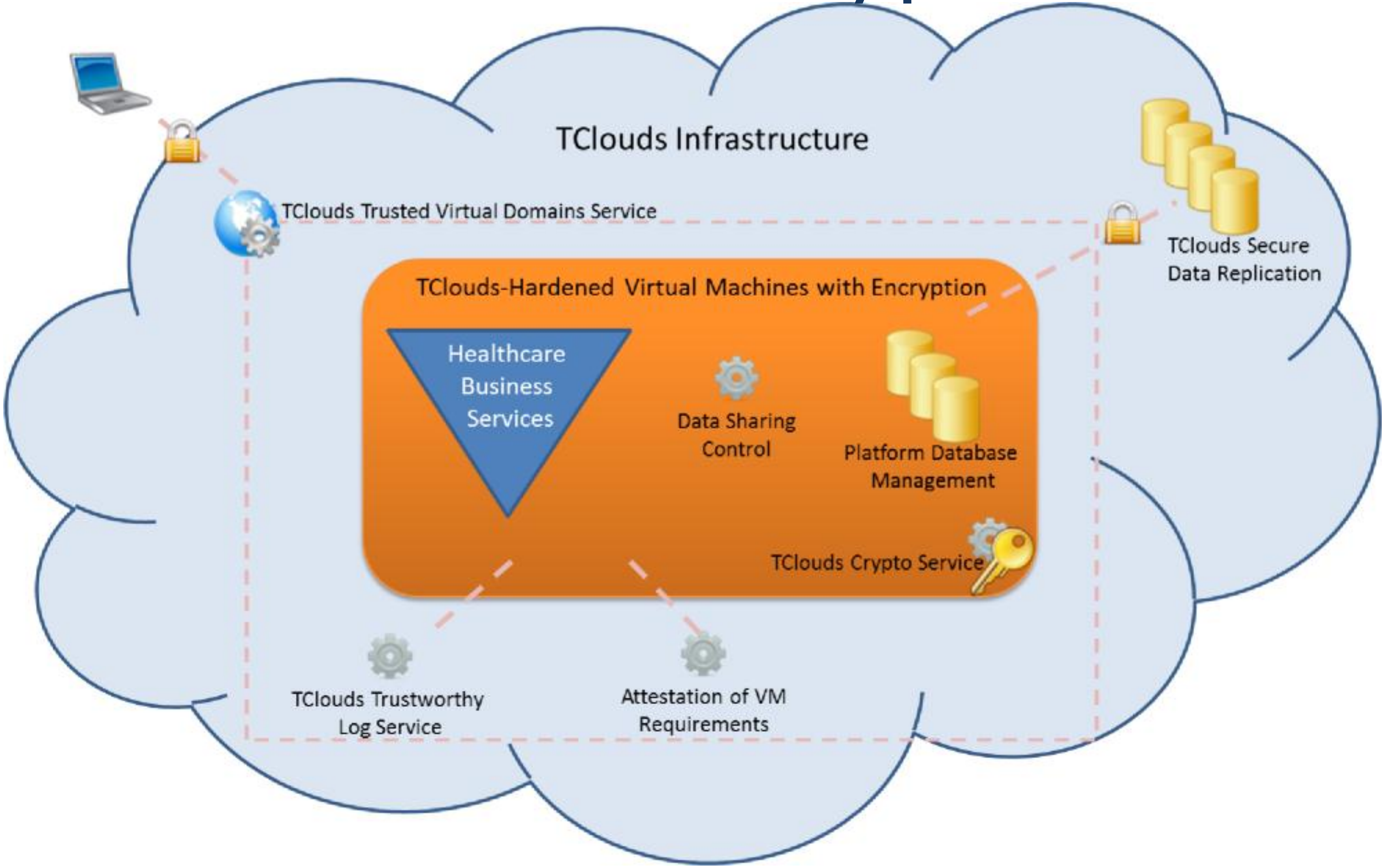
Rights Delegation in the Trustworthy Healthcare Platform

