IDENTITY MANAGEMENT FOR ARCHITECTS

Prabath Siriwardena
Senior Director - Security Architecture
prabath@wso2.com | prabath@apache.org

THE EVOLUTION OF IDENTITY

THE EVOLUTION OF IDENTITY

- Centralized

  Most of the systems started being centralized, where the identity attributes were maintained centrally in silos.

- Federated

  Microsoft came up with an initiative called Microsoft Passport to share the user data stored under Microsoft with other relying party web sites.

  Google started to share user data via OpenID/SAML 2.0 and today via OpenID Connect. Facebook started with Facebook Connect and today supports OAuth 2.0 for user data sharing.

THE EVOLUTION OF IDENTITY

- User Centric

  Kim Cameron, one of the distinguished identity architects at Microsoft, with the support from the community came up with the seven laws of identity, in 2005.

  This laid the foundation for user-centric identity paradigm, where the user is in the middle of an identity transaction, between the identity provider and the relying party.

- Self Sovereign

  Talks about giving total control of user data to the user himself/herself.

  You own your identity, not anyone else. There won't be one central authority to manage millions of user records.

THE SEVEN LAWS OF IDENTITY

THE SEVEN LAWS IDENTITY

- Understand the dynamics causing digital identity systems to succeed or fail in various contexts, expressed as the Laws of Identity.
- How we can prevent the loss of trust and go forward to give Internet users a deep sense of safety, privacy, and certainty about whom they are relating to in cyberspace.
- Community effort initiated by Kim Cameron from Microsoft.

USER CONTROL AND CONSENT

- Identity systems must only reveal information identifying a user with the user's consent.
- OpenID user consent
- OAuth 2.0 scopes
- UMA access control policies

MINIMAL DISCLOSURE FOR A CONSTRAINED USE

- The solution that discloses the least amount of identifying information and best limits its use is the most stable long-term solution.
- Bartender only needs to know whether his customer’s age is greater than 21, not the age.
- Uber drivers need to call its passengers, only within a given, limited time period, but they do not want to know the passenger's’ phone numbers.

JUSTIFIABLE PARTIES

- Digital identity systems must be designed so the disclosure of identifying information is limited to parties having a necessary and justifiable place in a given identity relationship.
- Relying party information should be revealed to the user.
- Microsoft Passport

PLURALISM OF OPERATORS AND TECHNOLOGIES

- A universal identity system must channel and enable the interworking of multiple identity technologies run by multiple identity providers.
- No single identity system is going to suffice in all contexts, and no single identity provider is going to be justifiable in all contexts.

HUMAN INTEGRATION

- The universal identity metasystem must define the human user to be a component of the distributed system integrated through unambiguous human-machine communication mechanisms offering protection against identity attacks.
- The last couple of feet between the computer console and the human is where most bad things happen.
- Phishing and other social engineering attacks exploit the user.
- A stable identity system mitigates these threats.

CONSISTENT EXPERIENCE ACROSS CONTEXTS

- The unifying identity metasystem must guarantee its users a simple, consistent experience while enabling separation of contexts through multiple operators and technologies.
- A stable identity system presents an easy-to-understand abstraction to the end user that is consistent no matter what underlying technology or identity provider is involved.

DIRECTED IDENTITY

- A universal identity system must support both "Omni-directional" identifiers for use by public entities and "unidirectional" identifiers for use by private entities, thus facilitating discovery while preventing unnecessary release of correlation handles.
- Many public entities need to behave like beacons, broadcasting their identities to the world. However, expect them to use a private identifier to track user's personal activity, so stable identity systems must support both omnidirectional identity (beacons) and unidirectional identity (my private relationship).
- SAML NameID Policy (Persistent Pseudonym Identifiers / Transient Pseudonym Identifiers)

BUSINESS BENEFITS (ROI)

IDENTITY AND ACCESS MANAGEMENT

- Identity and Access Management (IAM) is the security discipline that enables the right individuals (or things) to access the right resources at the right times for the right reasons. (Gartner)
- At its core, an IAM facility must have:
  An administrative capability with trained staff and a management console of some type.
  A storage facility that can accommodate and protect the identity data.
  A policy definition capability that provides guidance on how protected data to be treated.
  Recognition of regulation that affects the management of identity data and a mechanism to demonstrate adherence.

BUSINESS BENEFITS

- Business oversight

  Provide greater visibility and sustainability of users and their access rights within and across targeted systems.

- Business relationships

  Employees, customer, partners, suppliers

- Business agility

  Respond rapidly to changes in the internal and external environment without losing momentum or vision. Adaptability, flexibility and balance are three qualities essential to long-term business agility.

  IAM systems must become agile enough to support new business initiatives and move quickly — almost in real time — to deal with threats as they arise

BUSINESS BENEFITS

- Service delivery

  The manner in which a corporation provides application access to users throughout its lifecycle. Service delivery covers the design, development, deployment, operation and retirement of the applications or other services.

- User productivity

  Single Sign On. One set of credentials.

