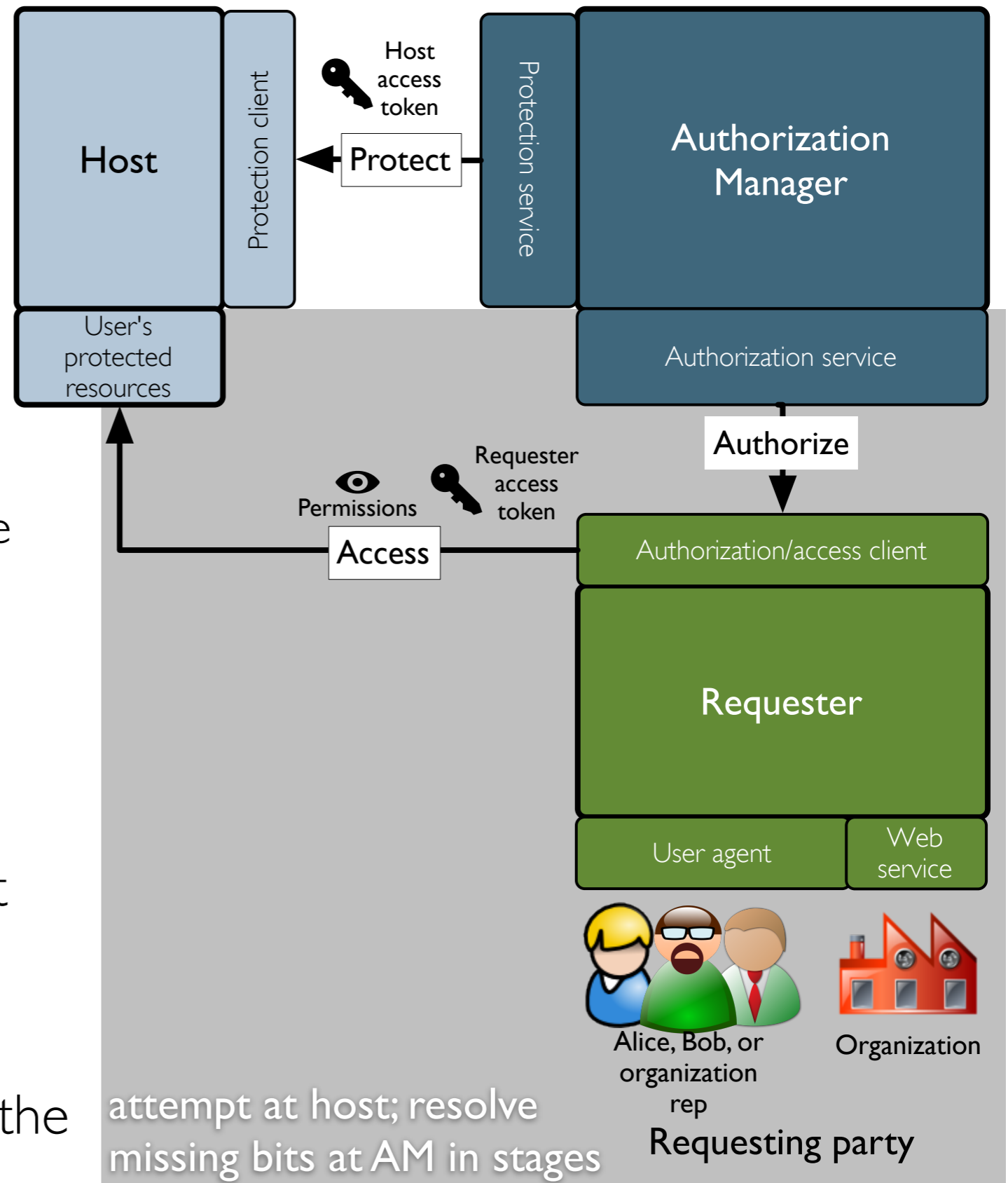
Phase 2: Get authorization

- Requester attempts access but has to get, in turn...
 - A token from AM *requester token endpoint*
 - Permission for sought-after scope from AM *authorization endpoint*
 - Likely providing *claims* to win permission
 - Host uses AM *token status endpoint* to check each attempt by requester
 - Host uses AM *permission registration endpoint* to register the sought-after scope
- tinyurl.com/umawg



