

Bringing Continual Dynamic Authorization to COTS Applications

Paul Heaney

CTO

ProofID



**Paul
Heaney**

CTO

ProofID

Authentication - Solved



Historical Authorization

Static set of
rules/permissions

Defined once and
enforced

Problem

Not aware of context

- Where are you coming from?
- What you're attempting to do?
- What you've done recently?
- Signals from the wider ecosystem e.g.:
 - lots of transfers to a particular account
 - high volume of people looking at a person's medical records

Problem

Not aware of context

Legislation requiring risk based Authorization

- e.g. PSD2

Problem

Not aware of context

Legislation requiring risk based Authorization

Doesn't support a Dynamic Authorization model

- Such as in an identity first/identity first models

Scenarios

Banking / Finance

Health Care

Loyalty Schemes

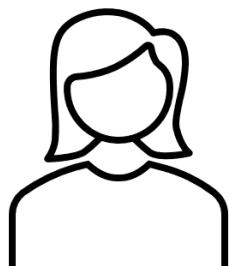
Publishing

and many more

Inhouse vs Commercial off the Shelf (COTS)

- For custom applications this is easier
 - The code can be updated
- For COTS apps this is more difficult
 - We can't make changes to these applications

Accessing a COTS application

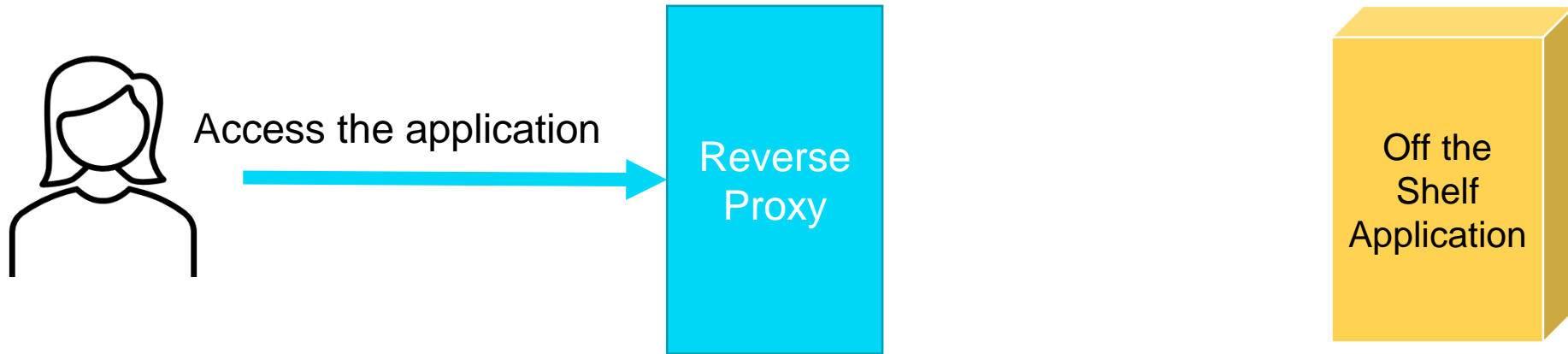


Access the application

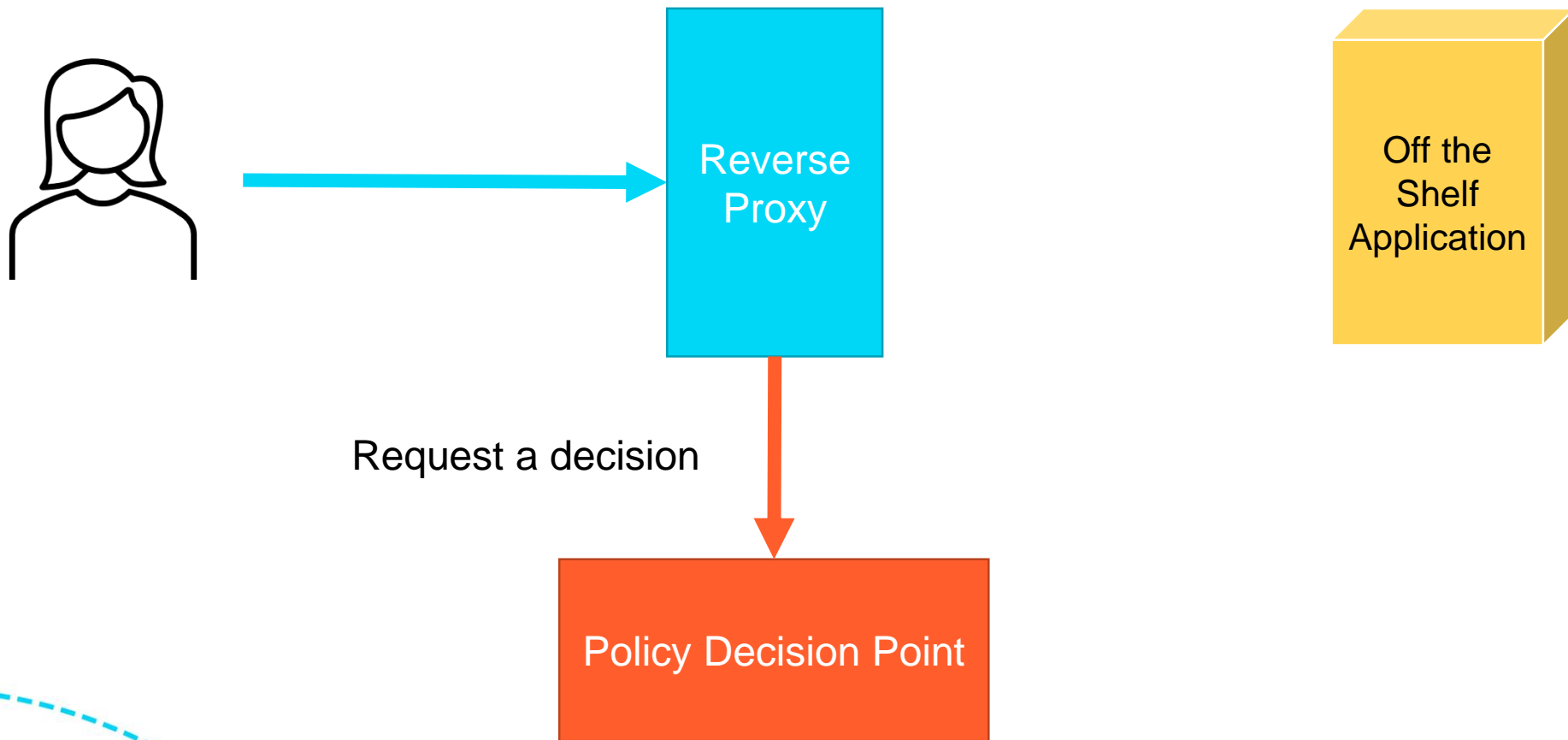


Bringing Dynamic Authorization

Add one reverse proxy



Add a Policy Decision Point (PDP)



Decision Request

- What goes into the decision request:

Decision Request

- What goes into the decision request:

Who

sub: a123456
email: fred@example.com
department: Finance
location: Manchester

Decision Request

- What goes into the decision request:

Who

sub: a123456
email: fred@example.com
department: Finance
location: Manchester

What

Request URL: https://example.com/admin/dolt
HTTP Method: POST
Request Payload: action=save&name=ACME%20Corp
Request Headers: User-Agent=Chrome, Cookies=...

Decision Request

- What goes into the decision request:

