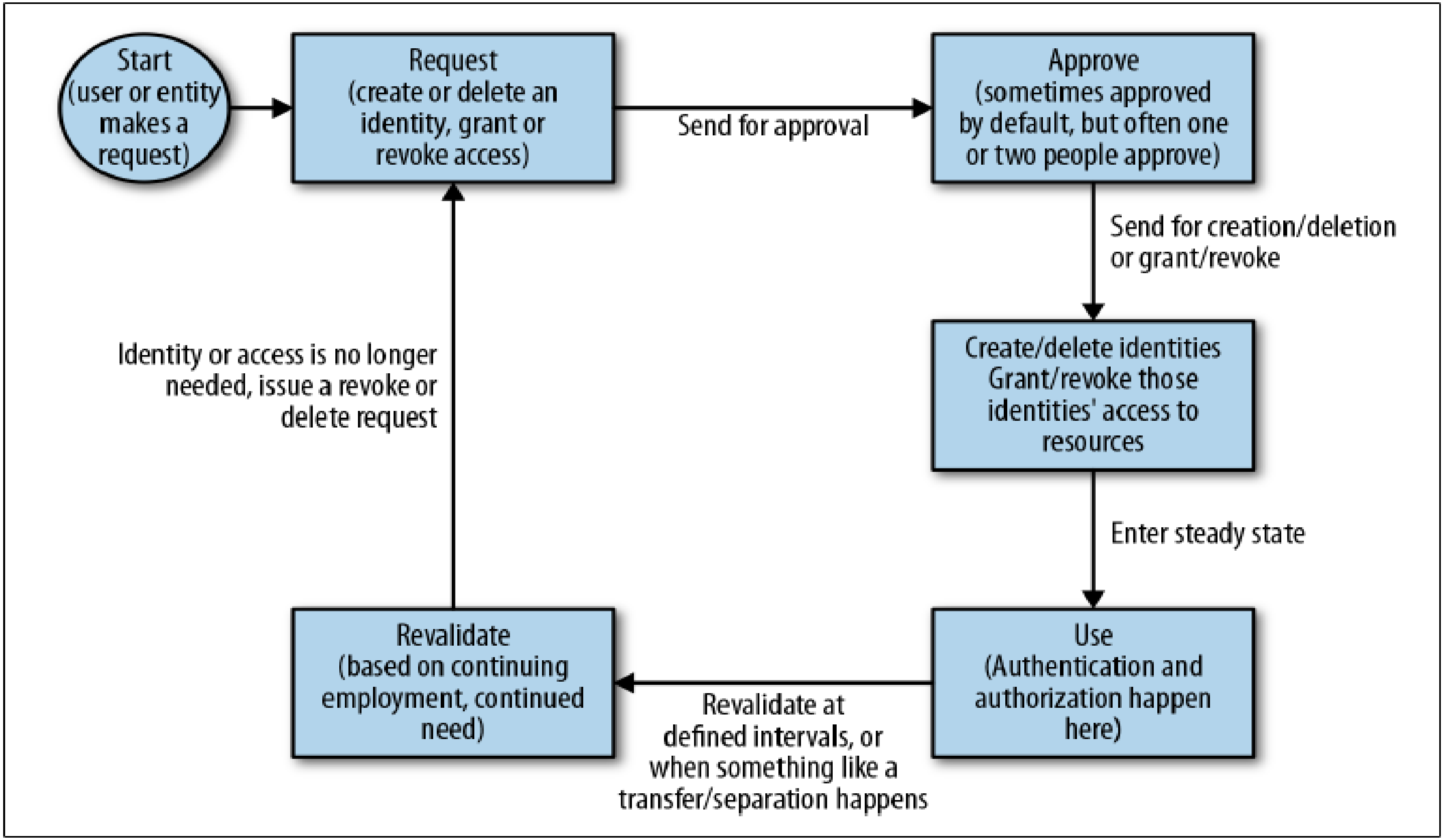
IAM in Cloud Computing

Dr. Amit Praseed

Identity and Access Management

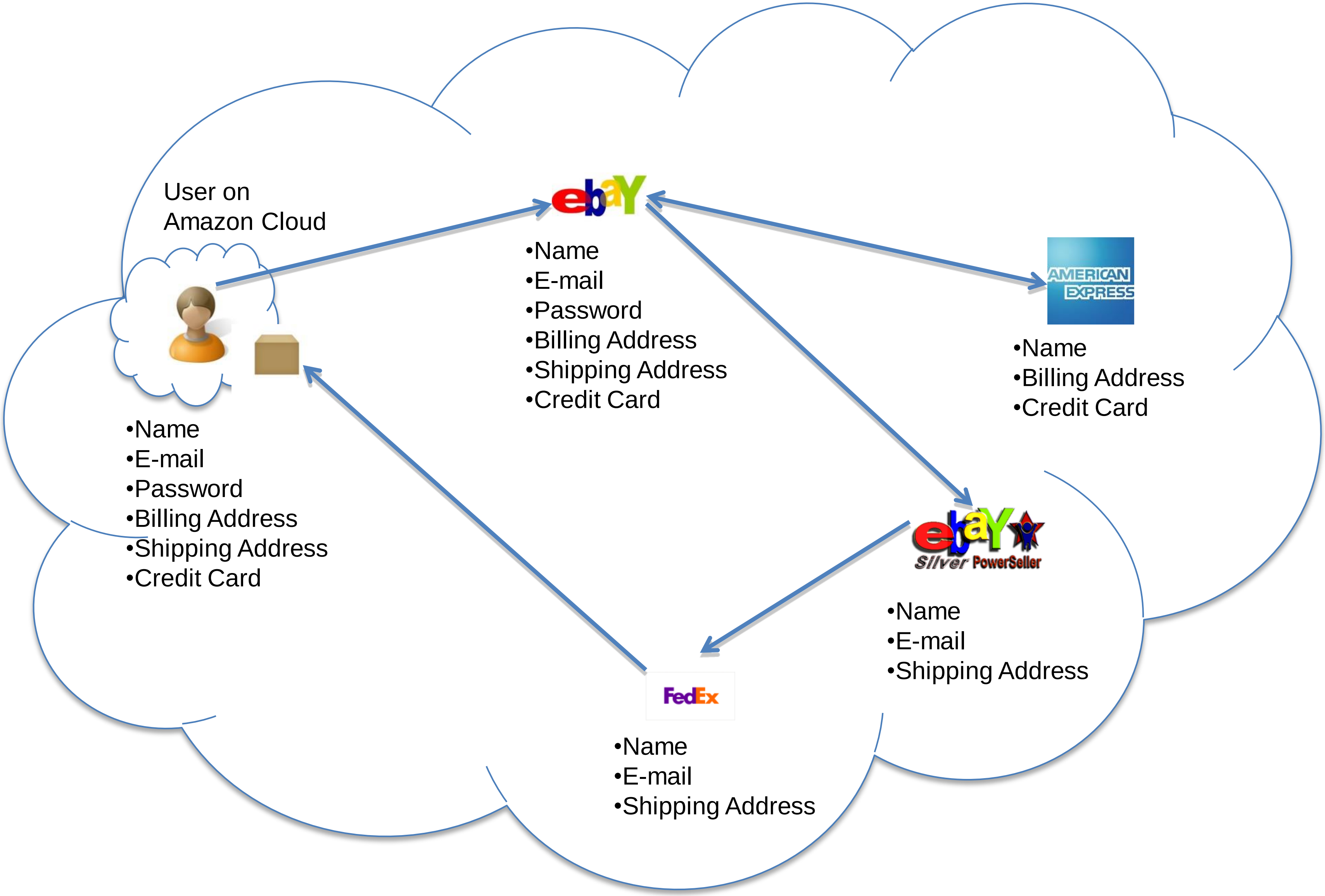
- Each entity (such as a user, administrator, or system) needs an identity
  - The process of verifying that identity is called *authentication*
- Access management is about ensuring that entities can perform only the tasks they need to perform.
  - The process of checking what access an entity should have is called *authorization*

IAM Life Cycle



Start (user or entity makes a request)
Request (create or delete an identity, grant or revoke access)
Send for approval
Approve (sometimes approved by default, but often one or two people approve)
Send for creation/deletion or grant/revoke
Create/delete identities Grant/revoke those identities' access to resources
Enter steady state
Use (Authentication and authorization happen here)
Revalidate at defined intervals, or when something like a transfer/separation happens
Revalidate (based on continuing employment, continued need)
Identity or access is no longer needed, issue a revoke or delete request

So many Identities!!!