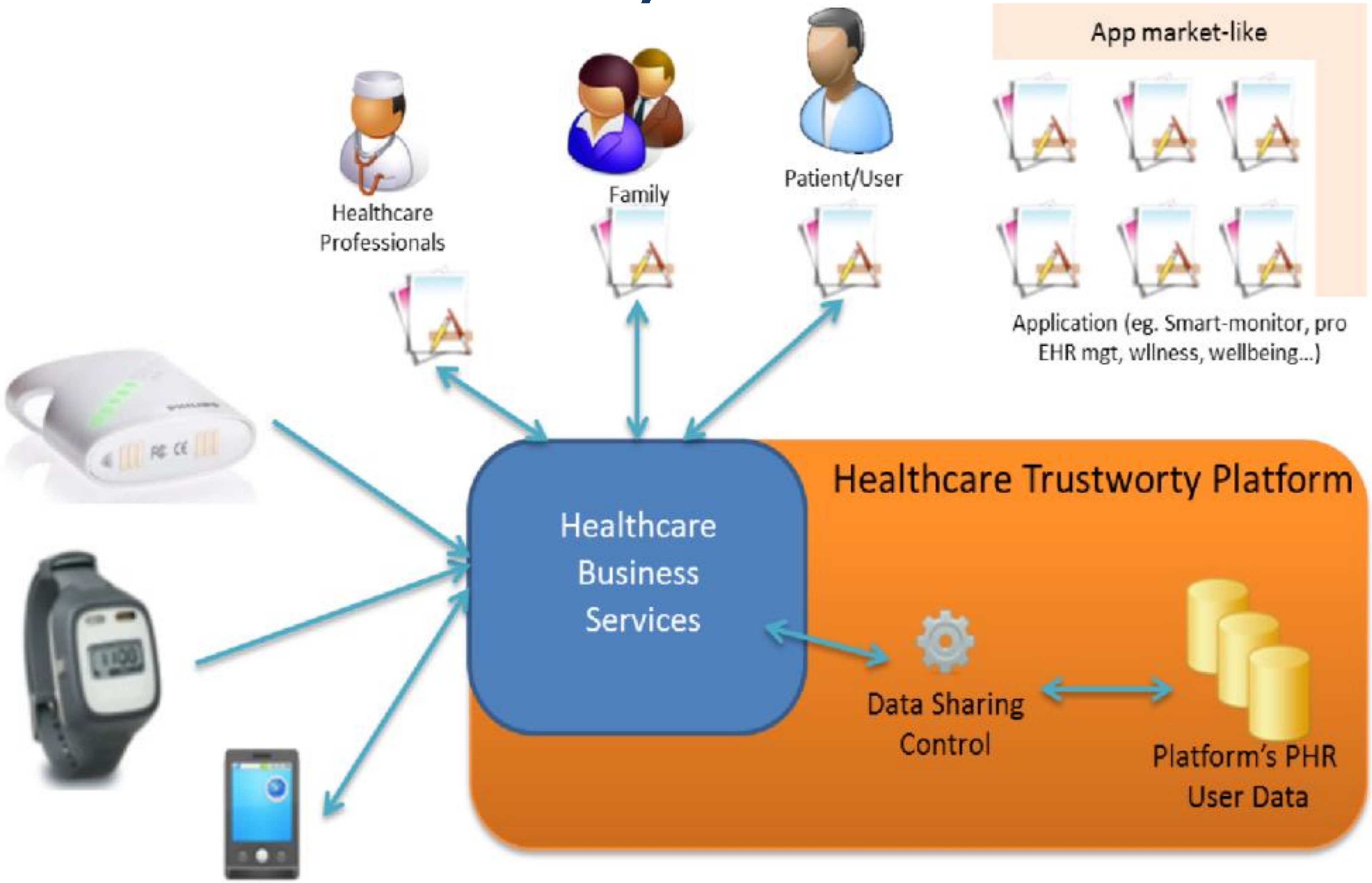
healthcare trustworthy platform



characteristics

- Privacy Policies, End-User‘s Responsibility & Third Parties
- Secure In-Transit Data
- Legal Compliances
- Trusted Audits and Log
- Security on Commodity Clouds
- Secure Data Storage
- Geolocalization of EHR Databases
- REST interfaces to support applications

eco system


App market-like
Healthcare Professionals
Family
Patient/User
Application (eg. Smart-monitor, pro EHR mgt, wllness, wellbeing...)
Healthcare Trustworty Platform
Healthcare Business Services
Data Sharing Control
Platform's PHR User Data

requirements

- patient is in control of his or her personal health data (includes EHR)

- hospital personal needs access e.g.
  - doctor to add and read data
  - administrative personal to make appointments