- Cost reduction

  Less help desk cost. Self services

BUSINESS BENEFITS

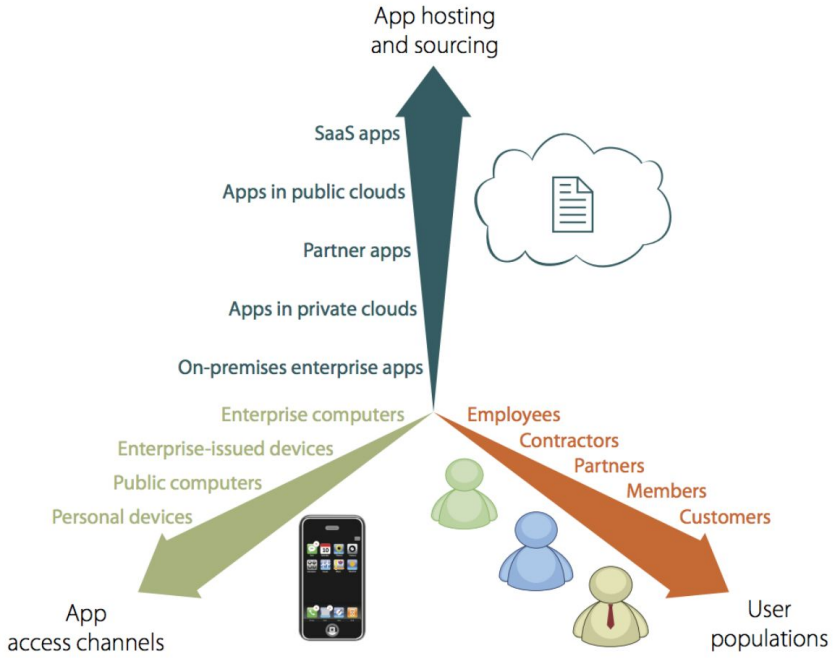
- Security

  Multi-factor authentication, fraud detection/preventions

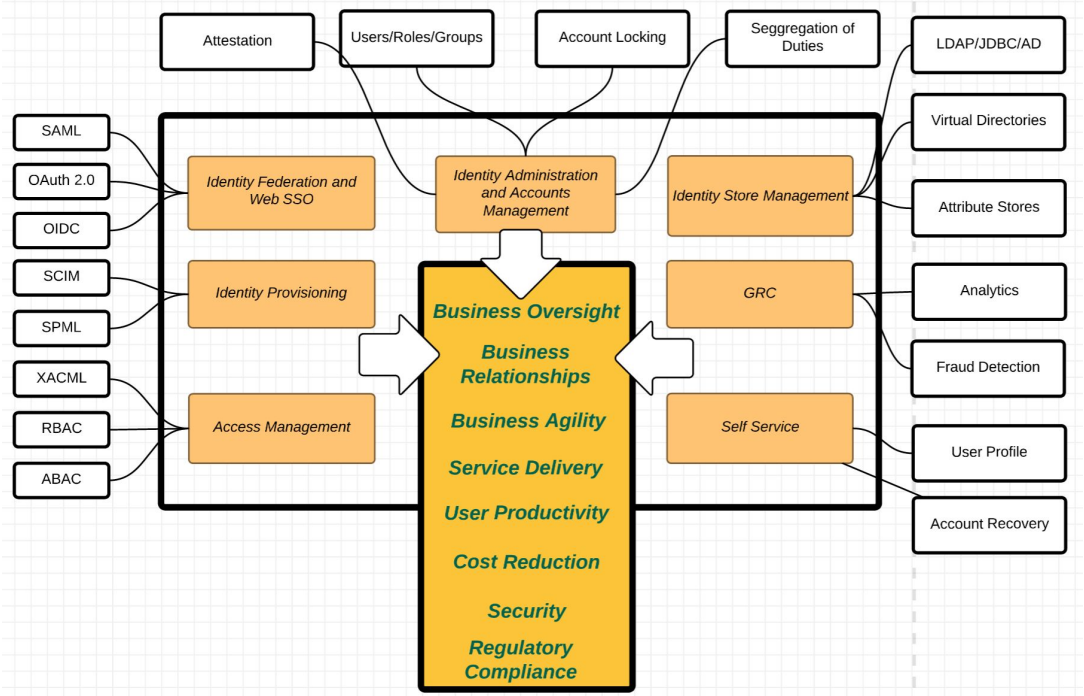
- Regulatory compliance

  HIPAA, GDPR, PCI

BUSINESS BENEFITS



BUSINESS BENEFITS



FORRESTER IDENTITY MANAGEMENT MATURITY MODEL

Nonexistence (level-0)

- No identity management system in place — and do not realize the need.
- The Human Resource (HR) department possibly maintains a spreadsheet to manage all the employee information and their salaries.
- These kind of environments do not require users to login — having just logged into the wireless network assumes users can access anything.

Ad hoc (level-1)

- Occasionally, not consistent, not planned, disorganized.
- Not all the applications require users to login and access.
- On case by case basis users are manually provisioned into applications — and the user records are duplicated across multiple applications.
- The single user may be in different applications with different usernames. This may be due to some constraints on username by each application.

Repeatable (level-2)

- Intuitive, not documented, occurs only when necessary.
- Many organizations are at this level.
- An employee on his/her first day has to meet an IT guy and get his/her email and other applications setup.
- IT admin knows exactly what needs to be done. Whenever an employee resigns, the IT admin has to manually deprovision the user from all the applications.
- Still the user records are duplicated across multiple applications — and users may have different credentials for different applications.

Defined (level-3)

- Documented, predictable, occurs only when necessary.
- This is a better version of level-2.
- The entire identity management process is documented — possibly maintained in a check list.
- When an employee joins the company, the HR department sends an email to the IT department and the IT department creates all the access required for the employee by his/her role.
- Still the user may be provisioned to multiple applications manually.

Measured (level-4)

- Well-managed, formal, often automated, evaluated frequently.
- The level-4 maturity level removes lot of manual involvement from the level-3.
- Once the user record is created in the HR application, the user will be automatically provisioned to all the applications with the appropriate level of access rights.
- The user will be deprovisioned automatically when he/she resigns.

Optimized (level-5)

- Continuous and effective, integrated, proactive, usually automated.
- This is the ultimate wisdom expanded on the level-4 maturity level.
- Identity governance play a key role here.
- There will be multiple dashboards, based on the organizational roles to monitor what's going on.
- For example, how many external users signed up by month — and out of all signed up users how many are actively using the system..

IDENTITY FEDERATION & SINGLE SIGN ON

OVERVIEW

- Single Sign On - login once - access a set of services without further login.
- Federated identity management enables identity information to be developed and shared among several entities and across trust domains.
- Single Sign On can be within a single trust domain and between multiple trust domains.

STANDARD BASED IDENTITY FEDERATION

- SAML 2.0 Web SSO
- OpenID Connect
- WS-Federation
- OpenID
- CAS

DEFINITIONS

- Identity Provider

  The authority behind user identities

  Makes assertions about users (authentication, authorization, attribute)

- Relying Party / Service Provider / Client

  Relying on an assertion provided by the identity provider. Provides services to end users

  Can be a mobile app / web app Trusts one or more identity providers

SAML 2.0 OVERVIEW

- An XML standard for exchanging authentication and authorization data between entities which is a product of the OASIS Security Services Technical Committee.
- SAML 1.0 was adopted as an OASIS standard in Nov 2002
- SAML 1.1 was ratified as an OASIS standard in Sept 2003
- SAML 2.0 became an OASIS standard in Mar 2005
- Liberty Alliance donated its Identity Federation Framework (ID-FF) specification to OASIS, which became the basis of the SAML 2.0 specification. Thus SAML 2.0 represents the convergence of SAML 1.1, Liberty ID-FF 1.2, and Shibboleth 1.3.

SAML 2.0 BASE STANDARDS

- Extensible Markup Language (XML)
- XML Schema
- XML Signature
- XML Encryption (SAML 2.0 only)
- Hypertext Transfer Protocol (HTTP)
- SOAP

SAML 2.0 COMPONENTS

- Assertions

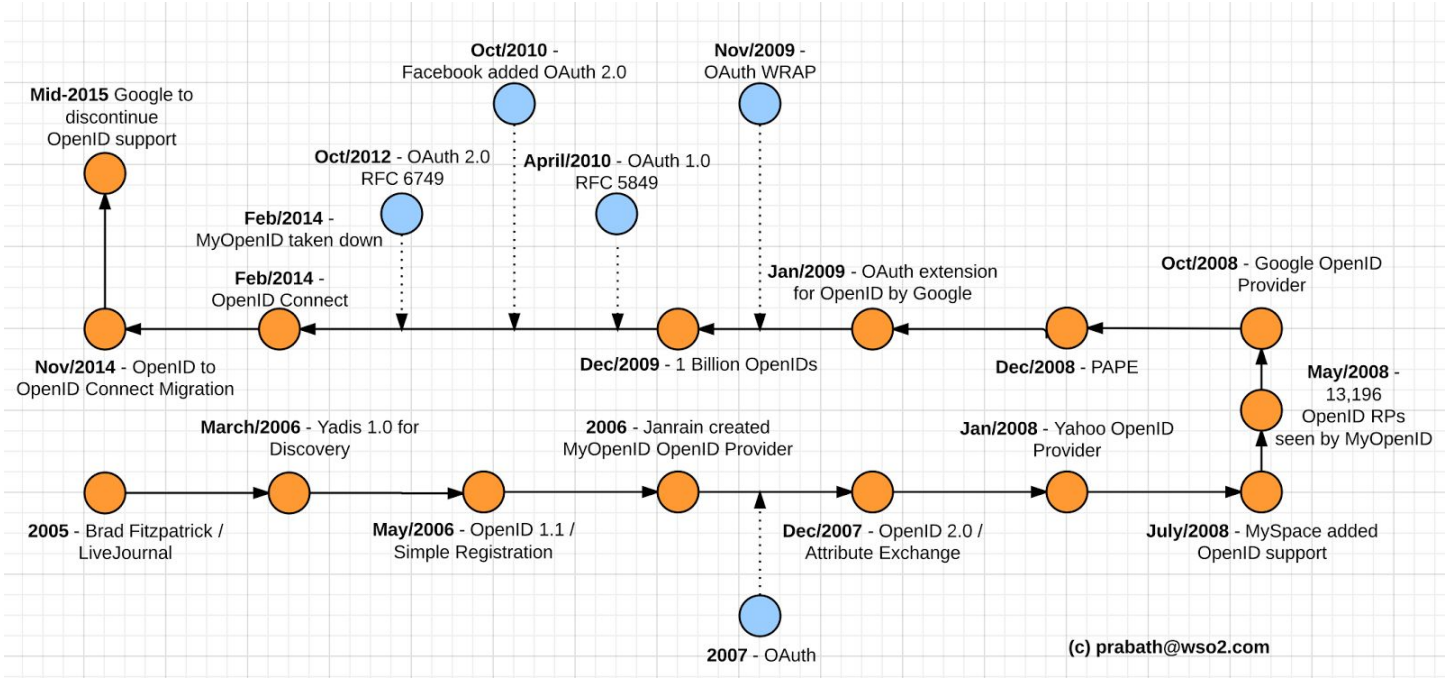
  Authentication, Attribute and Authorization information
- Protocol

  Request and Response elements for packaging assertions
- Bindings

  How SAML Protocols map onto standard messaging or communication protocols
- Profiles

  How SAML protocols, bindings and assertions combine to support a defined use case

OPENID CONNECT



SAML 2.0 vs. OPENID CONNECT

- SAML is XML based while OIDC is JSON based
- SAML has multiple bindings while OIDC has one binding
- Both are enterprise ready In last couple of years - there were more OIDC implementations than SAML
- OIDC is more mobile and SPA friendly
- Build a new app today? Use OIDC!