Who

sub: a123456
email: fred@example.com
department: Finance
location: Manchester

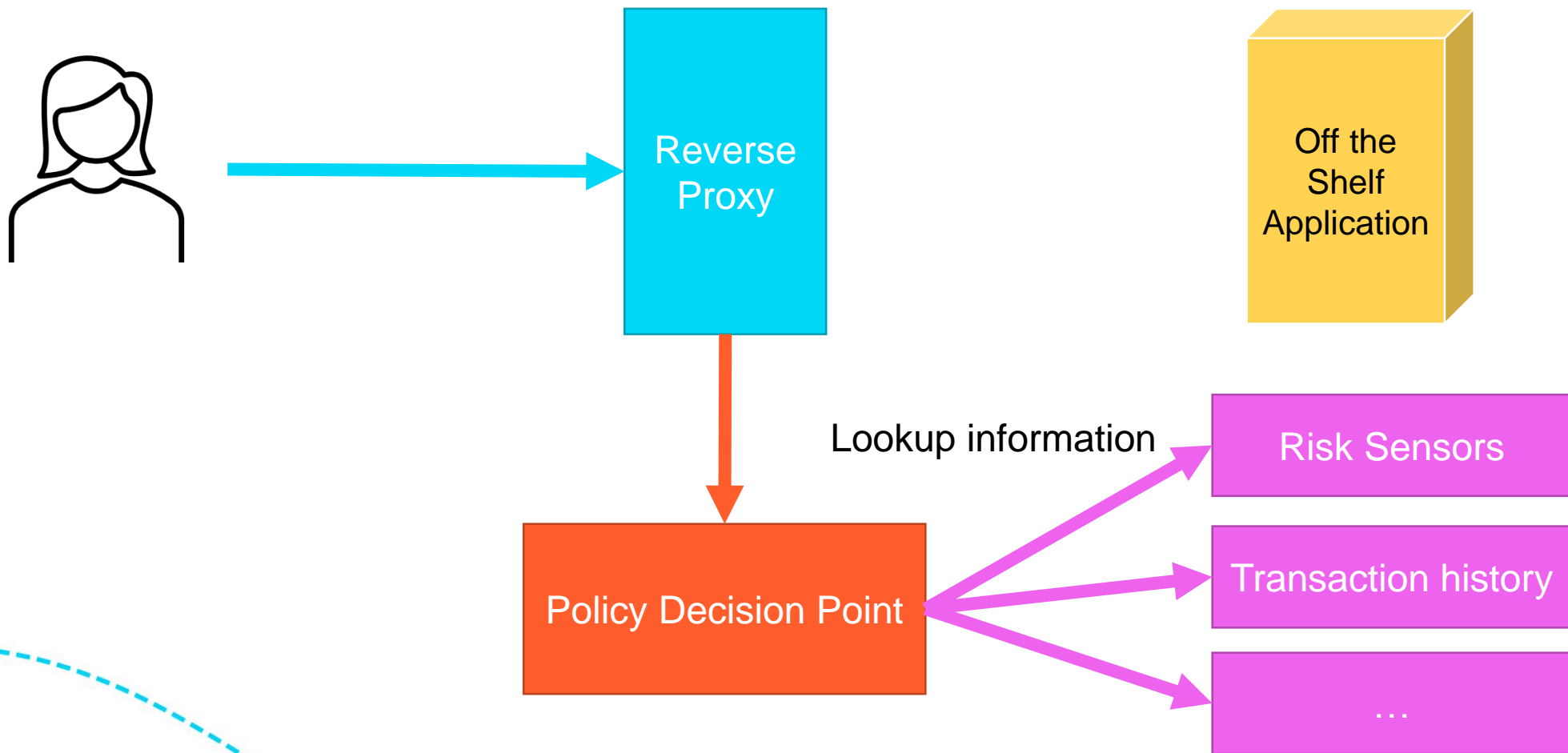
What

Request URL: https://example.com/admin/dolt
HTTP Method: POST
Request Payload: action=save&name=ACME%20Corp
Request Headers: User-Agent=Chrome, Cookies=...

Where
(Context)

ip: 13.22.198.65
Managed Device: false
Session ID: 9FC1438D-131B-4604-928D-5E12B4ADE8C5
Request location: Las Vegas

Add a sprinkling of Information Points



Policy Information Points that may be consulted

Risk Systems	Customer transaction history	Consent Stores
Consortium/external	Customer transaction history	Explicit and implicit
User Behaviour Analytics (UBA)	Historical behaviour	Family relationships
Internal risk systems		Vehicle / Dealer / owner / mechanic / valet

Decision

- PDP applies policies and decides what to do:

Allow

Require transaction Authorization

Filter / Modify Request

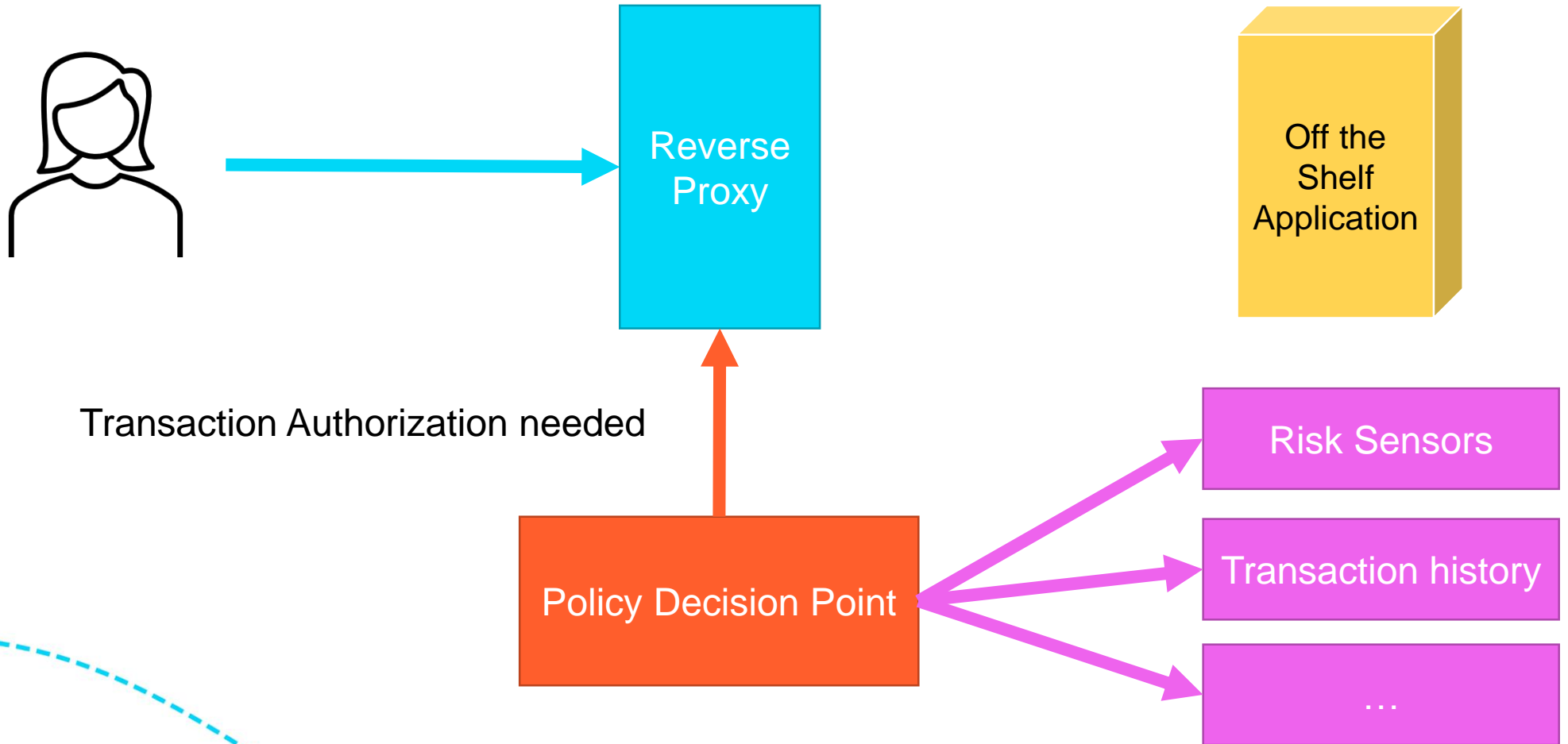
Log details/update truth data

Deny

Terminate Session

- Enforced by Reverse Proxy (Policy Enforcement Point)

Transaction Authorization



Triggering Transaction Authorization

Pause and redirect

- When your authentication solution needs to occur within the browser

Client Initiated Backchannel Authentication
(CIBA)

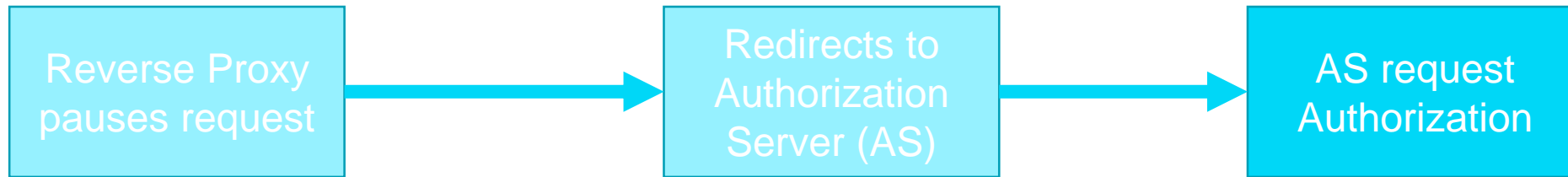
- When you're able to perform push notifications to a customer

Pause and redirect



- Store current request in state parameter (JWE)
- Store the details of the transaction in the OpenID Connect Request Object
- Request a scope of `txnAuth:{hash of transaction}`