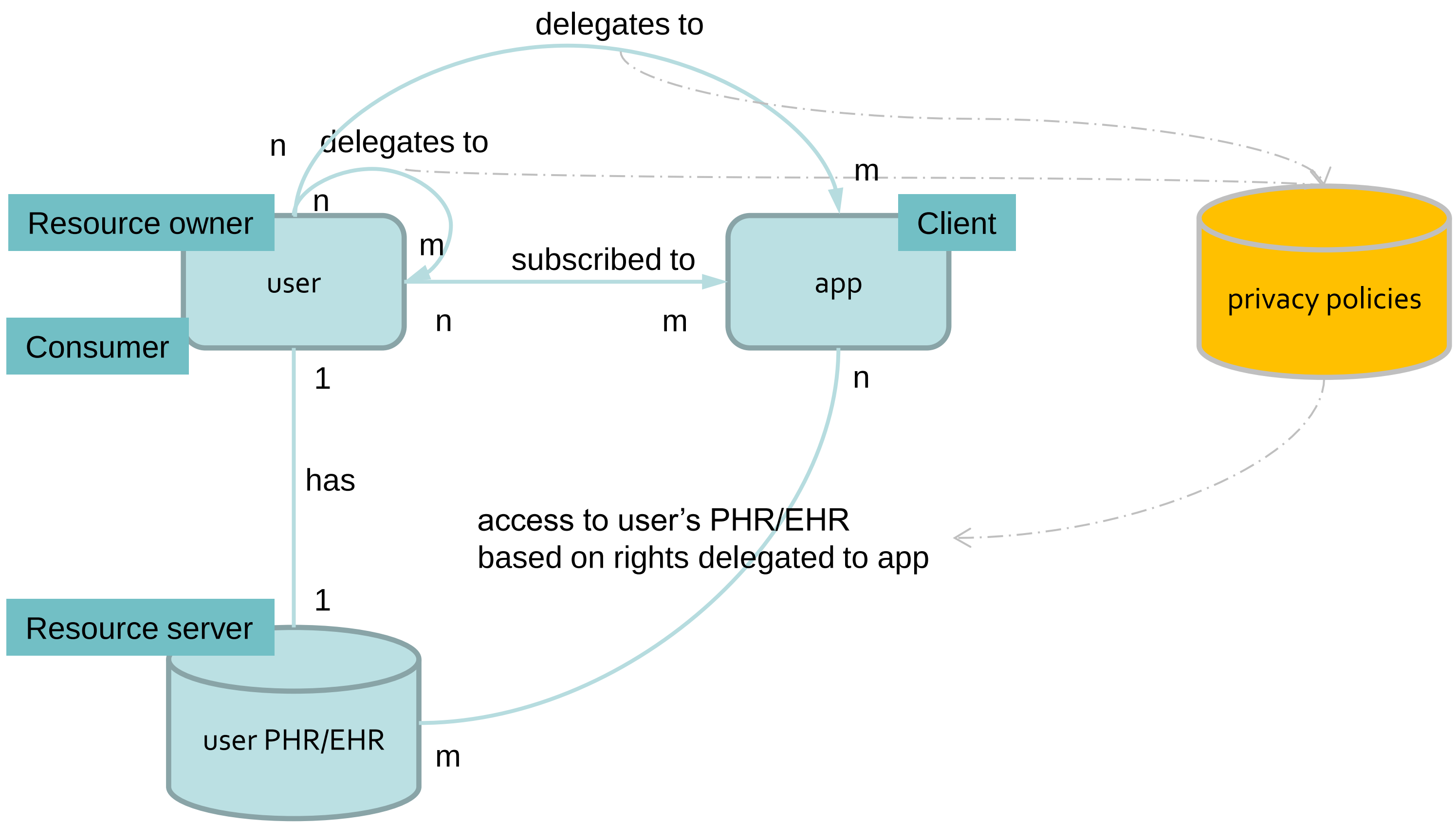
- devices need to be able to add data e.g.
  - activity by ActiWatch
  - weight information by a scale

use cases

1. owner
   - user accesses his or her own data
2. client
   - user delegates right to access (part of) his or her data to an independent device
3. authorized user
   - user delegates right to access (part of) his or her data to another user
4. authenticated user
   - user delegates right to access (part of) his or her data to an application to be used by any identified user