FEDERATION ASSURANCE LEVELS

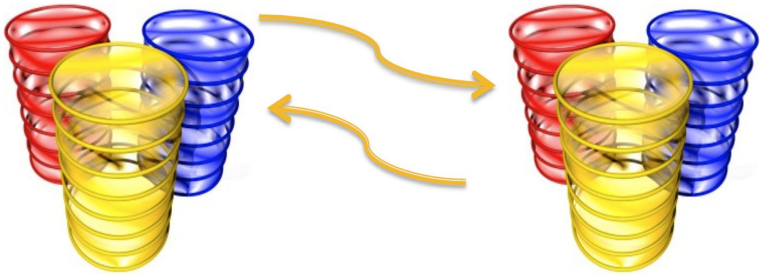
- Defined by NIST SP 800-63C
- The FAL describes requirements for how assertions are constructed and secured for a given transaction.

FAL	Requirement
1	Bearer assertion, signed by IdP.
2	Bearer assertion, signed by IdP and encrypted to RP.
3	Holder of key assertion, signed by IdP and encrypted to RP.

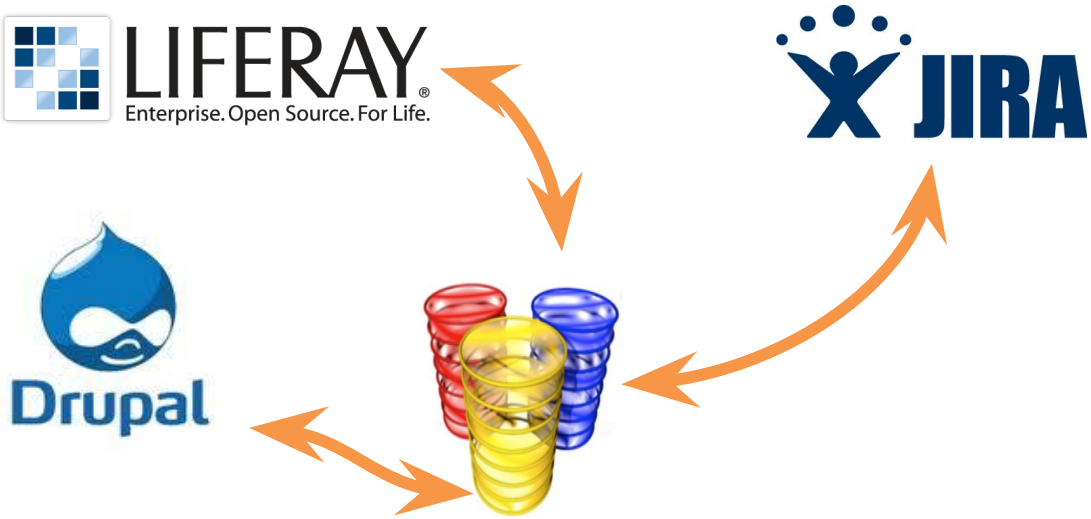
IDENTITY PROVISIONING

SYNCHRONIZATION

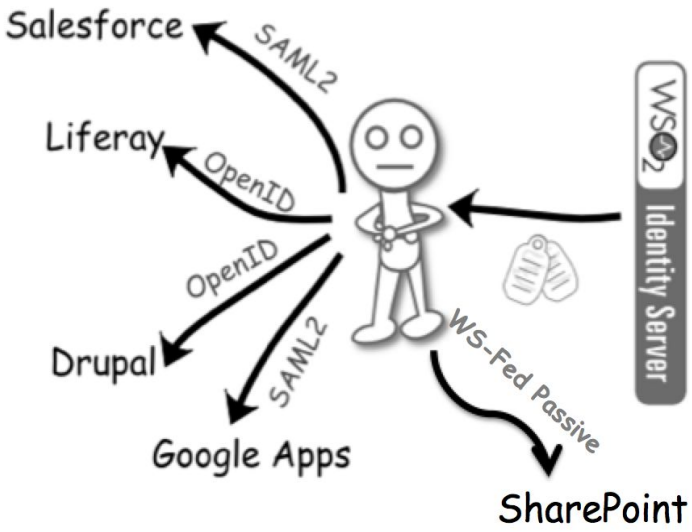
SYNCHRONIZATION



SHARING

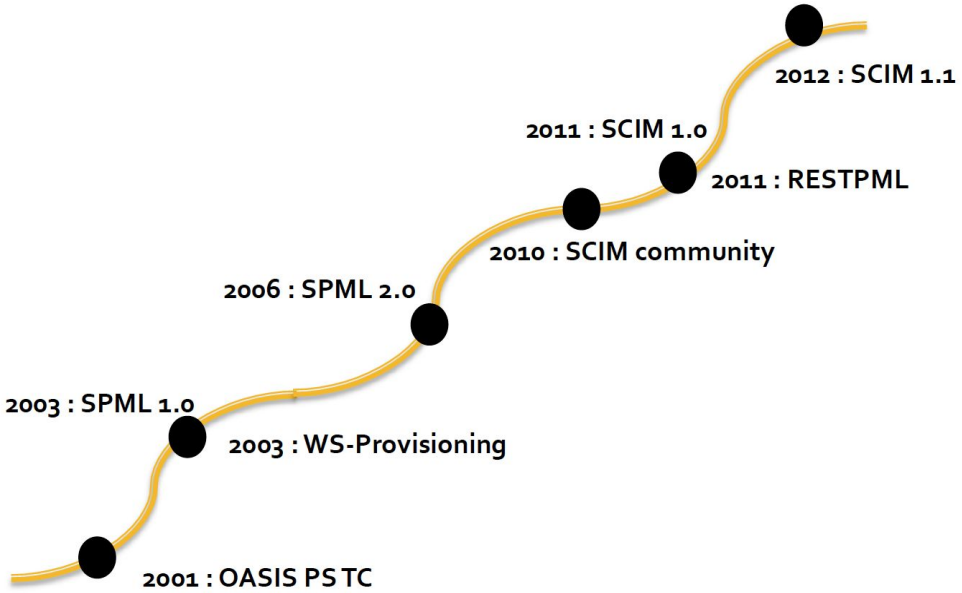


SINGLE SIGN ON



PROVISIONING

STANDARD BASED PROVISIONING



SPML 1.0

```
<addRequest>
    <attributes>
        <attr name="objectclass">
          <value>emailUser</value>
        </attr>
        <attr name="cn">
            <value>Jane Doe</value>
        </attr>
        <attr name="gn">
            <value>Jane</value>
        </attr>
        <attr name="sn">
            <value>Doe</value>
        </attr>
    </attributes>
</addRequest>
```

```
<addResponse result = "urn:oasis:names:tc:SPML:1:0#success">
    <identifier type = "urn:oasis:names:tc:SPML:1:0#EMailAddress">
        <spml:id>Jane.Doe@acme.com</id>
    </identifier>
    <attributes>
        <attr name="mailBoxLimit">
            <value>50MB</value>
        </attr>
    </attributes>
</addResponse>
```

