RSA Conference2015
San Francisco | April 20-24 | Moscone Center

SESSION ID: IDY-F03

UMA in Health Care: Providing Patient Control or Creating Chaos?

David Staggs JD, CISSP

Technologist / IP Attorney

Staggs PLLC

UMA Value Proposition

- User Managed Access (UMA) brings granular control to the health care ecosystem
- scalable, secure, and provides uninterruptible consent
- patient control encourages trust and participation
- extends electronic workflow:
  - reduces paper
  - simplifies audit and compliance
  - multi-use workflows possible

Making Possible Real

- unlock access to electronic health records (EHR) and personal health records (PHR)
- develop an ecosystem that opens entrepreneurial opportunities and accelerates progress
- establish publicly available APIs to the software ecosystem and share vast stores of data
- the solution must respect individuals' privacy and guard against data breaches

The Future is RESTful

- RESTful Health Exchange (RHEx)
- links to specific EHR data – not just moving entire record
- allows app providers to address small practices
- adds capabilities that are missing in secure email
- uses OAuth 2 and OpenID Connect (OIDC) profiles

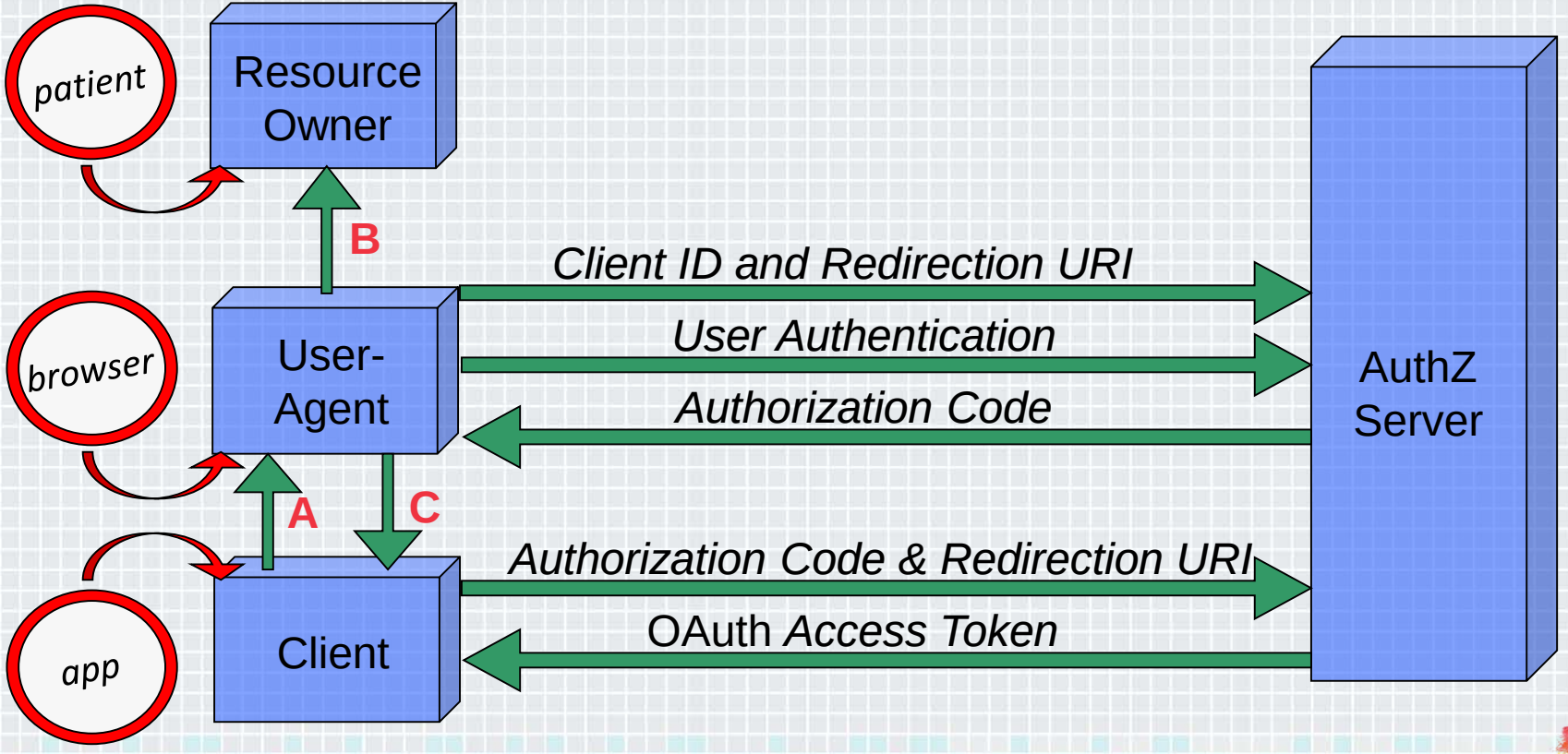
The Future is SMART

- Substitutable Medical Applications Reusable Technologies
- opens up the EMR system silo
- open-source, developer-friendly API
- gives application ecosystem access to data
  - encourages innovation
- uses OAuth 2 and OpenID Connect (OIDC) profiles

The Future is on FHIR

- Fast Healthcare Interoperability Resources
- data formats and elements with an API for exchanging EHR
- uses an HTTP-based RESTful protocol
- uses OAuth 2 for authentication to APIs
- adopted by RHEx and SMART
- supported by Health Level Seven (HL7)