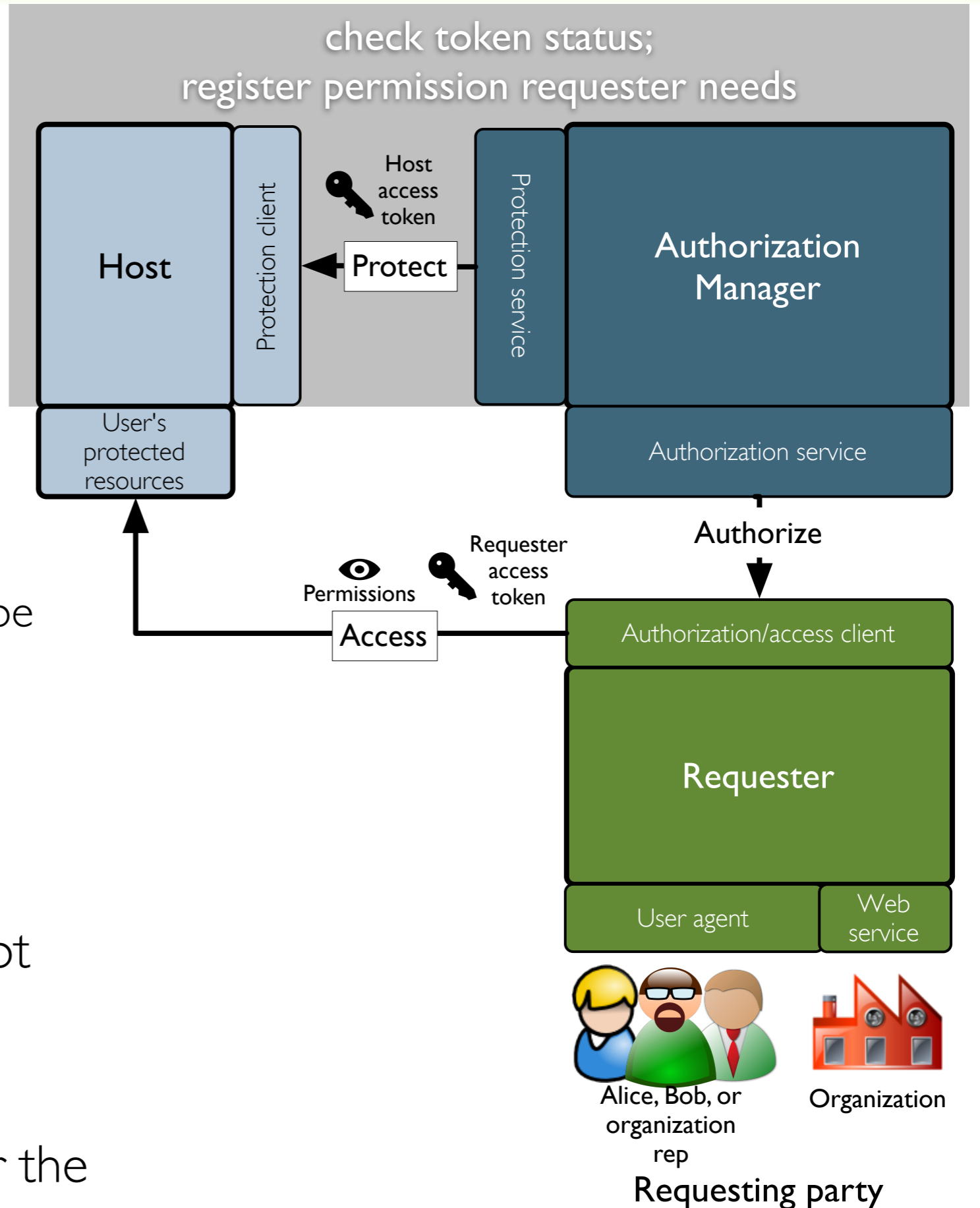
Phase 2: Get authorization

- Requester attempts access but has to get, in turn...
 - A token from AM *requester token endpoint*
 - Permission for sought-after scope from AM *authorization endpoint*
 - Likely providing *claims* to win permission
 - Host uses AM *token status endpoint* to check each attempt by requester
 - Host uses AM *permission registration endpoint* to register the sought-after scope
- tinyurl.com/umawg



