

MODULE-3

Identity and Access Management

- **THIS CHAPTER PRESENTS THE CURRENT STATE OF THE PRACTICE OF**
- **IDENTITY AND ACCESS** management (IAM) and support for IAM features that aid in Authentication, Authorization, and Auditing (AAA) of users accessing cloud services.

Trust Boundaries and IAM

1. Traditional Trust Boundaries:

- In traditional IT environments, the trust boundary is static, managed by the IT department, and includes networks, systems, and applications.
- IT uses tools like VPNs, IDS/IPS, and multifactor authentication for security within this boundary.

2. Impact of Cloud Adoption:

- Cloud computing makes trust boundaries dynamic and moves them beyond IT's control, extending into the service provider's domain.
- This challenges traditional governance models, especially for large enterprises engaged in e-commerce, outsourcing, and supply chains.

3. Compensation for Network Control Loss:

- Organizations need higher-level software controls like application security, user access controls, and identity management to compensate for reduced control over networks.

4. Key Controls for Cloud Security:

- Strong authentication, role-based authorization, identity federation, SSO, user monitoring, and auditing are critical to ensuring security in cloud environments.

5. Importance of Identity Federation:

- Federation enables interaction between systems outside the traditional trust boundary and strengthens trust between organizations and cloud service providers.
- It supports strong authentication, web SSO, and centralized access control, which are vital for cloud adoption.

6. Challenges in IAM Practices:

- Poorly managed IAM within an organization, such as lack of governance or manual identity management, can weaken cloud security and extend bad practices.

7. IAM as a Two-Way Street:

- Both organizations and cloud service providers must support IAM standards like SAML for federation and compliance with internal policies.

8. Benefits of Well-Implemented IAM:

- Proper IAM practices enhance confidentiality, integrity, and compliance in cloud environments.
- CSPs supporting IAM standards accelerate cloud adoption and IT migration to cloud services.



SAML (Security Assertion Markup Language)

- **Definition:** SAML is an open standard for exchanging authentication and authorization data between parties, particularly between an identity provider (IdP) and a service provider (SP).
- **Functionality:**
 - **Single Sign-On (SSO):** Allows users to log in once and gain access to multiple applications without needing to authenticate again.
 - **Assertions:** SAML enables the creation of assertions that convey user identity and attributes from the IdP to the SP.
- **Use Cases:** Commonly used in enterprise environments to enable SSO across web applications and cloud services.

Why IAM?

1. IAM and Operational Efficiency:

- IAM improves efficiency by automating user onboarding and repetitive tasks (e.g., self-service for password resets).
- Reduces reliance on manual processes such as help desk tickets for common user issues.

2. IAM for Regulatory Compliance:

- Helps protect systems, applications, and data from internal/external threats (e.g., disgruntled employees).
- Supports compliance with regulatory requirements like HIPAA, SOX, and PCI DSS.
- Uses frameworks like ISO 27002 and ITIL for implementing access control and operational security.
- Assists auditors by mapping internal controls to IT controls for managing compliance.

1. HIPAA (Health Insurance Portability and Accountability Act):

- A U.S. law enacted in 1996 to protect sensitive patient health information.
- Requires organizations dealing with healthcare data to implement safeguards for the confidentiality, integrity, and availability of protected health information (PHI).
- Compliance ensures that healthcare organizations secure electronic health records (EHRs), preventing data breaches and unauthorized access.

2. SOX (Sarbanes-Oxley Act):

- A U.S. federal law passed in 2002 to protect investors by improving the accuracy and reliability of corporate financial reporting.
- Applies to publicly traded companies and focuses on corporate governance, financial transparency, and internal controls over financial reporting.
- SOX compliance requires organizations to implement controls for data security and to ensure the integrity of financial systems and processes.

3. PCI DSS (Payment Card Industry Data Security Standard):

- A set of security standards designed to ensure that all organizations that handle, process, or store credit card information maintain a secure environment.
- Enforced by major credit card brands (e.g., Visa, MasterCard) and applies to any organization that handles card transactions.
- Compliance involves securing cardholder data, ensuring strong access controls, and regularly monitoring and testing networks to prevent fraud or breaches.

- **Web Single Sign-On (Web SSO)** is a user authentication process that allows users to log in once and gain access to multiple related applications or services without needing to re-enter credentials for each one.