relationship

Used (OAuth2.0) terminology



delegates to

n

delegates to

m

n

Resource owner

m

subscribed to

Client

user

app

privacy policies

n

m

Consumer

1

n

has

access to user's PHR/EHR
based on rights delegated to app

1

Resource server

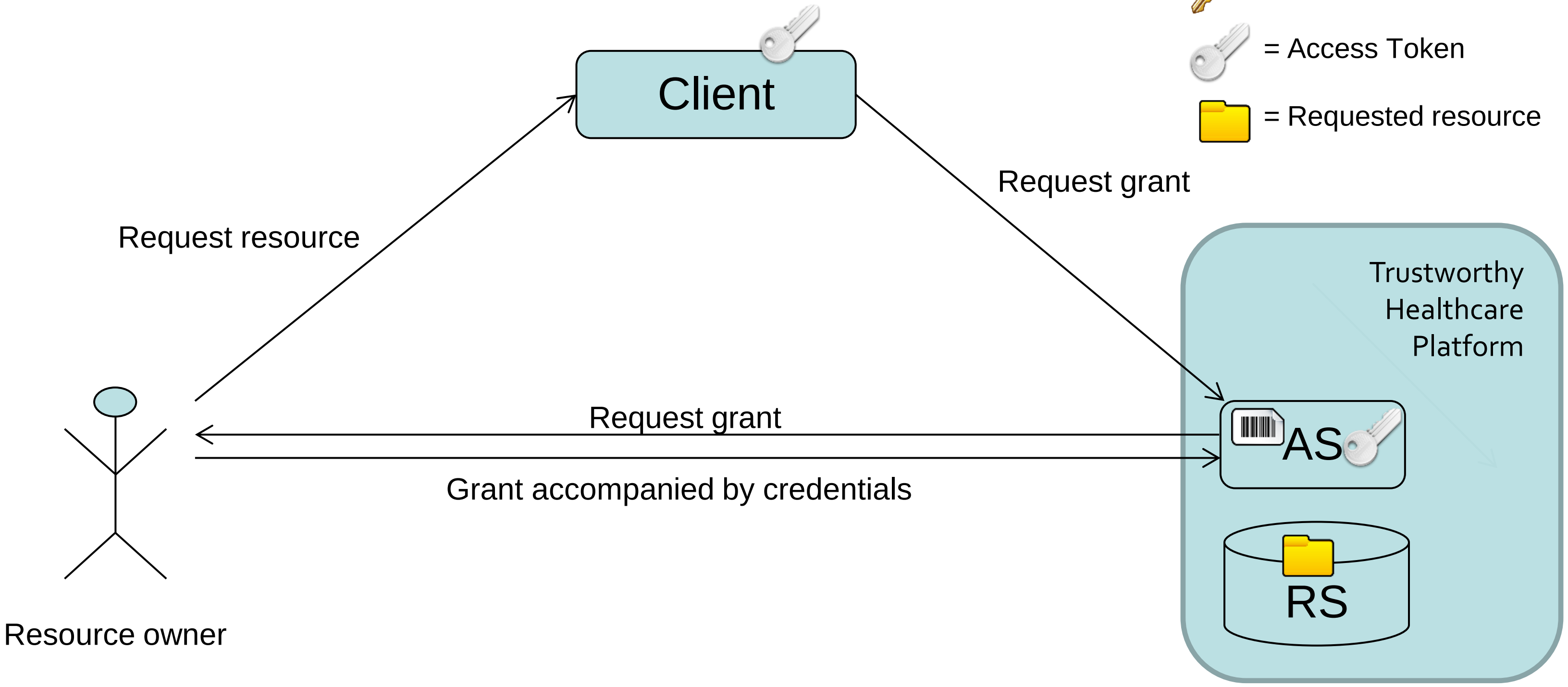
user PHR/EHR

m

use case: owner

- user accesses his or her own data
- uses standard OAuth 2.0

AS = Authorization Server
RS = Resource Server



= Owner Grant

= Refresh Token

= Access Token

= Requested resource

use case: client - setup

- Owner delegates rights to client
- OAuth2.0 framework is sufficient

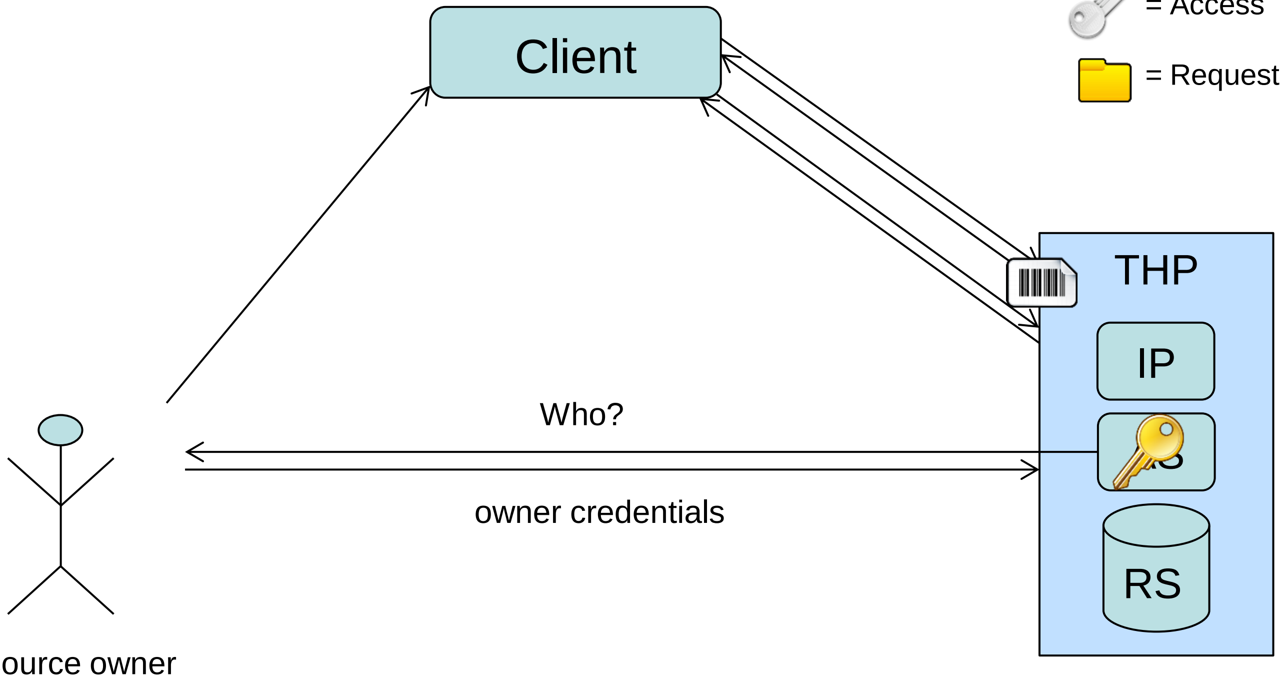
AS = Authorization Server
IP = Identity Provider
RS = Resource Server