Phase 2: Get authorization

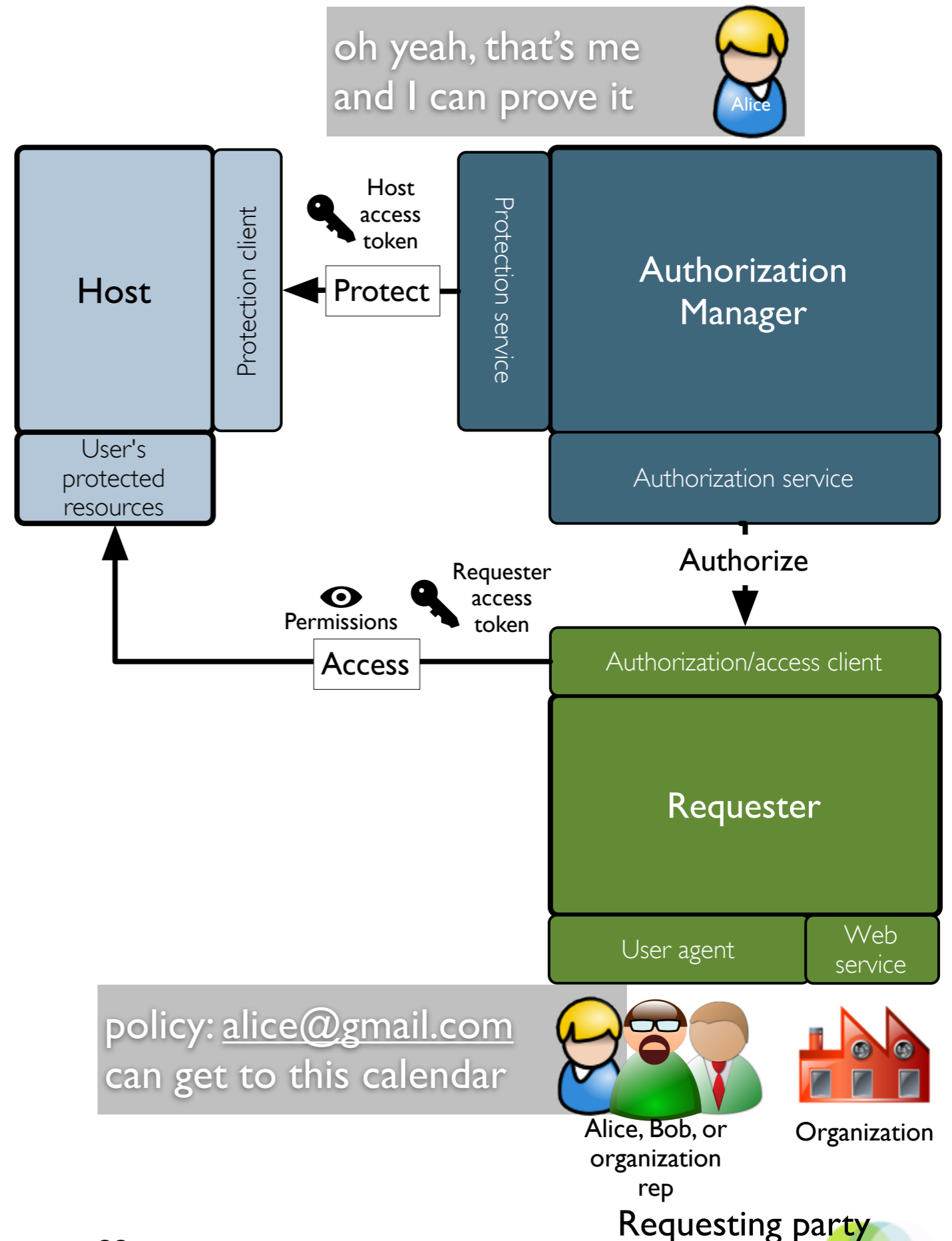
- Requester attempts access but has to get, in turn...
 - A token from AM requester token endpoint
 - Permission for sought-after scope from AM authorization endpoint
 - Likely providing *claims* to win permission
- Host uses AM token status endpoint to check each attempt by requester
- Host uses AM permission registration endpoint to register the sought-after scope



If Alice is *also* the requesting party...