5. SSO and Rapid Cloud Adoption:

- SSO enables businesses to externalize authentication, reducing the time needed to integrate with service providers (e.g., Salesforce.com).
- IAM enables outsourcing processes securely by extending user access management practices to partners.

6. Rapid Cloud Service Adoption:

- Organizations with strong IAM practices can quickly adopt cloud services while maintaining security and operational efficiency.
- IAM ensures that security controls remain effective even when services are outsourced or migrated to the cloud.

IAM Challenges

1. Diverse User Populations:

- Organizations manage access for various user groups (employees, contractors, partners), each with different needs and access rights.
- Changes in roles and responsibilities for business reasons complicate rapid provisioning of access.

2. Turnover and Staffing Fluctuations:

- High user turnover and seasonal changes (e.g., finance or retail) create frequent updates to access rights.
- Business changes like mergers, acquisitions, outsourcing, and new product releases also affect user access management.

3. **Decentralized Access Policies:**

- Organizations often apply inconsistent access policies, with multiple directories and disparate identity systems, leading to inefficiencies.
- This complexity in identity management increases security, compliance, and reputational risks.

4. **Complexity of IAM Initiatives:**

- Organizations turn to technology solutions to centralize and automate user access management.
- IAM initiatives are often large, costly, and time-consuming, requiring a balance between business and IT drivers to ensure efficiency and security.

5. **Long-Term Commitment:**

- Due to the complexity and cost of IAM initiatives, organizations must carefully plan their strategy to address inefficiencies while maintaining effective access control.

IAM (Identity and Access Management) definitions for key functions:

1. Authentication:

- The process of verifying the identity of a user or system.
- Examples include verifying credentials through protocols like **LDAP** for user ID authentication or service-to-service authentication (e.g., a travel service verifying a credit card through a gateway).
- Authentication ensures that the entity accessing the system is who they claim to be.

2. Authorization:

- The process of determining what privileges or actions a user or system is allowed to perform after authentication.
- This step enforces security policies to ensure users or systems only have access to the resources they are permitted to interact with.

- **LDAP (Lightweight Directory Access Protocol)** is an open, vendor-neutral protocol used for accessing and maintaining distributed directory information services over an IP network.
- Directory services provide a centralized way to store information about users, systems, networks, and applications in an organized, hierarchical manner.
- LDAP is widely used for authentication and authorization, particularly in corporate environments.

3. Auditing:

- Involves reviewing and examining authentication and authorization records.
- It ensures IAM controls are adequate, checks for policy compliance (e.g., separation of duties), detects breaches (e.g., privilege escalation), and recommends countermeasures if needed.