SPML 1.0

```
<modifyRequest>
    <identifier type = "urn:oasis:names:tc:SPML:1:0#EMailAddress">
        <id>Jane.Doe@acme.com</id>
    </identifier>
    <spml:modifications>
        <modification name="mailBoxLimit">
            <value>100MB</value>
        </modification>
    </modifications>
</modifyRequest>
```

```
<modifyResponse result = "urn:oasis:names:tc:SPML:1:0#success" />
```

SPML 2.0 (DSML)

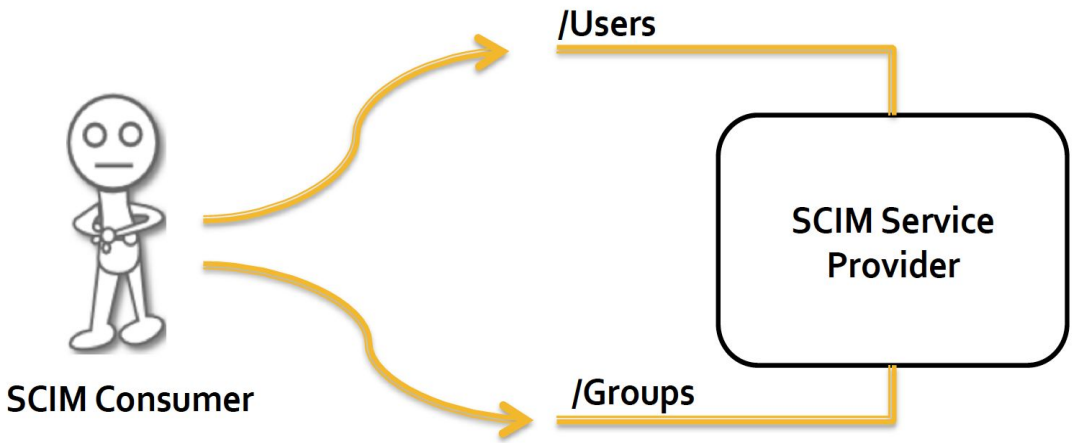
```
<spml:addRequest xmlns:spml="urn:oasis:names:tc:SPML:2:0">
   <spml:containerID  ID="OU=accounting,DC=acme.com" targetID="acme.com "/>
   <spml:data>
      <attr name="CN" xmlns="urn:oasis:names:tc:DSML:2:0:core">
         <value> John Doe </value>
      </attr>
      <attr name="uid" xmlns="urn:oasis:names:tc:DSML:2:0:core">
         <value>jdoe</value>
      </attr>
      <attr name="objectclass" xmlns="urn:oasis:names:tc:DSML:2:0:core">
         <value>user</value>
      </attr>
   </spml:data>
</spml:addRequest>
```

SPML 2.0 (XSD)

```
<spml:addRequest targetID="target2" xmlns:spml="urn:oasis:names:tc:SPML:2:0">
   <spml:data>
      <user>
         <cn>John Doe</cn>
         <uid>jdoe</uid>
         <email>jdoe@acme.com</email>
         <phone>
            <home>555-2323</home>
            <work>555-6767x321</work>
         </phone>
      </user>
   </spml:data>
</spml:addRequest>
```

STANDARD BASED PROVISIONING

SCIM 1.1



SCIM 1.1

```
{"schemas":[],
 "name": {"familyName":"siriwardena",
          "givenName":"prabath"},
 "userName":"prabath",
 "password":"prabath123",
 "externalId":"prabathext",
 "emails":[ {"primary":true,
          "value":"prabath@wso2.com",
          "type":"home"},
         {"value":"prabathsiriwardena@yahoo.com",
          "type":"work"}]
}
```

curl -k --user admin:admin -d @add-user.json --header "Content-Type:application/json" https://localhost:9443/wso2/scim/Users

SCIM 1.1

```
{
  "schemas":["urn:scim:schemas:core:1.0"],
 "displayName" : "OSDC",
 "externalId" : "OSDC",
 "members": [
   {
     "value": "f64e6507-756d-4a14-ac43-c9d02167f411",
     "display": "prabath"
   }
 ]
}
```

curl -k --user admin:admin -d @add-group.json --header "Content-Type:application/json" https://localhost:9445/wso2/scim/Groups

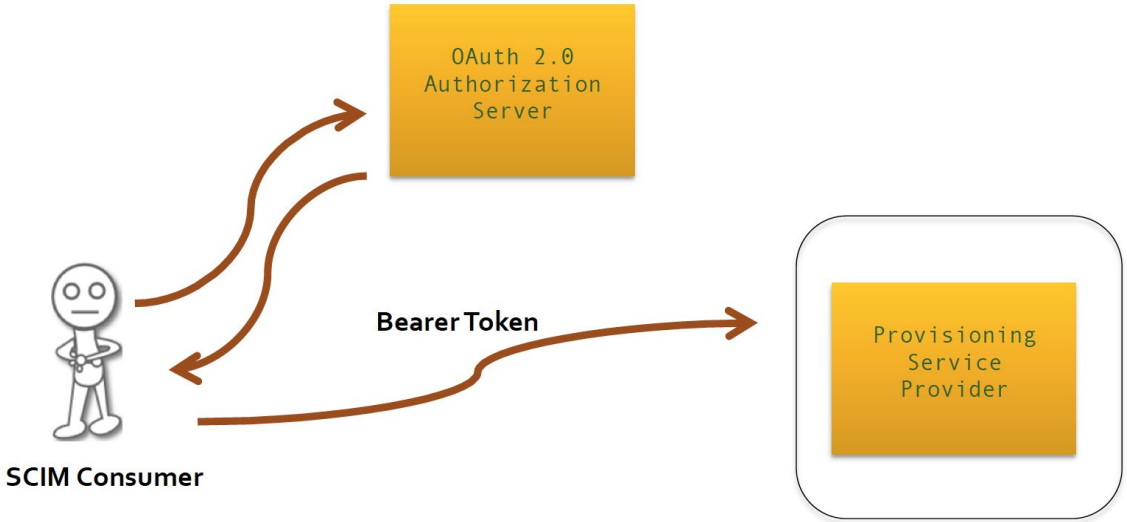
SCIM 1.1

Resource	Endpoint	Operations	Description
User	/Users	**GET**, **POST**, **PUT**, **PATCH**, **DELETE**	Retrieve/Add/Modify Users
Group	/Groups	**GET**, **POST**, **PUT**, **PATCH**, **DELETE**	Retrieve/Add/Modify Groups
Service Provider Configuration	/ServiceProviderConfigs	**GET**	Retrieve the Service Provider's Configuration
Schema	/Schemas	**GET**	Retrieve a Resource's Schema
Bulk	/Bulk	**POST**	Bulk modify Resources