User on
Amazon Cloud

- Name
- E-mail
- Password
- Billing Address
- Shipping Address
- Credit Card

- Name
- E-mail
- Password
- Billing Address
- Shipping Address
- Credit Card

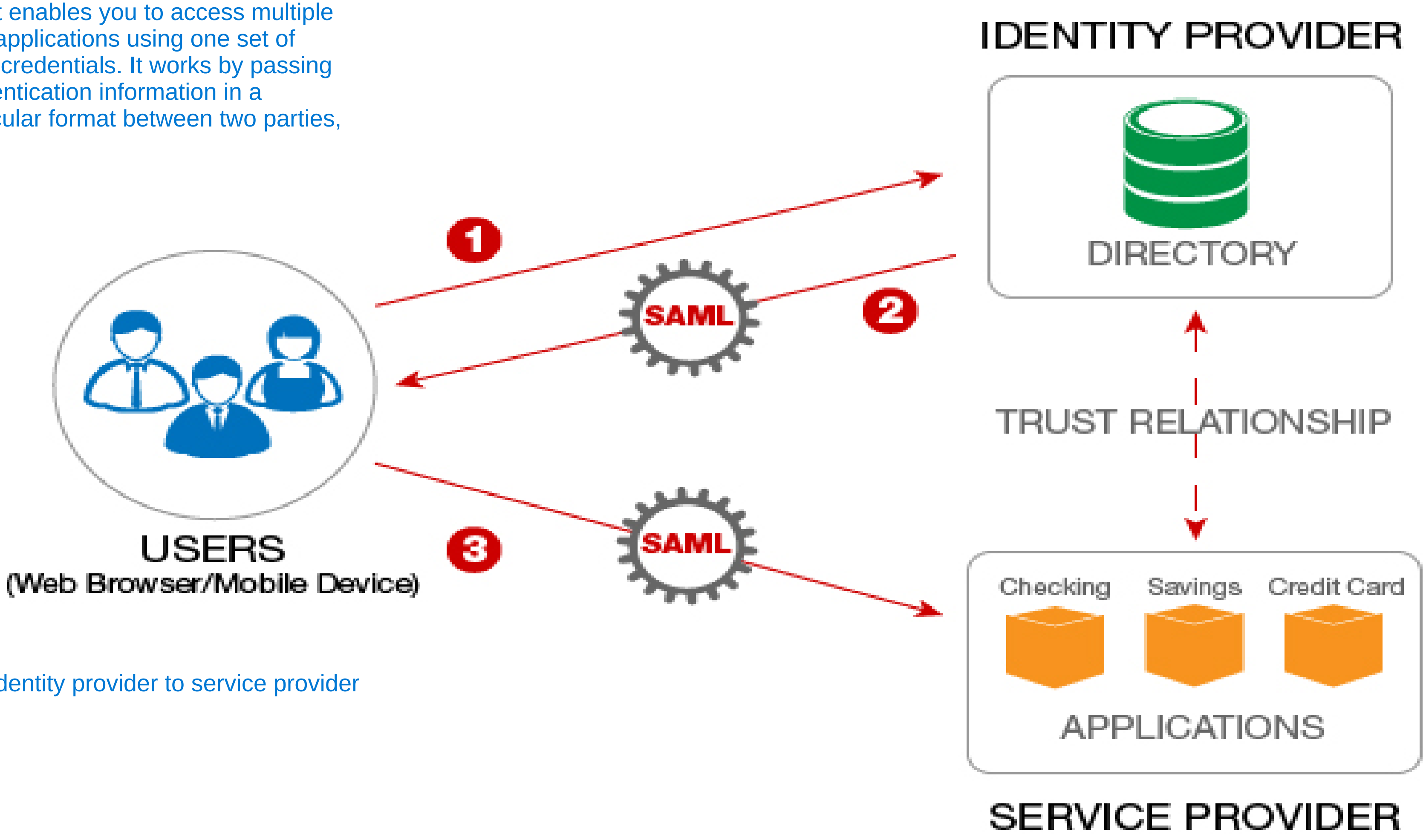
- Name
- Billing Address
- Credit Card

- Name
- E-mail
- Shipping Address

- Name
- E-mail
- Shipping Address

SSO to the Rescue!