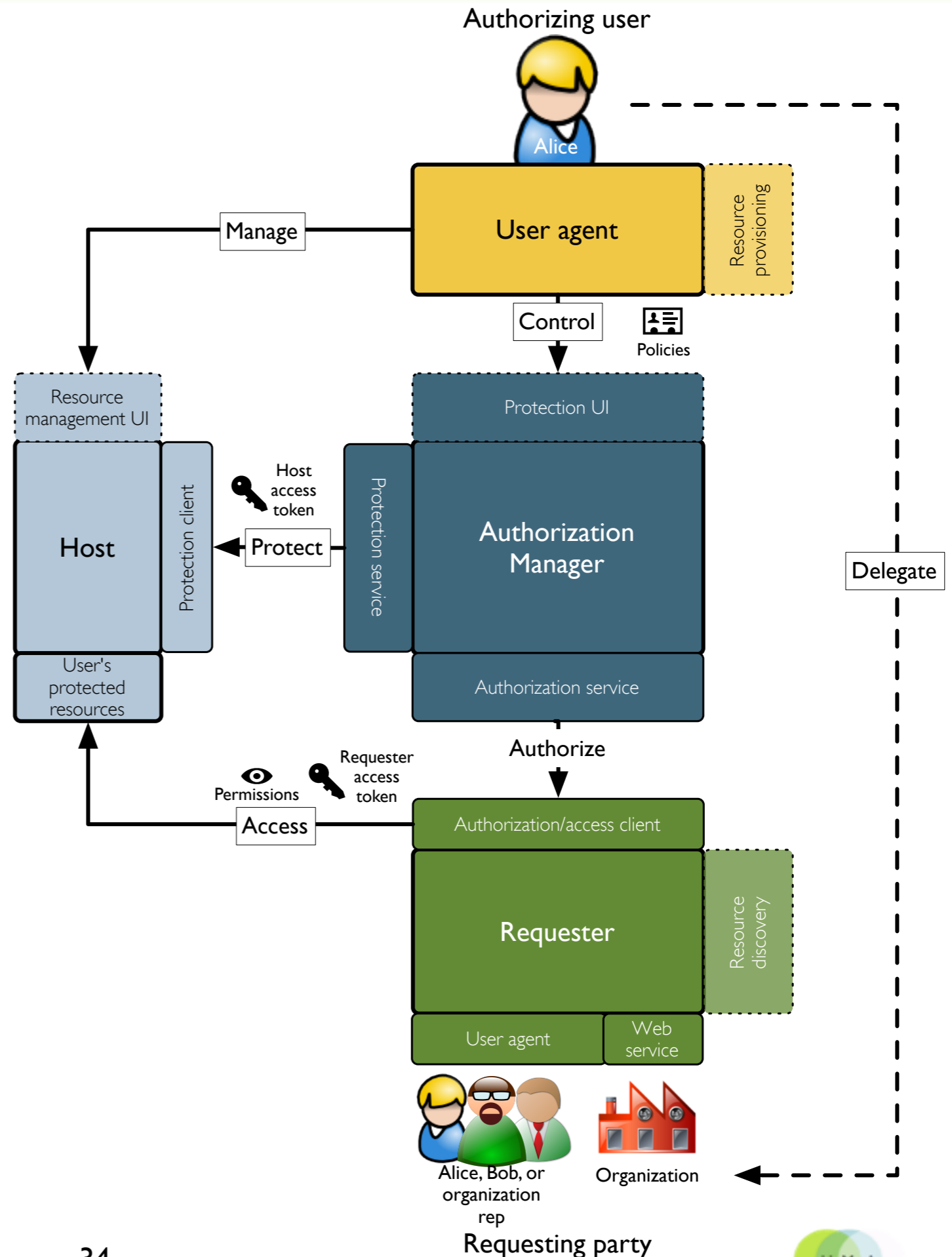
- She has a “synchronous” authorization experience because the claim she must provide is that she’s Alice
- The flow would be the same for Alice, Bob, or anyone else who needs to prove they satisfy the policy
- We are working on OpenID Connect integration for basic interoperable “trusted claims”