OAuth 2 Authorization (Real Time) Code Grant



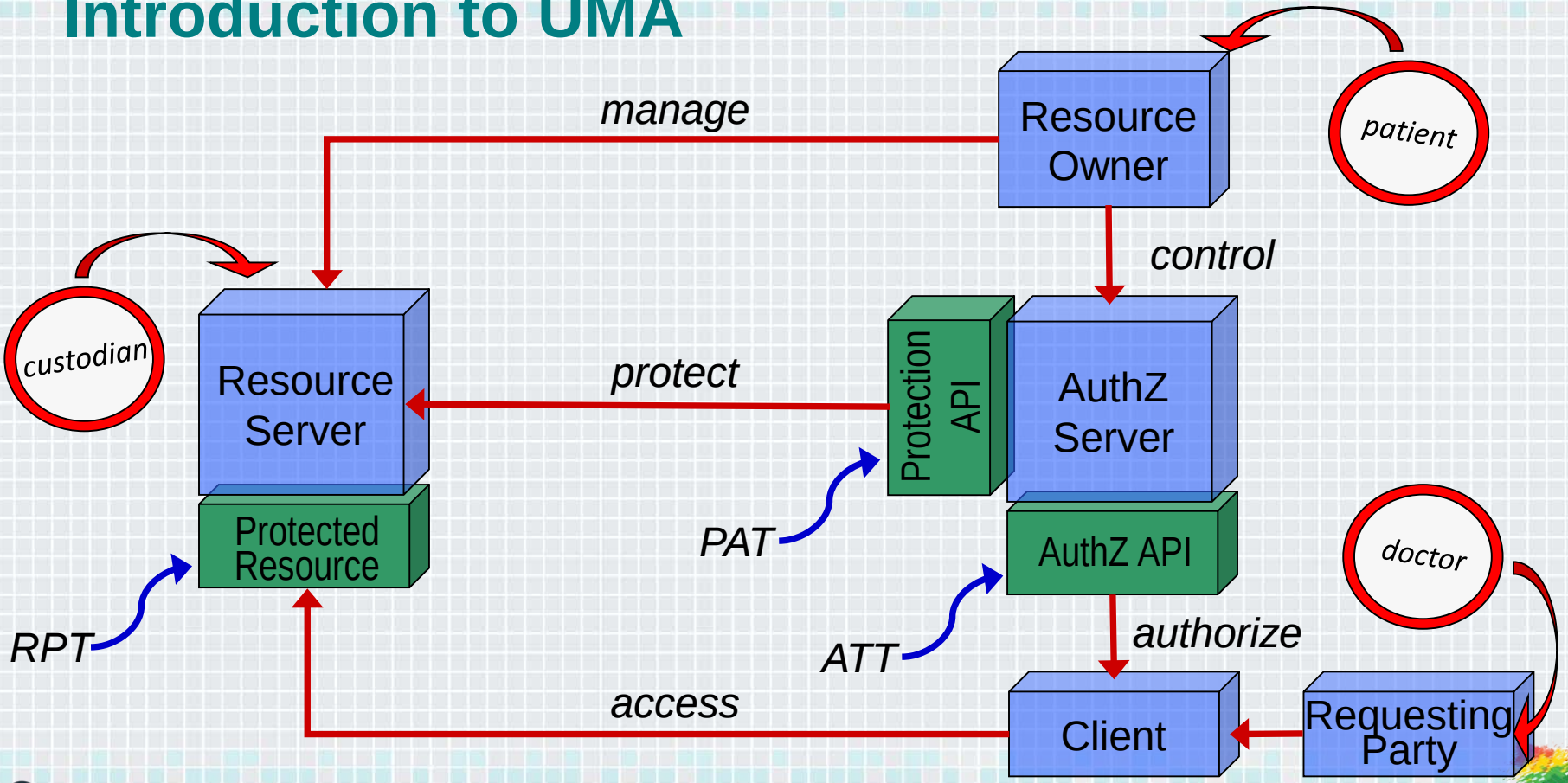
OAuth 2 Framework

- replaces the anti-password pattern
- resource owner OKs token for client's access
- HTTP-based RESTful protocol
- includes scopes / TTL that manage access rights
- permits service chaining (token that can be passed)
- Privacy by Design (PbD)

Security and Privacy

- Protected Health Information (PHI) and HIPAA
- patients should have control over their PHI
- need an extension to OAuth 2 / OIDC profiles
  - use OAuth to protect APIs and OIDC to get credentials
  - enforce patient's consent directives, even when the patient is not available (uninterrupted consent)
- User Managed Access (UMA) provides a solution