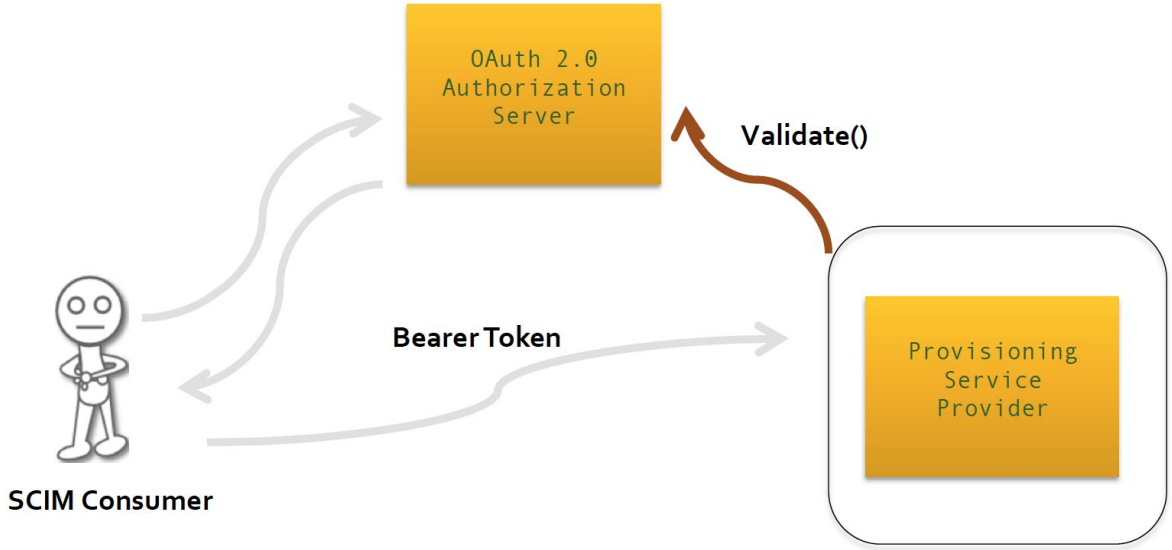
AUTHENTICATING SCIM REQUESTS

- HTTP Basic Authentication
- OAuth 2.0

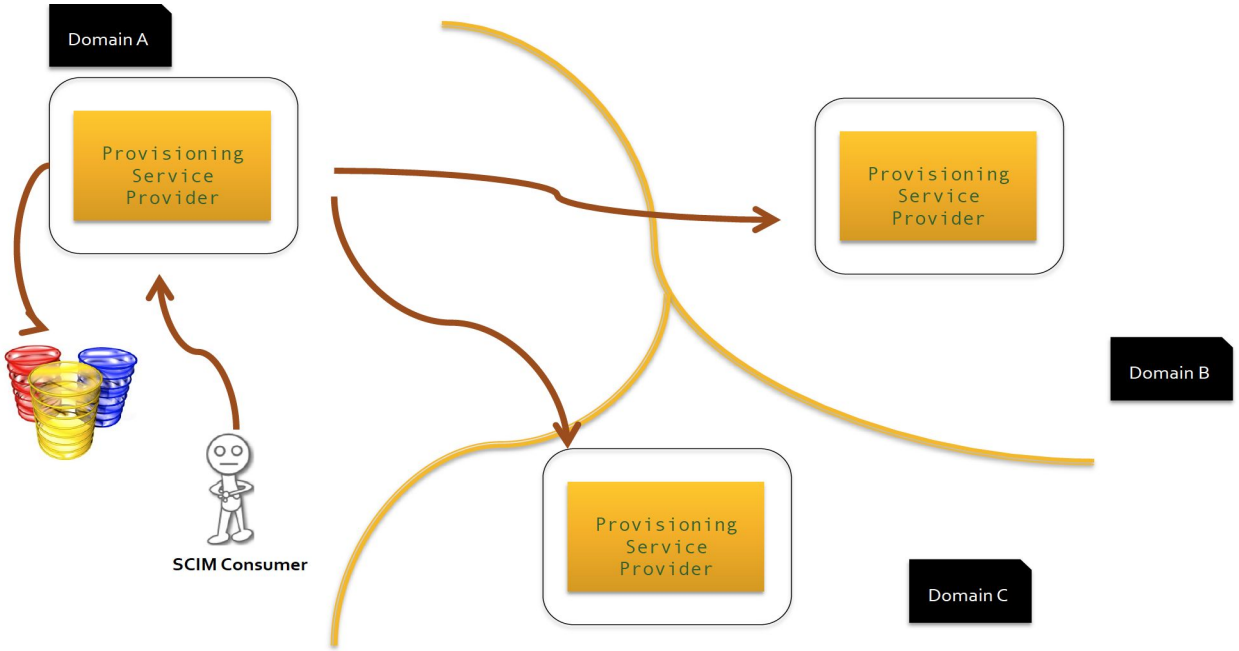
AUTHENTICATING SCIM REQUESTS



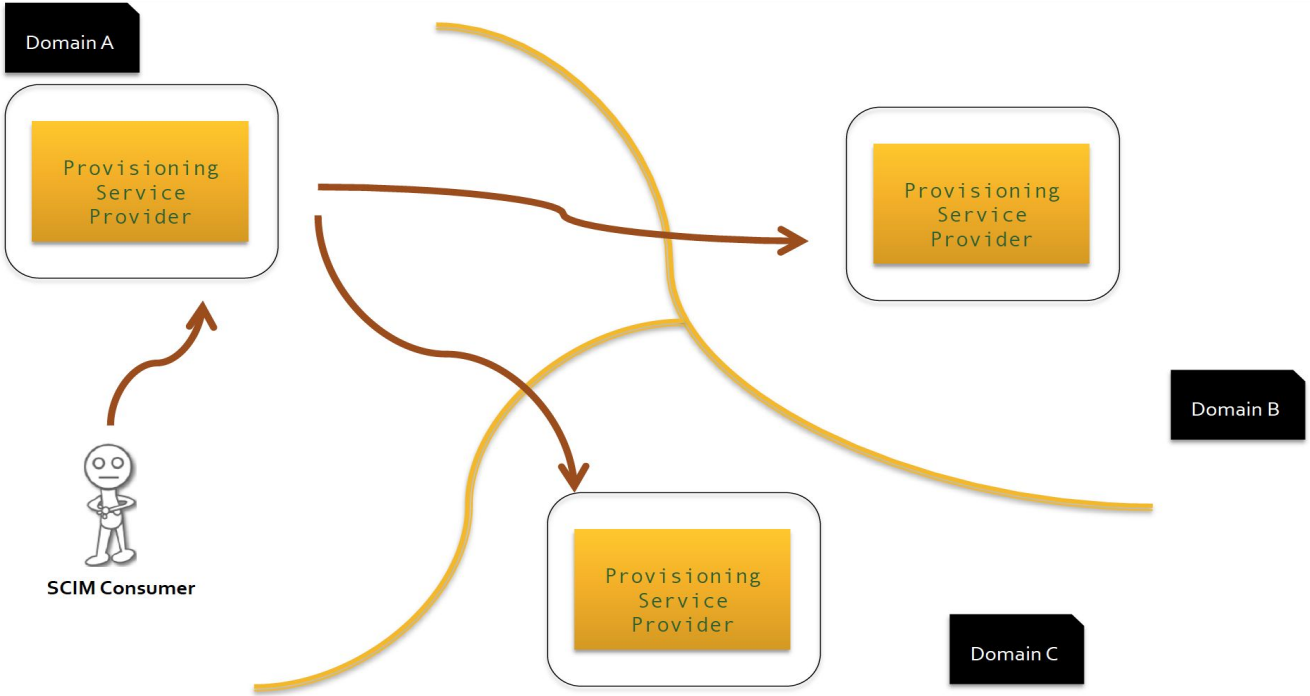
AUTHENTICATING SCIM REQUESTS



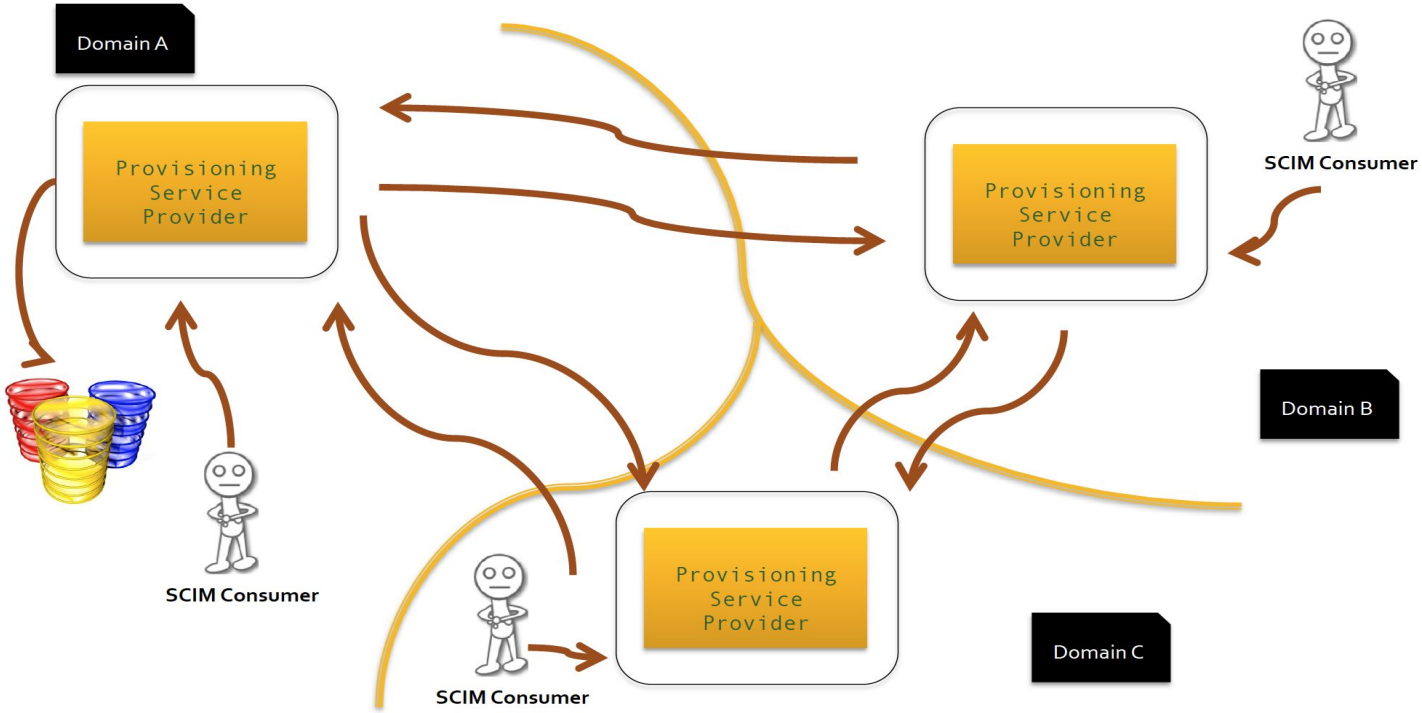
FEDERATED PROVISIONING PATTERNS



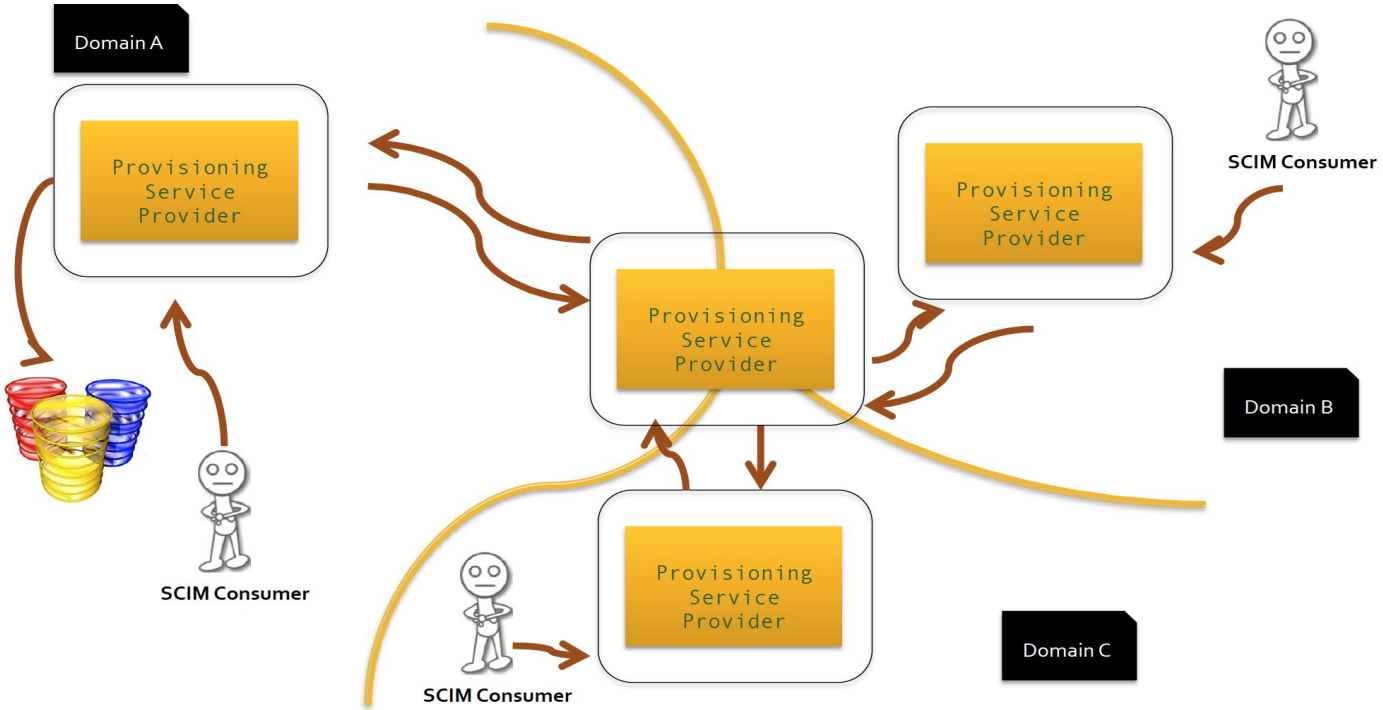
FEDERATED PROVISIONING PATTERNS



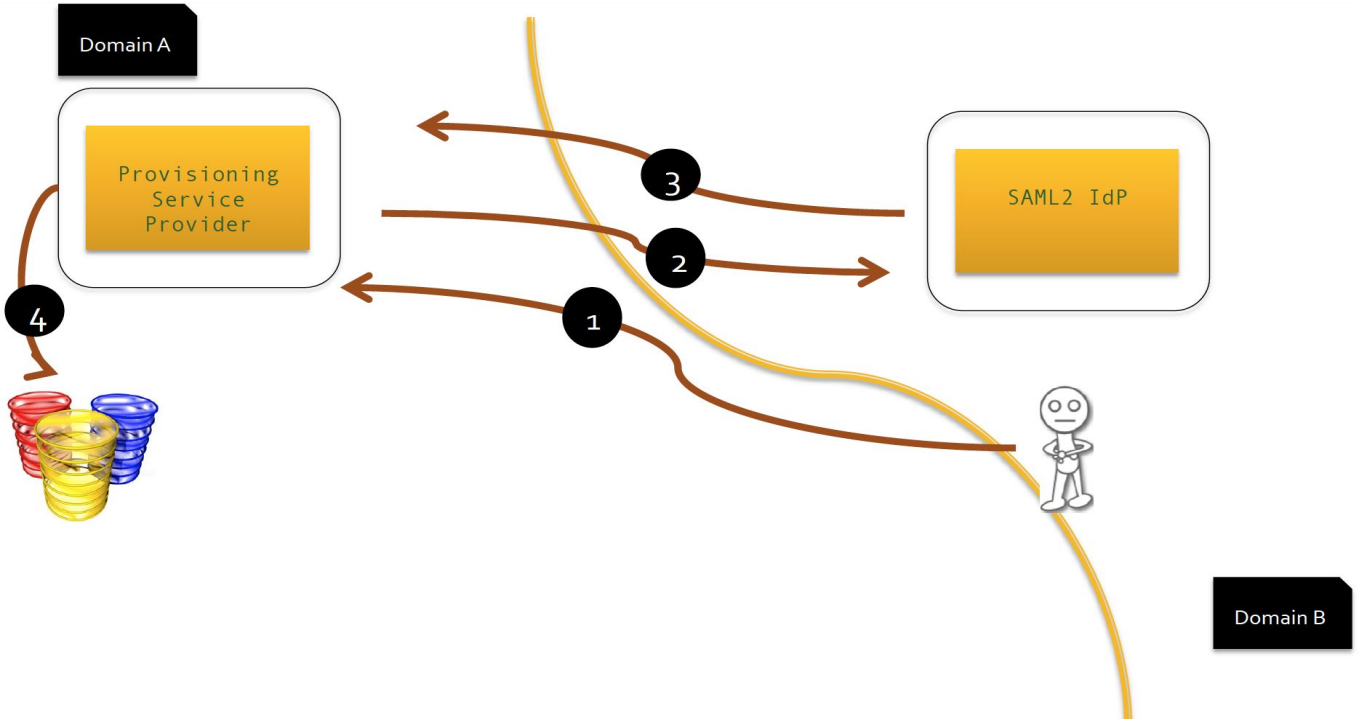
FEDERATED PROVISIONING PATTERNS



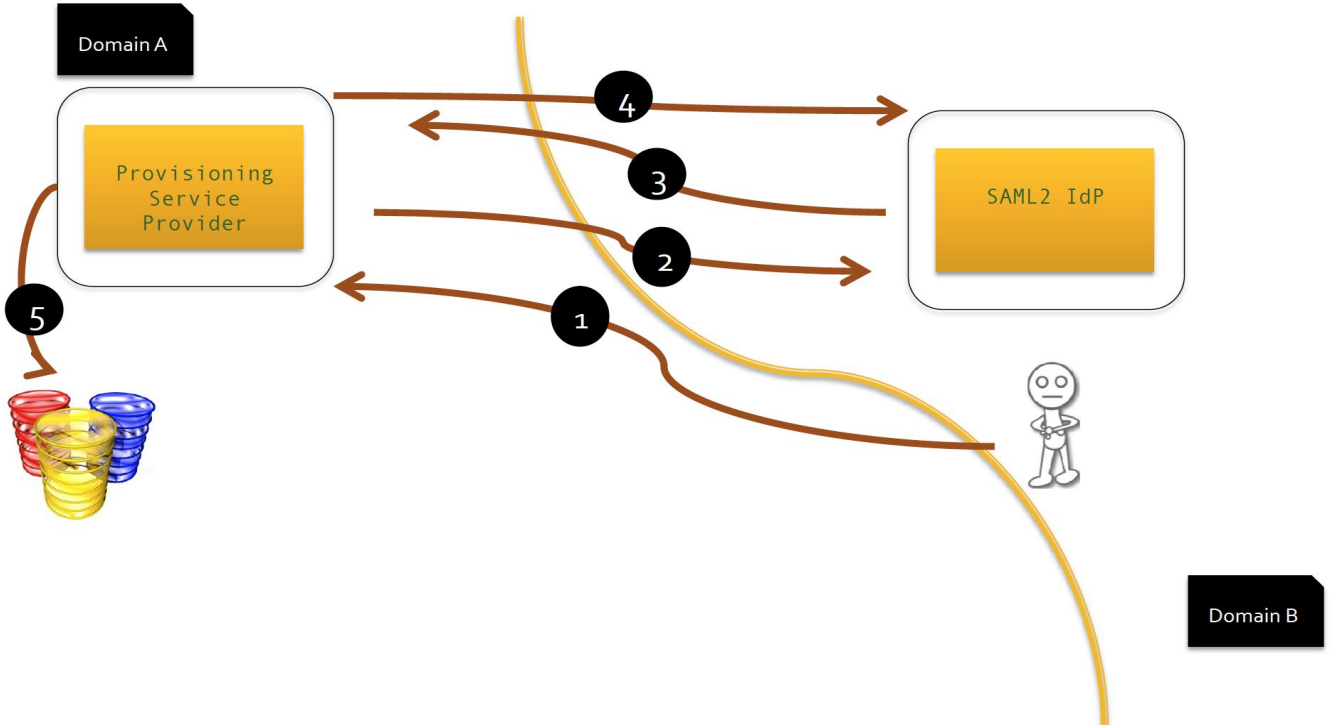
FEDERATED PROVISIONING PATTERNS



FEDERATED PROVISIONING PATTERNS



FEDERATED PROVISIONING PATTERNS



PROVISIONING SYSTEM CAPABILITIES

- Automated provision for employees
- Automated provision for all staff including contractors
- Zero-day start and stop for all staff
- Role-based assignment of identity attributes