tinyurl.com/umawg



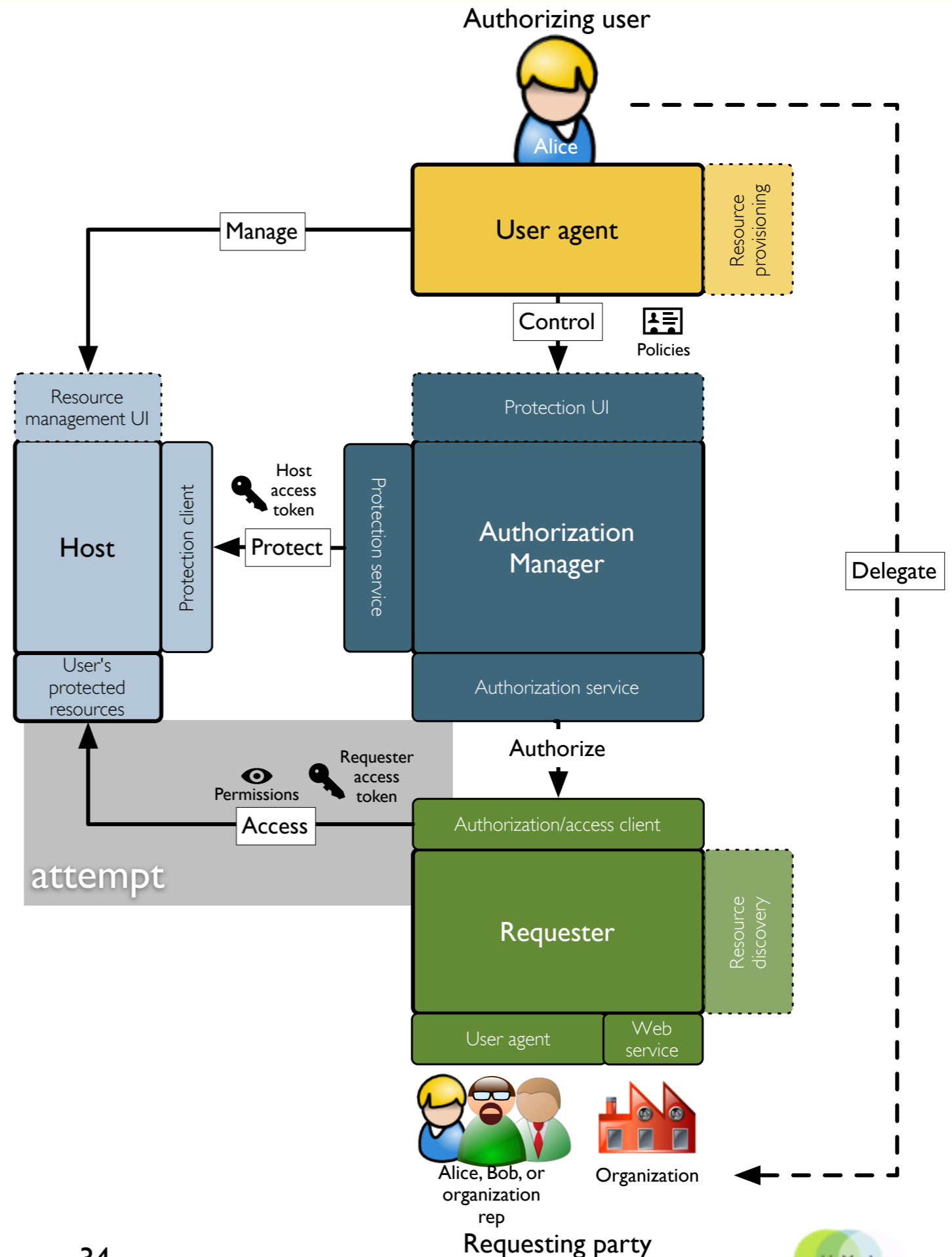
Phase 3: Access a resource

- Requester attempts access with a valid token associated with sufficient scope
- Host gets the permissions associated with it at AM's *token status endpoint*
- Host lets requester succeed in attempt