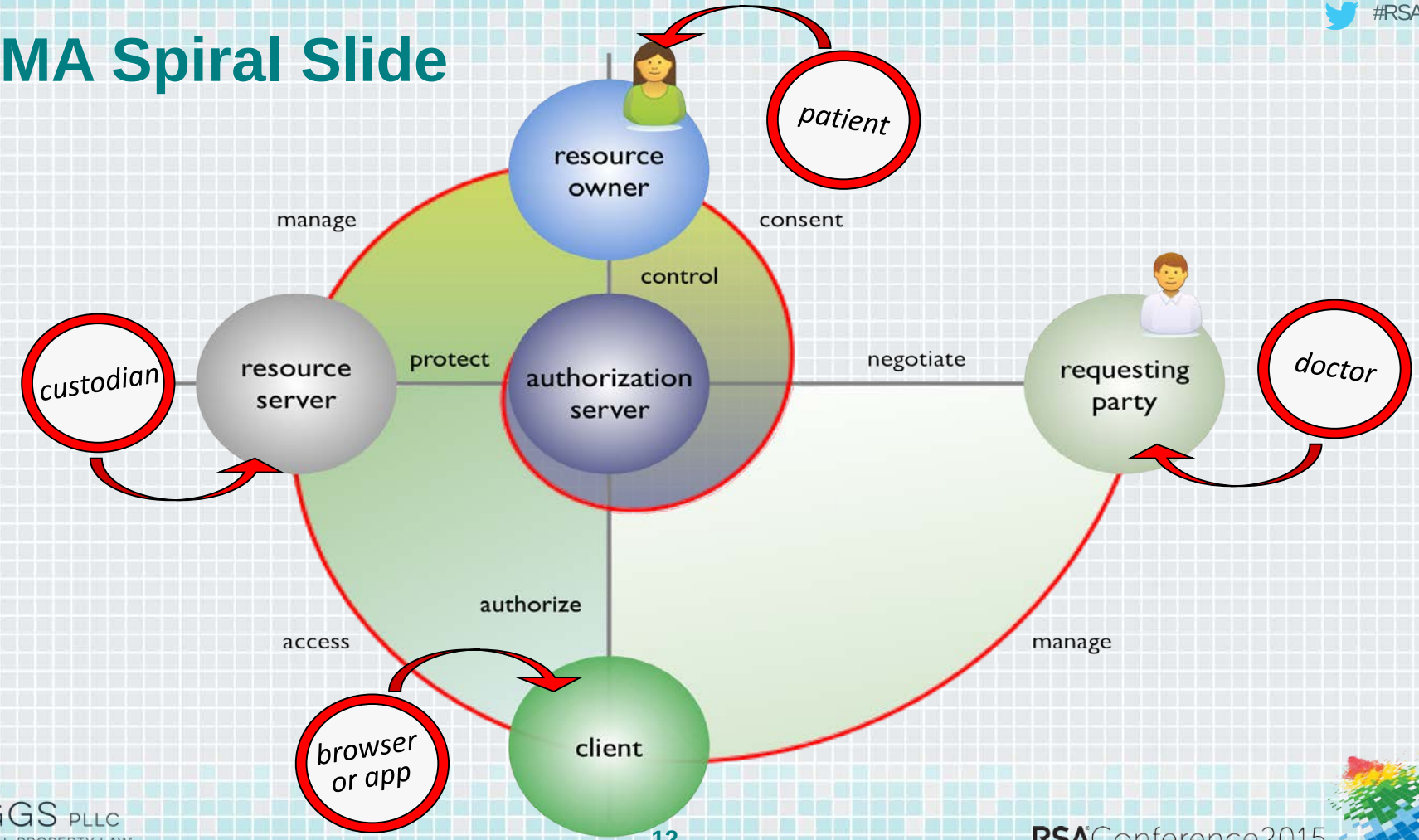
Introduction to UMA



UMA OAuth Tokens

- Authorization API (AAT)
  - authorization server, requesting party, and client
- Request API (RPT)
  - requesting party, client, resource server, and authorization server (not resource owner)
- Protection API (PAT)
  - resource server, authorization server, and resource owner
- *resource owner (e.g. patient) sets access policy and scope*