▥ = Owner Grant

🔑 = Refresh Token

🗝 = Access Token

📁 = Requested resource



use case: client - resource access

AS = Authorization Server
IP = Identity Provider
RS = Resource Server

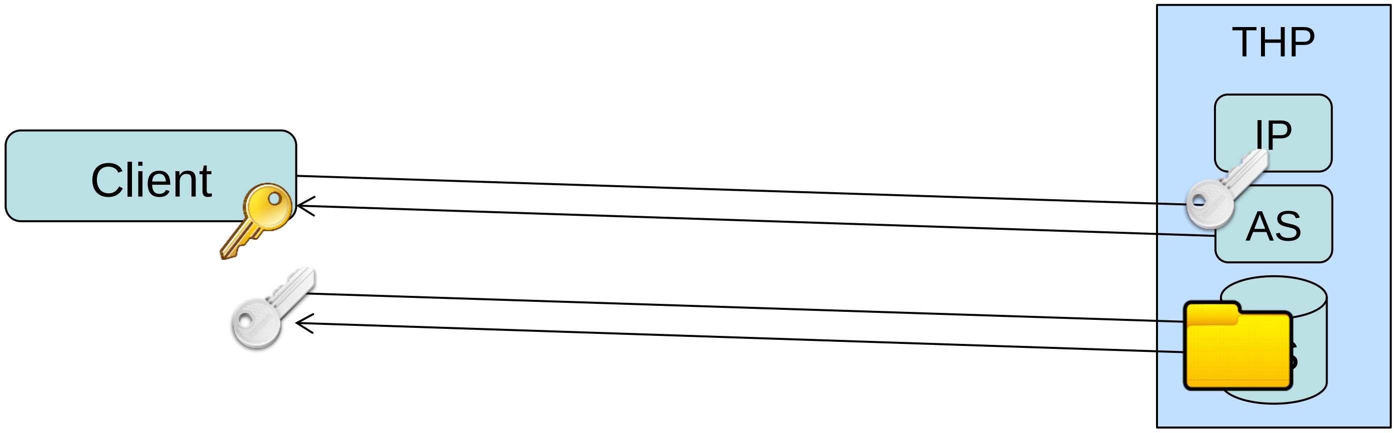
▥ = Owner Grant

🔑 = Refresh Token

🗝 = Access Token

📁 = Requested resource

- Refresh Token to generate Access Token with limited lifetime
- Same access token can be used to perform several accesses.
- Lifetime for refresh token established by owner.
- Difference w.r.t. to OAuth 2.0, client is autonomous