- Information architecture in place for identity data

ACCESS CONTROL

OVERVIEW

- Permission : A capability / Negative permissions are hard
- Role : A set of permissions
- Group : A set of users
- Role Based Access Control (RBAC) : Make access control decisions based on roles
- Attribute Based Access Control (ABAC) : Make access control decisions based on attributes

ROLE BASED ACCESS CONTROL

- Flat RBAC

  Permissions are assigned to a role and a role is assigned to user (a group of users)

  Many to many relationship between user to role. Many to many relationship between permission to role.

- Hierarchical RBAC

  Senior role and Junior roles.

  A senior role acquires all the permissions belong to its junior roles.

  A user can perform a given task if he/she inherits the required permissions, either from a role he/she belongs to or from any other juniors roles.

ROLE BASED ACCESS CONTROL

- Constrained RBAC

  Enforce Separation of Duties (SoD) / Separation of Duties (SoD) spreads authority and responsibility for an action or a task over multiple users.

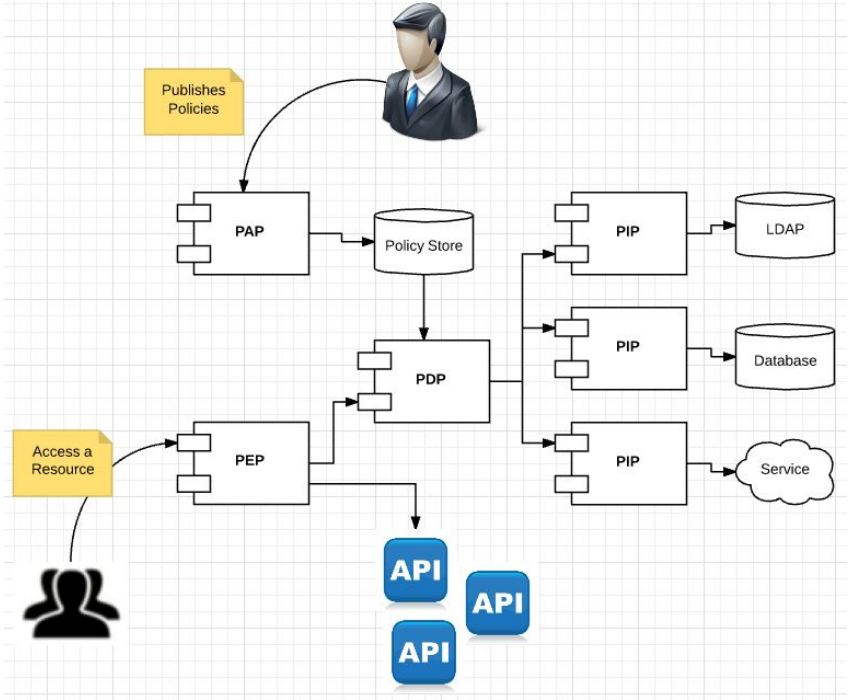
- Symmetric RBAC

  Supports permission-role review.

XACML OVERVIEW

- Fine-grained Access Control
- Requirements from Health Care, DRM, Registry, Financial, Online Web
- XACML 1.0 - OASIS Standard – 6 February 2003
- XACML 1.1 – Committee Specification – 7th August 2003
- XACML 2.0 – OASIS Standard – 1 February 2005
- XACML 3.0 – OASIS Standard – 10th Aug 2010

XACML REFERENCE ARCHITECTURE



XACML POLICY LANGUAGE

- XML based
- Represents access control logic in rules
- A given XACML policy can have multiple rules
- A XACML engine can have multiple XACML policies
- Only the XACML policies applicable to a given XACML request will be evaluated.