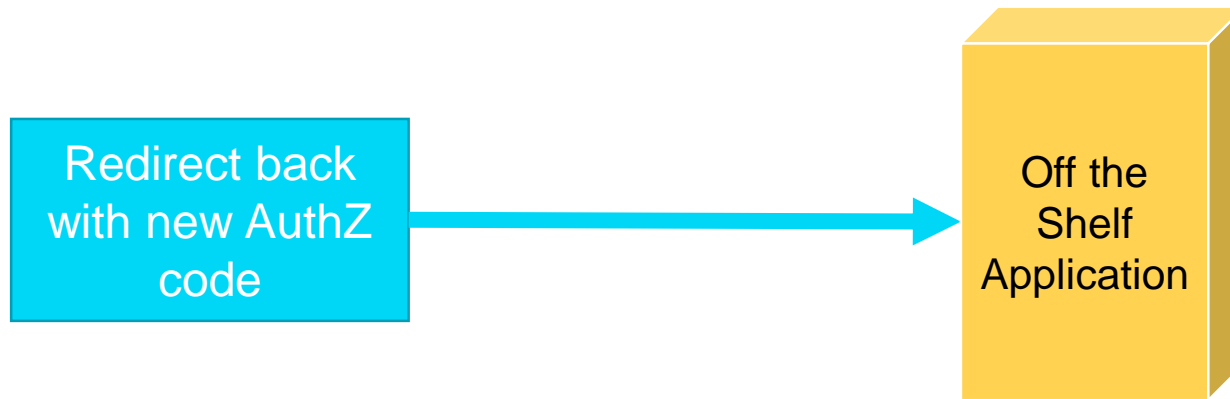
Pause and redirect



Display transaction details (from request object)

Have user Authenticate and Authorize transaction

Pause and redirect



Standard OpenID Connect dance

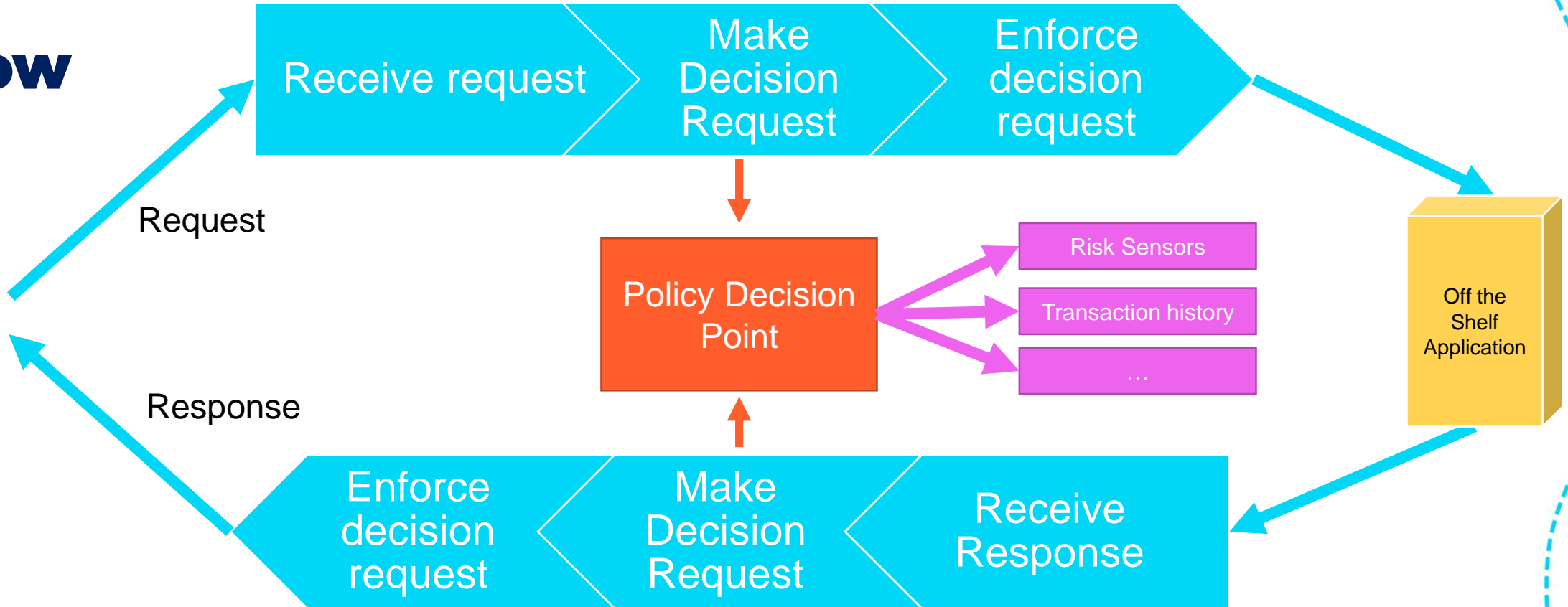
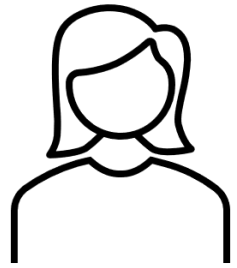
Un-pause transaction by reassembling from state

Create transaction hash of transaction

Verify scopes returned include the transaction hash

Resume request

Flow



Take aways

- Dynamic Authorization requirements are increasing
- Deploy a reverse proxy
- Implement a Policy Decision Point
- Plug in appropriate Information Points
- Utilise open standards

Q & A