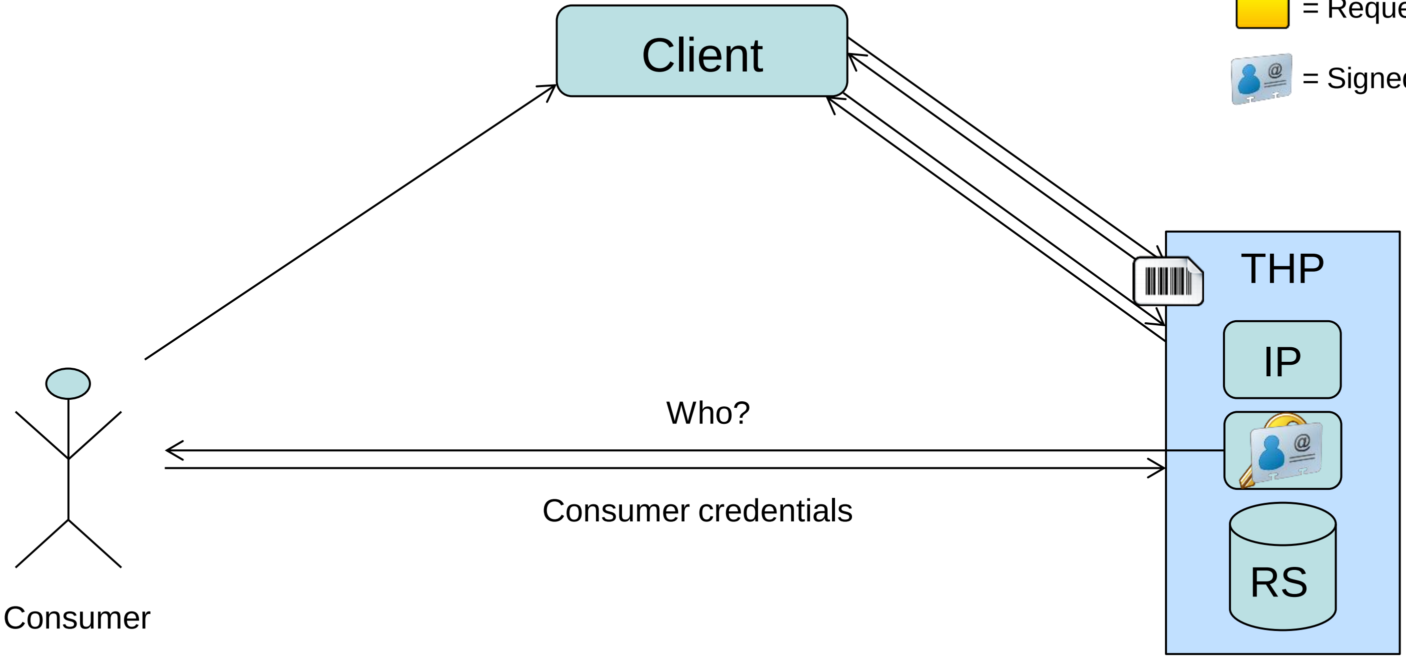


use case: authorized user - setup

AS = Authorization Server
IP = Identity Provider
RS = Resource Server

▥ = Owner Grant
🔑 = Refresh Token
🗝 = Access Token
📁 = Requested resource
🪪 = Signed Identity

- user delegates right to access (part of) his or her data to another user
- consumer are specific THP users, THP keeps track of delegations
- Identity of consumer required by client.
- Rights given by the owner to the consumers, not to the app/client.
- OpenIdConnect needed.



use case: authorized user - resource access

AS = Authorization Server
IP = Identity Provider
RS = Resource Server

- Refresh Token to generate Access Token with limited owner-scope
- Same access token can be used to perform several accesses.
- Lifetime for refresh token short. One refresh token per session.
- Consumer must be online.

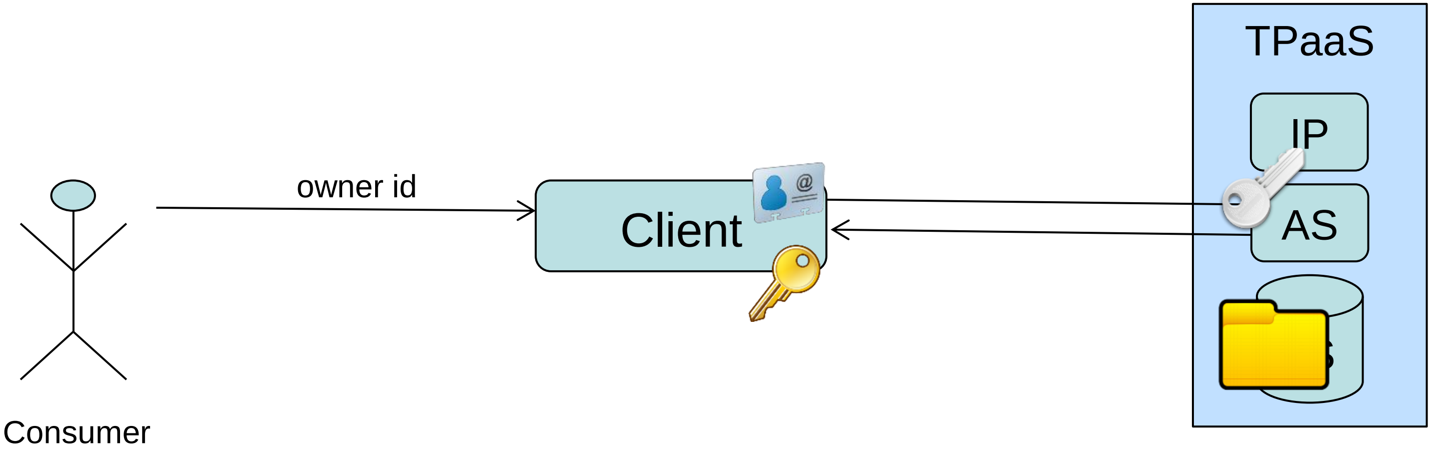
🎫 = Owner Grant

🔑 = Refresh Token

🗝 = Access Token

📁 = Requested resource

🪪 = Signed Identity



use case: authenticated user - setup

- user delegates right to access (part of) his or her data to an application to be used by any identified consumer
- the consumers are any THP users.
- identity of both owner and consumer required by client.
- rights given by the owner to the client, not to the consumers.
- OpenIdConnect needed.