Its primary role in online security is that it enables you to access multiple web applications using one set of login credentials. It works by passing authentication information in a particular format between two parties,



Identity provider to service provider

Protocols for SSO

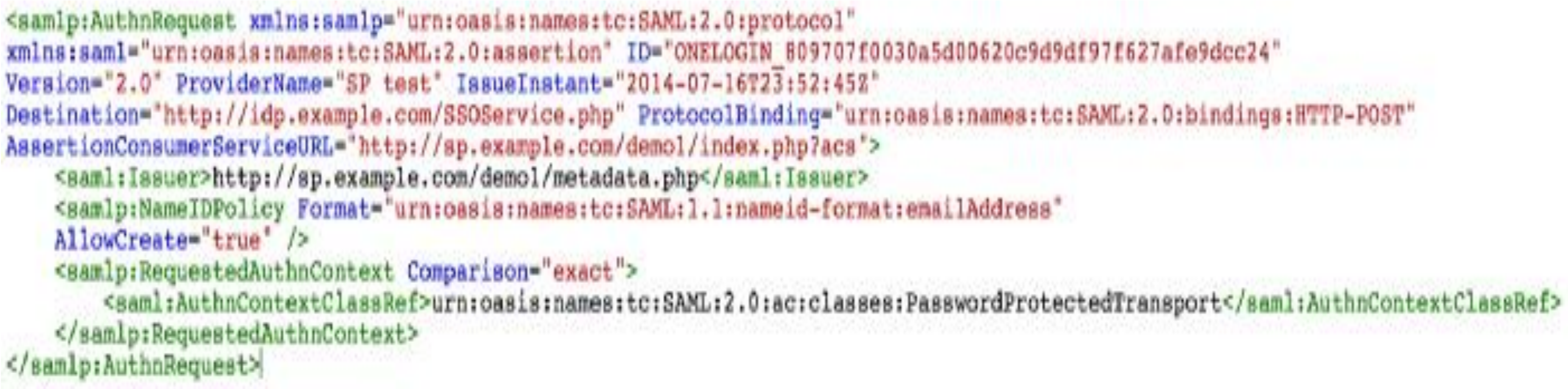
- There are three popular mechanisms that are used to provide SSO
  - Security Assertion Markup Language (SAML)
  - Open Authorization (OAuth)
  - OpenID

OAuth and SAML are both protocols we use for allowing access. However, the primary difference between the two is that we use SAML for authentication and OAuth for authorization.

SAML is an open standard used for authentication
XML USED
web applications use SAML to transfer authentication data between two parties - the identity provider (IdP) and the service provider (SP).