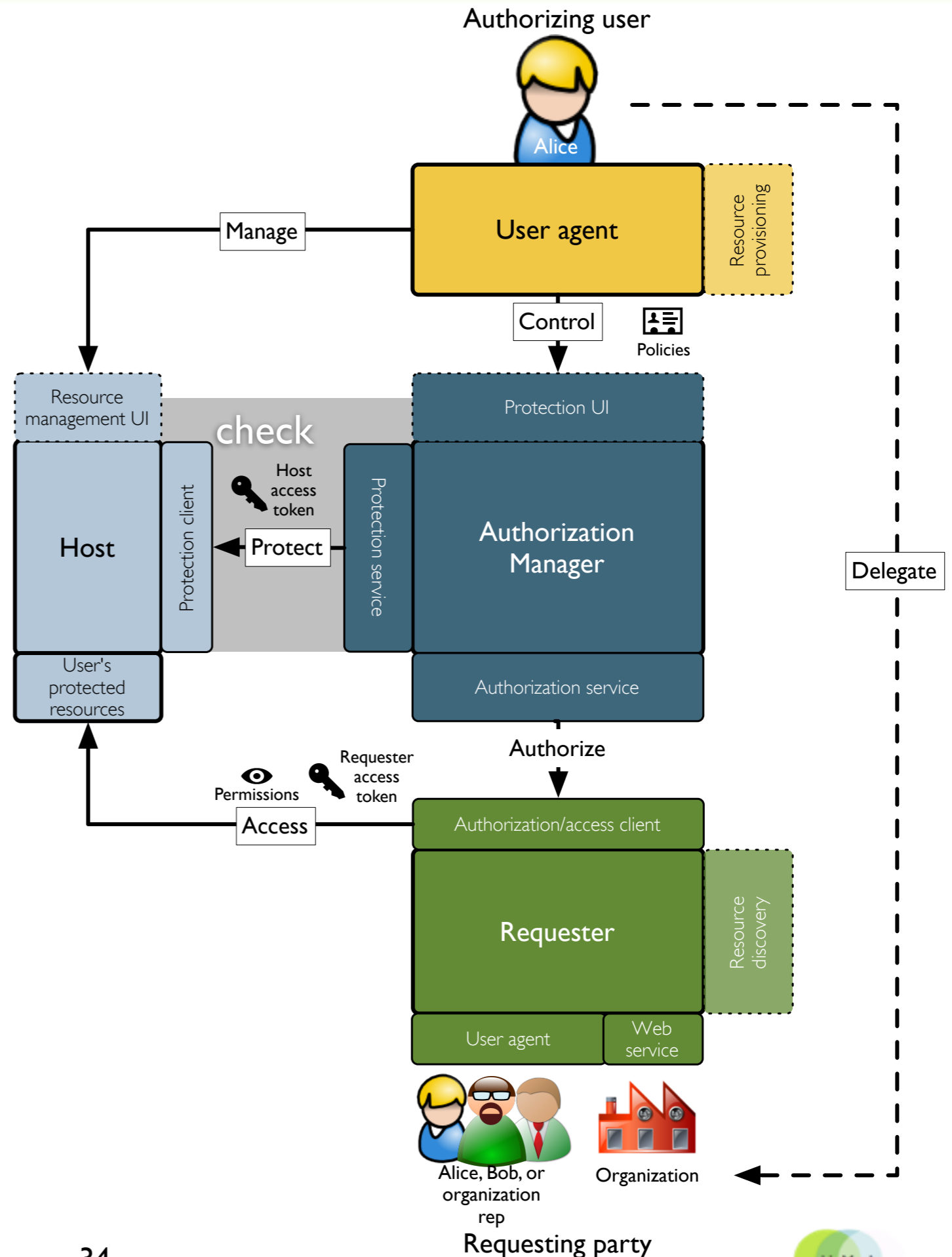
Phase 3: Access a resource

- Requester attempts access with a valid token associated with sufficient scope
- Host gets the permissions associated with it at AM's *token status endpoint*
- Host lets requester succeed in attempt