AS = Authorization Server
IP = Identity Provider
RS = Resource Server

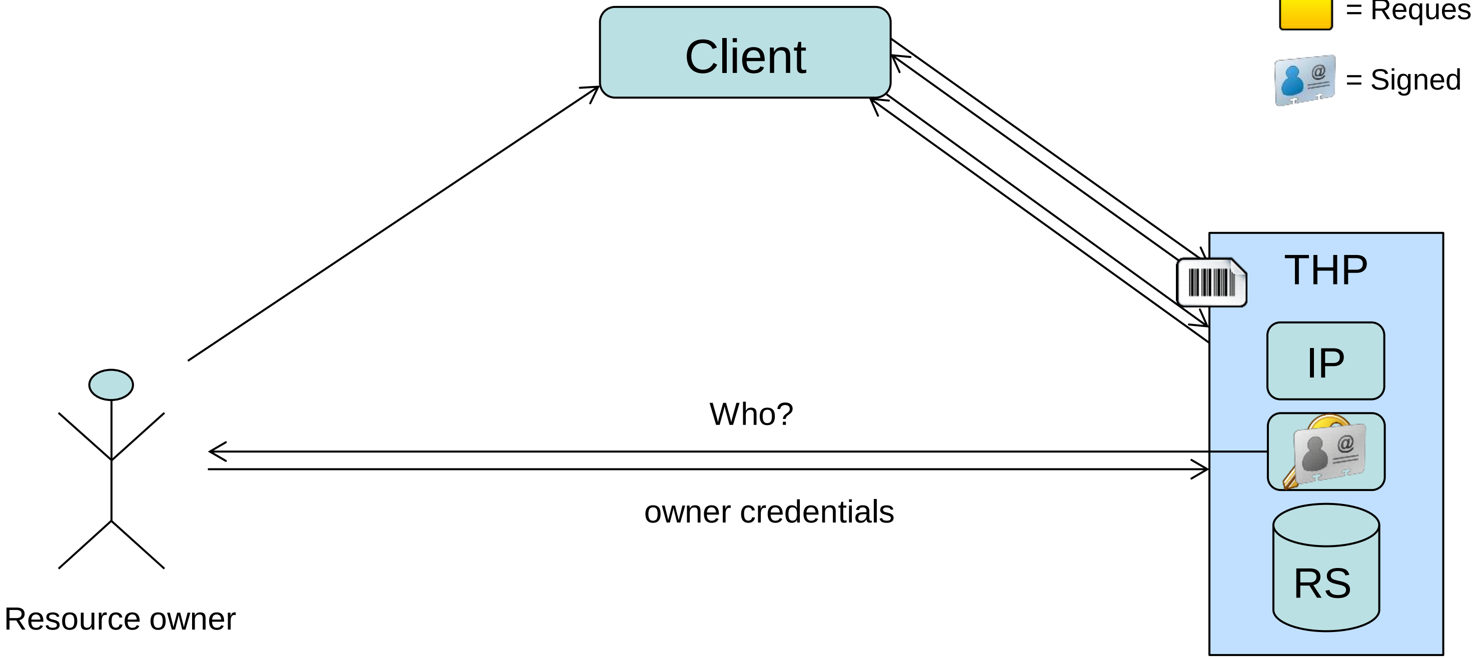
▥ = Owner Grant

🔑 = Refresh Token

🗝 = Access Token

📁 = Requested resource

🪪 = Signed Identity

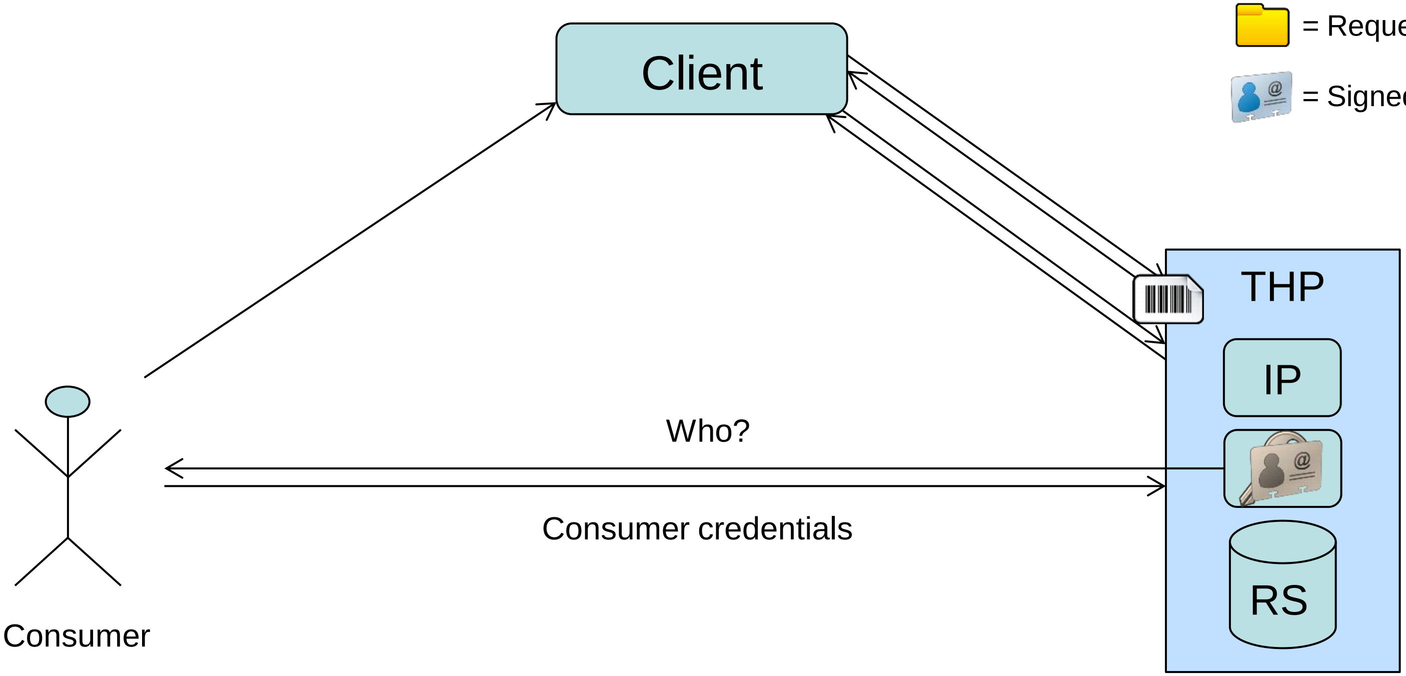


use case: authenticated user - “logon”

- user delegates right to access (part of) his or her data to an application to be used by any identified consumer
- the consumers are any THP users.
- identity of both owner and consumer required by client.
- rights given by the owner to the client, not to the consumers.
- OpenIdConnect needed.

AS = Authorization Server
IP = Identity Provider
RS = Resource Server

= Owner Grant
= Refresh Token
= Access Token
= Requested resource
= Signed Identity



use case: authenticated user - resource access

- Refresh Token to generate Access Token with different consumer
- Lifetime for refresh token established by owner.
- Consumer must be online.

AS = Authorization Server
IP = Identity Provider