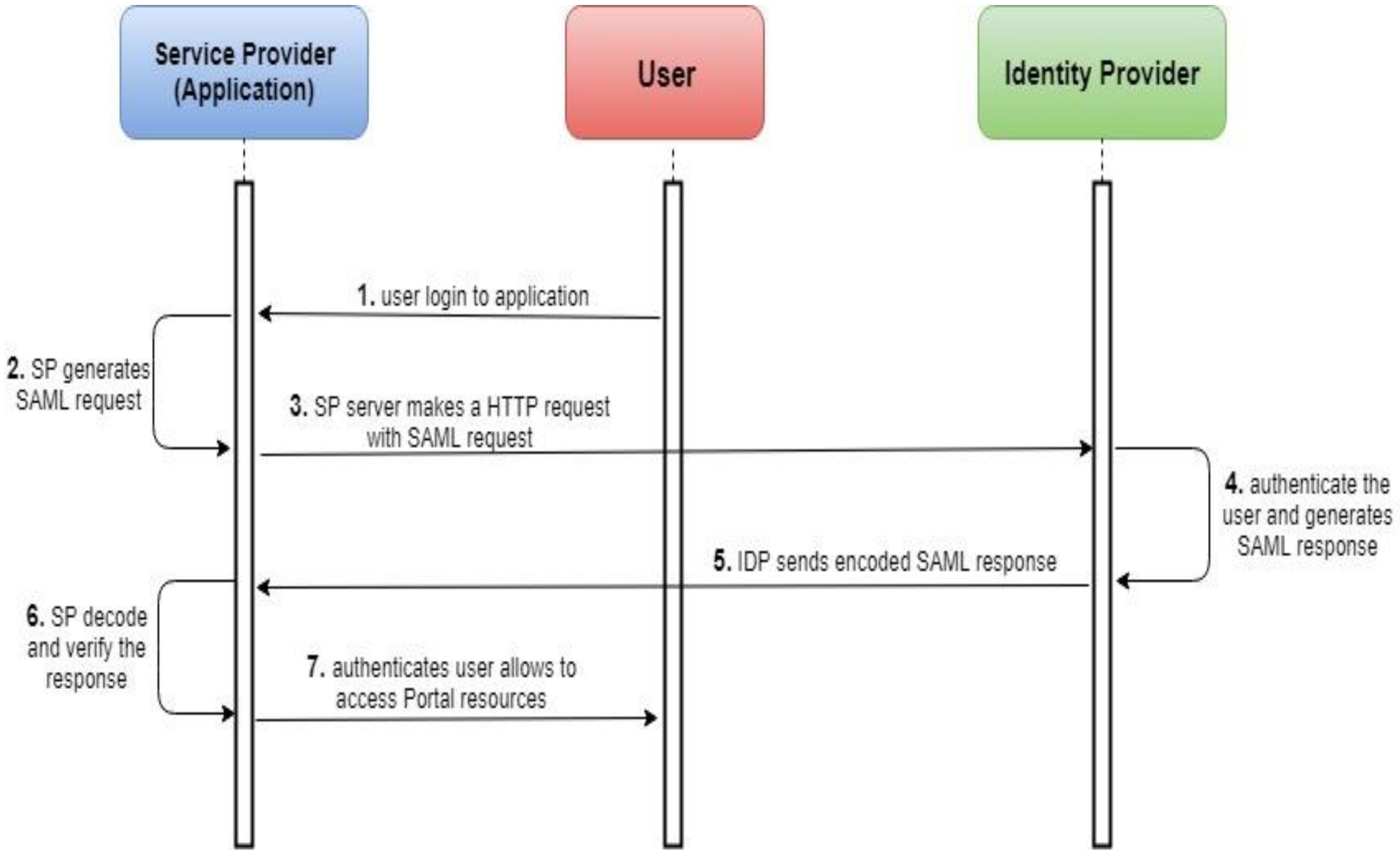
SAML

- SAML was developed in the early 2000s
  - define an XML framework for exchanging authentication and authorization information
  - allows a user’s identity to be passed from one place to another with digitally signed XML (eXtensible Markup Language) documents.
  - SAML Info: Version, ID, ProviderName, IssueInstant, Destination, ProtocolBinding, AssertionConsumerServiceURL, and Issuer



```
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" ID="ONELOGIN_809707f0030a5d00620c9d9df97f627afe9dcc24"
Version="2.0" ProviderName="SP test" IssueInstant="2014-07-16T23:52:45Z"
Destination="http://idp.example.com/SSOService.php" ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
AssertionConsumerServiceURL="http://sp.example.com/demo1/index.php?acs">
    <saml:Issuer>http://sp.example.com/demo1/metadata.php</saml:Issuer>
    <samlp:NameIDPolicy Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress"
    AllowCreate="true" />
    <samlp:RequestedAuthnContext Comparison="exact">
        <saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport</saml:AuthnContextClassRef>
    </samlp:RequestedAuthnContext>
</samlp:AuthnRequest>
```

SAML Workflow



Service Provider (Application)