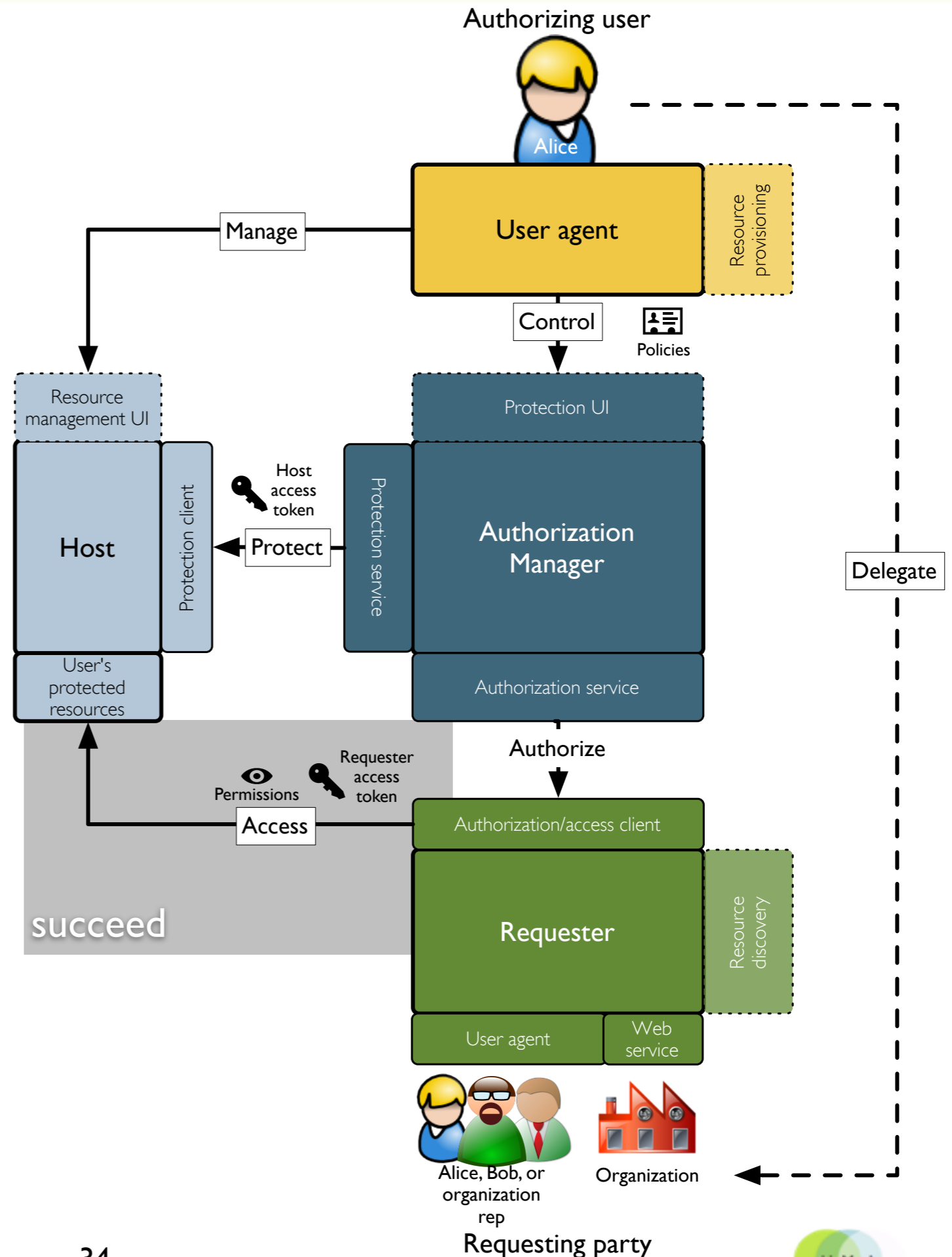
Phase 3: Access a resource

- Requester attempts access with a valid token associated with sufficient scope
- Host gets the permissions associated with it at AM's *token status endpoint*
- Host lets requester succeed in attempt