RS = Resource Server

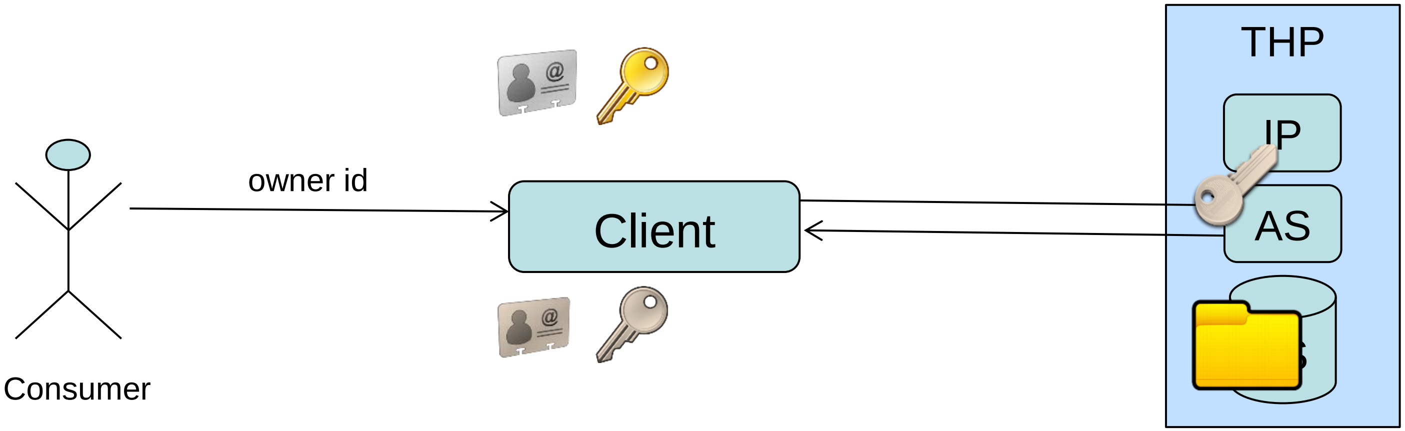
▥ = Owner Grant

🔑 = Refresh Token

🗝 = Access Token

📁 = Requested resource

🪪 = Signed Identity



Summary

- extending OAuth2.0
- using OpenIDConnect
- logging all resource requests for auditing purposes
- REST full interface

demo description

- demo shows use cases 1 and 2
- 4 VMs
  - one running all databases
  - one running logservice
  - one running OAuth2Share
  - one running the applications

TClouds EC CONTRACT No: 257243

"The TClouds project has received funding from the European Union's Seventh Framework Programme ([FP7/2007-2013]) under grant agreement number ICT-257243."

If you need further information, please contact the coordinator:

Technikon Forschungs- und Planungsgesellschaft mbH
Burgplatz 3a, 9500 Villach, AUSTRIA
Tel: +43 4242 233 55   Fax: +43 4242 233 55 77
E-Mail: coordination@tclouds-project.eu

The information in this document is provided "as is", and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.