User

Identity Provider

1. user login to application

2. SP generates SAML request

3. SP server makes a HTTP request with SAML request

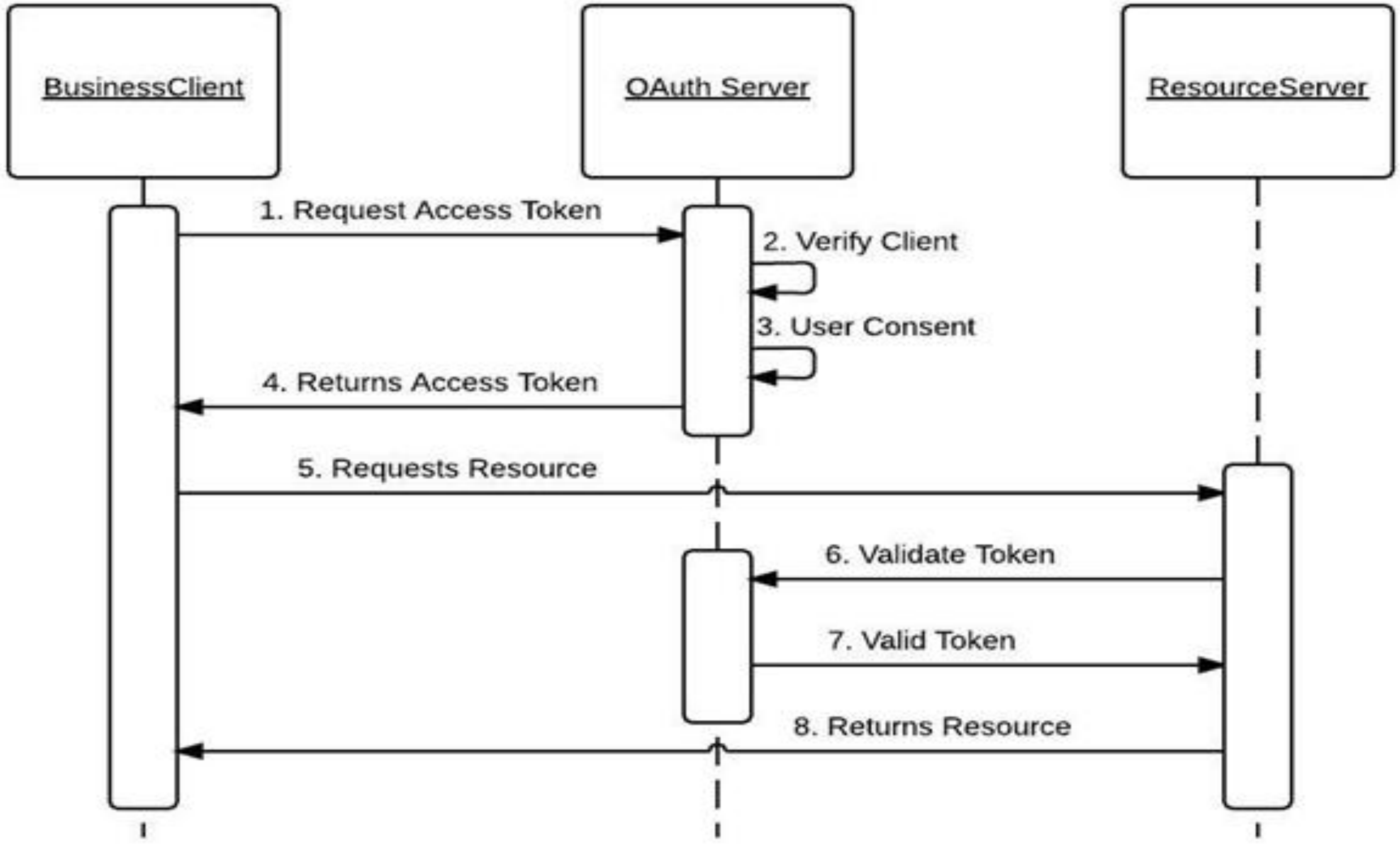
4. authenticate the user and generates SAML response

5. IDP sends encoded SAML response

6. SP decode and verify the response

7. authenticates user allows to access Portal resources

OAuth Workflow



SSO and Privacy

- Cloud introduces several issues to IDM
  - Collusion between Cloud Services
    - Users have multiple accounts associated with multiple service providers.
    - Sharing sensitive identity information between services can lead to undesirable mapping of the identities to the user.
  - Lack of trust
    - Cloud hosts are untrusted
    - Use of Trusted Third Party is not an option
  - Loss of control
    - Service-centric IDM Model

SSO and Privacy

- Anonymous Identification
  - Based on cryptographic zero knowledge proofs
- Multi party Authentication
  - Based on secure multi party computation
  - Information is distributed and managed by multiple identity providers, all of whom hold non-overlapping information
  - Unless *k* IdPs collude, user information cannot be effectively leaked