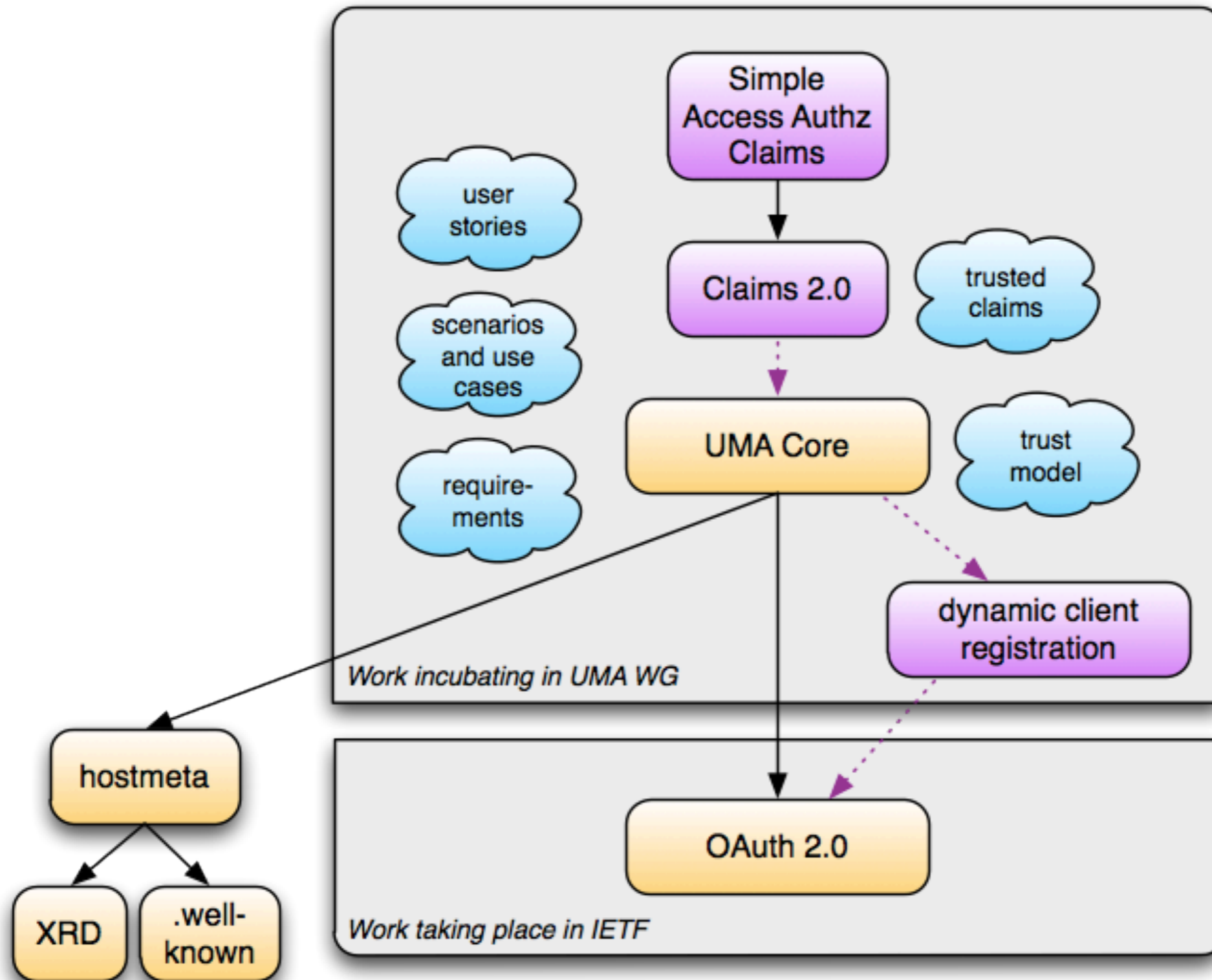


Phase 3: Access a resource

- Requester attempts access with a valid token associated with sufficient scope
- Host gets the permissions associated with it at AM's *token status endpoint*
- Host lets requester succeed in attempt



The UMA spec “call tree” @@update



29 May 2011

Agenda

Introduction: What is UMA, why do we need it, and how far along is it?

Use cases illustrating UMA's unique strengths

Why would an organization want to UMA-enable its apps?

Existing UMA-conforming implementations

How UMA works to build technical and business trust

Q&A

Thanks for joining us today

Become an UMANitarian!

Webinar recording will appear soon!

Visit <http://tinyurl.com/umawg>

Eve Maler on behalf of and with thanks to the UMA
Work Group
14 December 2011