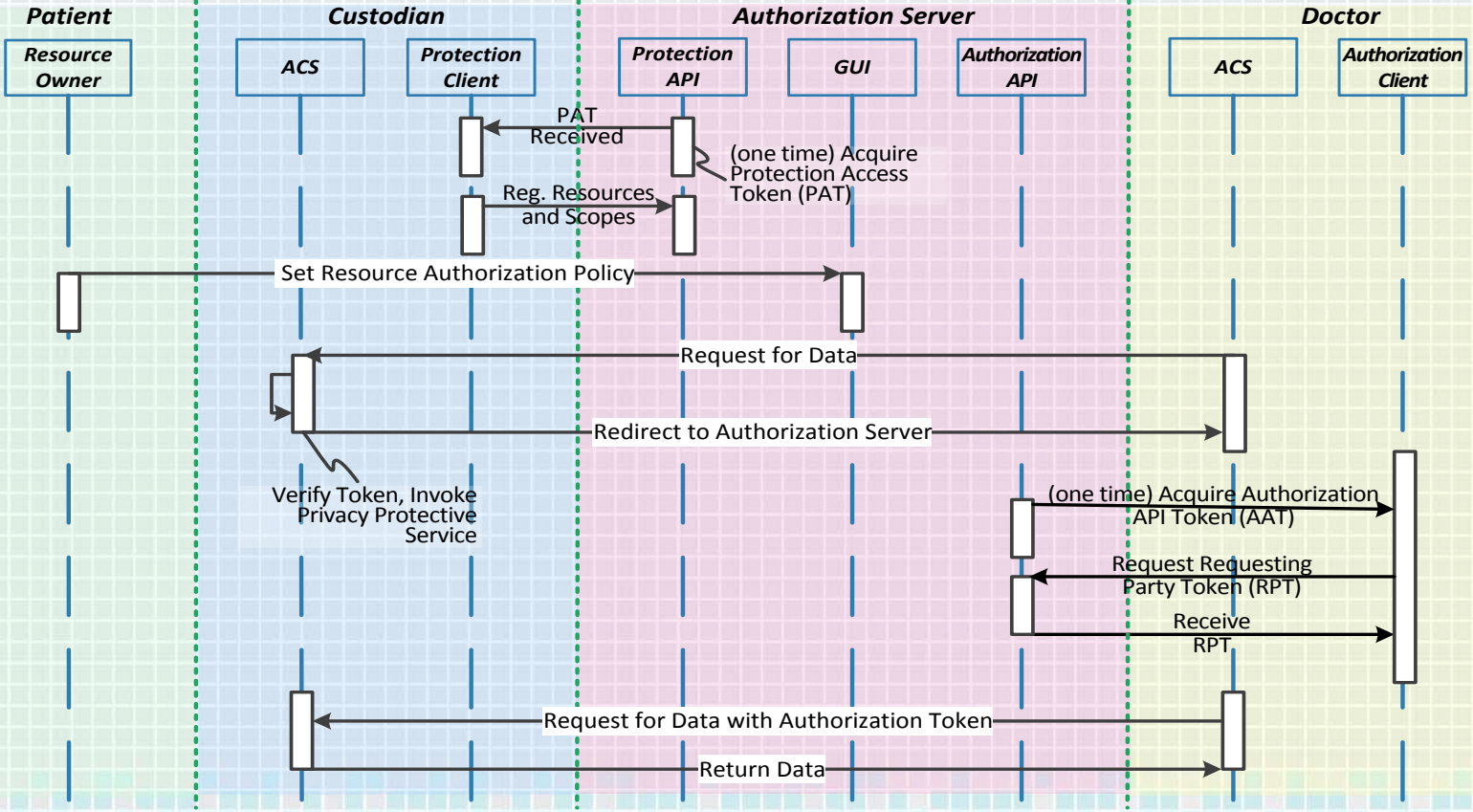
UMA Spiral Slide



UMA's Chaotic Potential

- if patients pick their resource servers (personal cloud) how do they keep track of where everything is?
- will health care providers allow you to use any authorization server to control access to records they create?
- will treatment by multiple providers cause conflicts on which authorizations server is used to control?
  - provider/custodian A requires using only authorization server X
  - provider/custodian B requires using only authorization server Y

UMA Health Ecosystem Deep Dive



Scopes

- scopes provide finer grained control
- scopes have the following:
  - name of the resource that can be displayed to owner
  - human-readable string describing some extent of access
- for example, scope involving reading or viewing resources:

```
{
"name" : "View",
"icon_uri" : "http://www.example.com/icons/reading-glasses"
}
```

Scope Description Documents

- scope description documents have the following:
  - name, type, scopes, icon_uri

```
{
"name" : "Photo Album",
"icon_uri" : "http://www.example.com/icons/flower.png",
"scopes" : [
"http://photoz.example.com/dev/scopes/view",
"http://photoz.example.com/dev/scopes/all"
],
"type" : "http://www.example.com/rsets/photoalbum"
}
```

More Potential Chaos

- token introspection by resource server at authorization server
  - need to understand semantics of the token
- OpenID OAuth profile
  - ID Token – a signed and optionally encrypted JWT containing identity and attribute claims about the user
  - UserInfo Endpoint – a protected resource where the relying party can request additional claims about the user
  - OAuth scopes are used to request individual user attributes

Can We Just Get Along

- health records in a personal cloud spread across resource servers should have uniform scope syntax
- authorization servers' scope description documents
  - simplify resource set registration mechanism
  - prevent scope names from revealing PHI
  - is a pointer to standard scope descriptions politically possible?
- consider HEART (Health Relationship Trust)

OpenID HEART

- health-related profiles layering: OAuth 2.0, OpenID Connect, FHIR, OAuth 2.0 scopes, and UMA
- HEART WG is defining use cases and requirements
- expect an implementation guide soon
- demonstration of current capabilities
  - Eve Maler, ForgeRock, HEART WG Co-Chair

Apply Slide

- identify your use cases requiring uninterrupted consent
- use HEART open source code for a test bed
- mitigate token vulnerabilities
  - audience parameter, state parameter, signed JWTs, redirection URIs
- identify what resources need protection and define terminology
- identify your role in the ecosystem
  - patient UX, authorization server, EHR custodian, OpenID claims provider, organization offering standard scope descriptions