XACML POLICY LANGUAGE

- The smallest execution unit in a XACML policy is a Rule
- A Rule can return back Permit or Deny
- Rule combining algorithms decide how to combine multiple decisions from multiple Rules
- The policy combining algorithms decide how to combine multiple decisions from multiple policies.
- Obligations and Advices

XACML REQUEST/RESPONSE PROTOCOL

- The XACML core specification defines XML based schema for the XACML request and response.
- JSON Profile for XACML define XACML request and response in JSON
- The REST profile XACML define how to invoke the XACML PDP in a RESTful manner.
- Multiple decisions

IDENTITY GOVERNANCE AND ADMINISTRATION

OVERVIEW

- Identity governance and administration (IGA) solutions manage identity and access life cycles across multiple systems.
- Automated provisioning of accounts among heterogeneous systems.
- Fulfillment of access requests (including self-service)
- Password management
- Governance over user access to target systems via workflows and automated policies.
- Risk scoring of a user's combined entitlements
- Segregation of duties (SOD) enforcement

ROLE MANAGEMENT AND ROLE MINING

- Configuring a Role-Based Access-Control (RBAC) system, i.e., creating roles and assigning users to roles and permissions to roles.
- The term “role mining” is used in a more narrow sense to refer to automated approaches to role engineering. At the very core, it is about identifying and extracting meaningful “roles” in an enterprise from “row data” (e.g. access control rights, ACLs, etc.) by using different techniques.
- The role mining process discovers relationships between users based on similar access permissions that can logically be grouped to form a role.
- Role engineers can specify the applications and attributes that will return the best mining results.

ROLE CONSOLIDATION

- Prevents the creation of new roles with almost the same membership and entitlements of existing role
- Role Consolidation tells how similar two roles are based on the two criteria: role membership and entitlements

USER ENTITY BEHAVIOR ANALYTICS (UEBA)

- UEBA is separate area of study, which focuses on analyzing the behaviors of organizations' insiders (employees), outsiders connected to their networks (such as third party contractors) and flagging security vulnerabilities across organizations' assets that hold sensitive data.

WORKFORCE IAM VS CUSTOMER IAM

OVERVIEW

- The workforce IAM looks inward. It focuses on B2E (business-to-employee) and B2B (business-to-business) interactions.
- The goal of workforce IAM is to reduce the risk and cost associated with on-boarding and off-boarding new employees, partners and suppliers.
- The purpose of customer IAM (CIAM) is to help drive revenue growth by leveraging identity data to acquire and retain customers.
- If CIAM processes are cumbersome, customers will go to your competition where these processes are more streamlined or easier to use. The same is not true of employees. Very few employees leave their employer because business-to-employee (B2E) IAM processes are archaic or hard to use.

CUSTOMER IAM

- User onboarding
- Progressive profiling
- Multi-factor authentication
- Self-Service
- CxO Dashboard

CUSTOMER IAM

- Security, compliance & fraud detection
- Omnichannel access
- Help desk & delegated administration
- Scalability
- APIs and integration

IAM DESIGN PRINCIPLES

IAM DESIGN PRINCIPLES

- Immutable private identifiers / mutable public identifiers
- Decouple core/static personally identifiable information (PII) from transactional data
- Decouple biometrics from other personally identifiable information (PII)
- Externalize access control rules
- Low trust panes have no write access

IAM DESIGN PRINCIPLES

- Decouple B2C from B2E and B2B
- Persistent pseudonyms for attribute sharing
- Standards rule!. feel free to fix it — but do not break!
- Self-expressive credentials
- Privilege accounts are a different species

IAM SOLUTIONS

PROJECT - 1

- More than 50 applications maintain their own user repositories for user authentication and authorization.
- Same employee record is duplicated across different departments, no correlation handle.
- Build the IAM architecture for a unified identity platform.

PROJECT - 2

- An organization has both customers and employees.
- There are employee only applications and applications both customers and employees can access.
- Build the IAM architecture for onboarding and login.

PROJECT - 3

- A digital aviation platform has multiple airlines, suppliers and other vendors.
- Suppliers and vendors can serve multiple airlines.
- Build the IAM architecture for onboarding and login.

PROJECT - 4

- A company has multiple service providers - and each service provider has its own attribute requirements for user sign up.
- Service providers are open to the public as well as to its own employees.
- Build the IAM architecture for onboarding and login.

PROJECT - 5

- A company has its own SaaS app used by its customers.
- Each customer has its own identity provider.
- More than one customers can share the same external identity provider.
- Build the IAM architecture for onboarding and login.

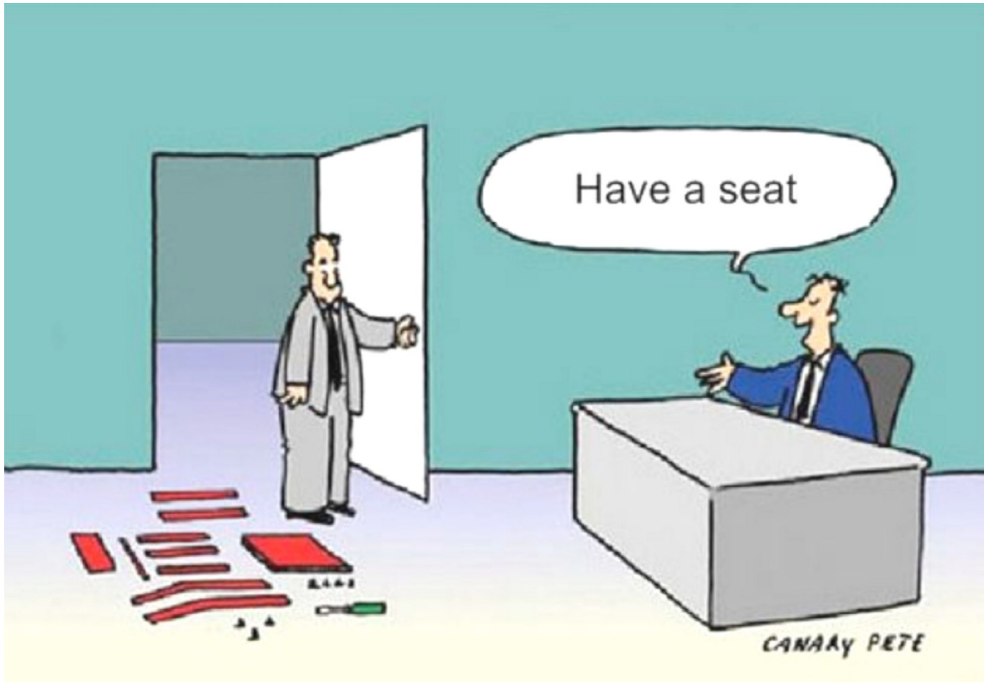
WHY IAM PROJECTS FAIL?

ALL YOU CAN EAT RFPs/RFIs

DISCONNECTED DECISION MAKERS

SEED WHAT HARVESTS SOON

DO IT MYSELF! I BUILD MY OWN IAM!



BUZZWORD LOVERS

THE GREEN SWAN

SWISS ARMY KNIFE

SHORT SIGHTED

OPERATING IN SILOS

VENDOR LOCKING

PRIVACY

LESSONS LEARNT FROM GDPR

- The EU General Data Protection Regulation (GDPR) is the regulation 2016/679 of the European parliament and of the council.
- Designed to harmonize data privacy laws across Europe, to protect and empower all EU citizens (and residents) data privacy and to reshape the way organizations across the region approach data privacy.
- Effective from 25th May 2018.
- GDPR became quite prominent due to the heavy penalties introduced by it for violators — which could be as much as 4% of the annual global turnover or €20 Million (whichever is greater).

GDPR LESISSONS LEARNED

- Capture only the minimal personal data
- Must not capture personal data for the anticipated usage in the future
- Record the user consent on privacy/data policy
- Record user interactions/audits against a pseudonym - and main the user/ pseudonym mapping separately. On request to delete user information, remove the mapping to make all the other data store about the user anonymous.
- Self-service user portal / consent management
- Strong authentication for IAM administrators

GDPR LESISSONS LEARNED

- Audit all the actions in the IAM infrastructure — not just by IAM administrators — but also by the system administrators
- Occupy identity analytics to detect anomalous activities and fire notifications
- Do not share the IAM infrastructure between customers (users) and corporate employees.

THANK YOU

wso2.com