IAM Architecture and Practice

1. IAM as a Multi-Layered Solution:

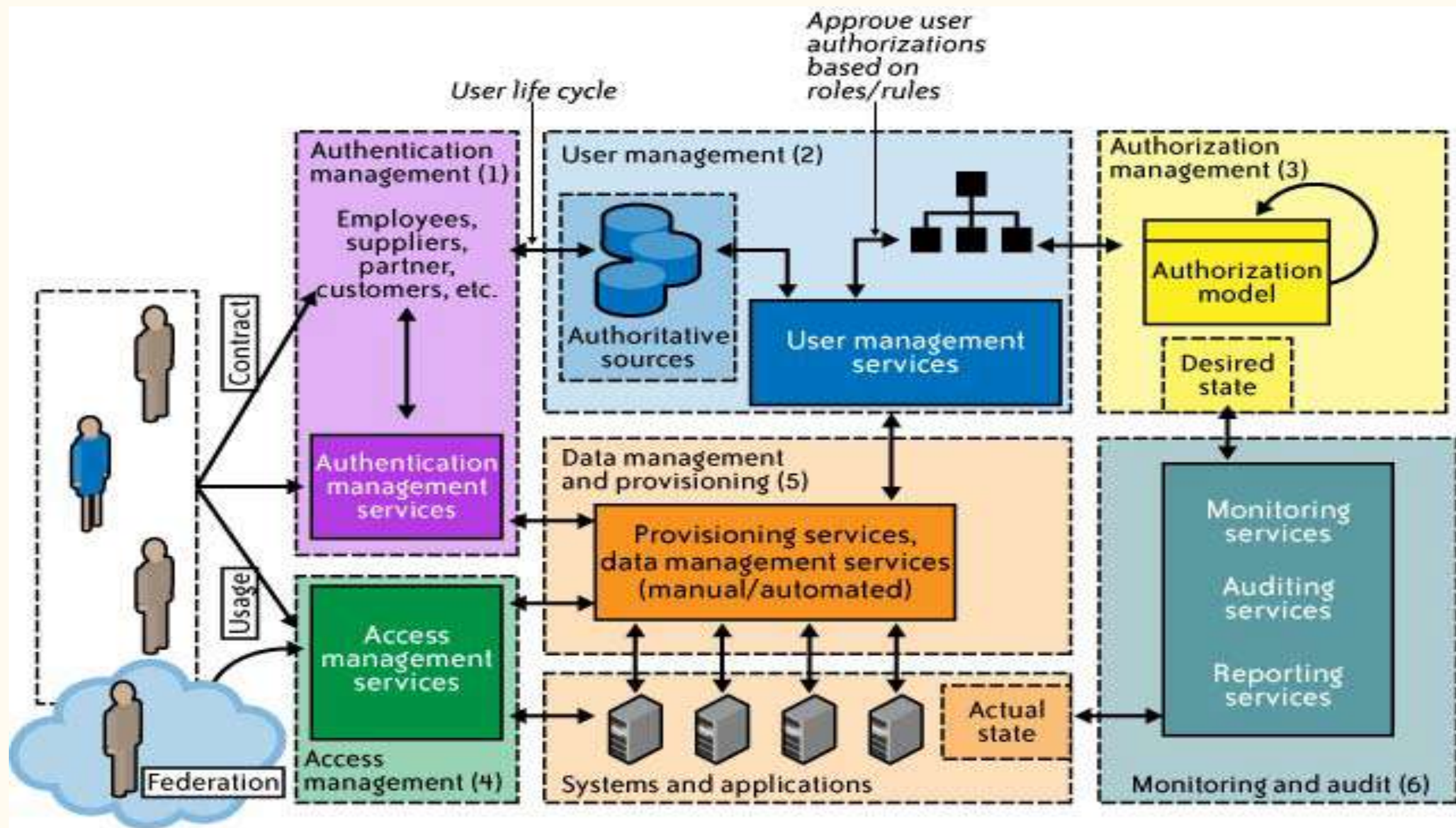
- IAM is not a simple, monolithic system. It involves a mix of architecture, technology components, processes, and standard practices.
- Core technology includes a **directory service** (e.g., LDAP, Active Directory) as the central repository for user identities, credentials, and attributes.
- The directory interacts with other IAM components, such as **authentication, user management, provisioning, and federation services**, to support standard IAM processes.

2. Multiple Directories:

- Organizations may have multiple directories (e.g., Active Directory for Windows systems, LDAP for Unix) due to different environments or through mergers and acquisitions.

3. Key IAM Processes:

- **User Management:** Governs and manages the life cycle of identities within the organization.
- **Authentication Management:** Governs the process of verifying that an entity is who or what it claims to be.
- **Authorization Management:** Governs the process of determining what resources an entity can access, based on entitlements and policies.
- **Access Management:** Enforces access control policies when a user or service requests access to IT resources.
- **Data Management and Provisioning:** Manages the propagation of identity and authorization data to IT resources, either automatically or manually.
- **Monitoring and Auditing:** Involves monitoring, auditing, and reporting user compliance with access policies.



IAM processes support the following operational activities:

1. Provisioning:

- **On-boarding users** to systems and applications by assigning access to data, systems, applications, and databases based on user identity.
- **Deprovisioning** works in reverse, deactivating or deleting user identities and privileges.
- It's an essential aspect of resource management, combining IT and HR duties.

2. Credential and Attribute Management:

- **Credential management** involves creating, issuing, managing, and revoking user credentials (e.g., passwords, tokens) to prevent unauthorized access.
- **Attribute management** manages the life cycle of user attributes and enforces privacy and regulatory handling of these attributes.
- Ensures credentials are secure both in transit and at rest, including handling expiration and encryption.



3. Entitlement Management:

- **Entitlements** refer to the authorization policies determining user privileges to access systems, applications, and databases.
- Ensures users have only the necessary privileges for their roles (principle of least privilege), enhancing security for web services, applications, and physical systems.

4. Compliance Management:

- **Monitoring and tracking access rights** to ensure resources are secured and compliant with internal policies and regulations (e.g., segregation of duties, access monitoring).
- Facilitates periodic auditing and reporting to support regulatory compliance and verifies access through processes like user certification.

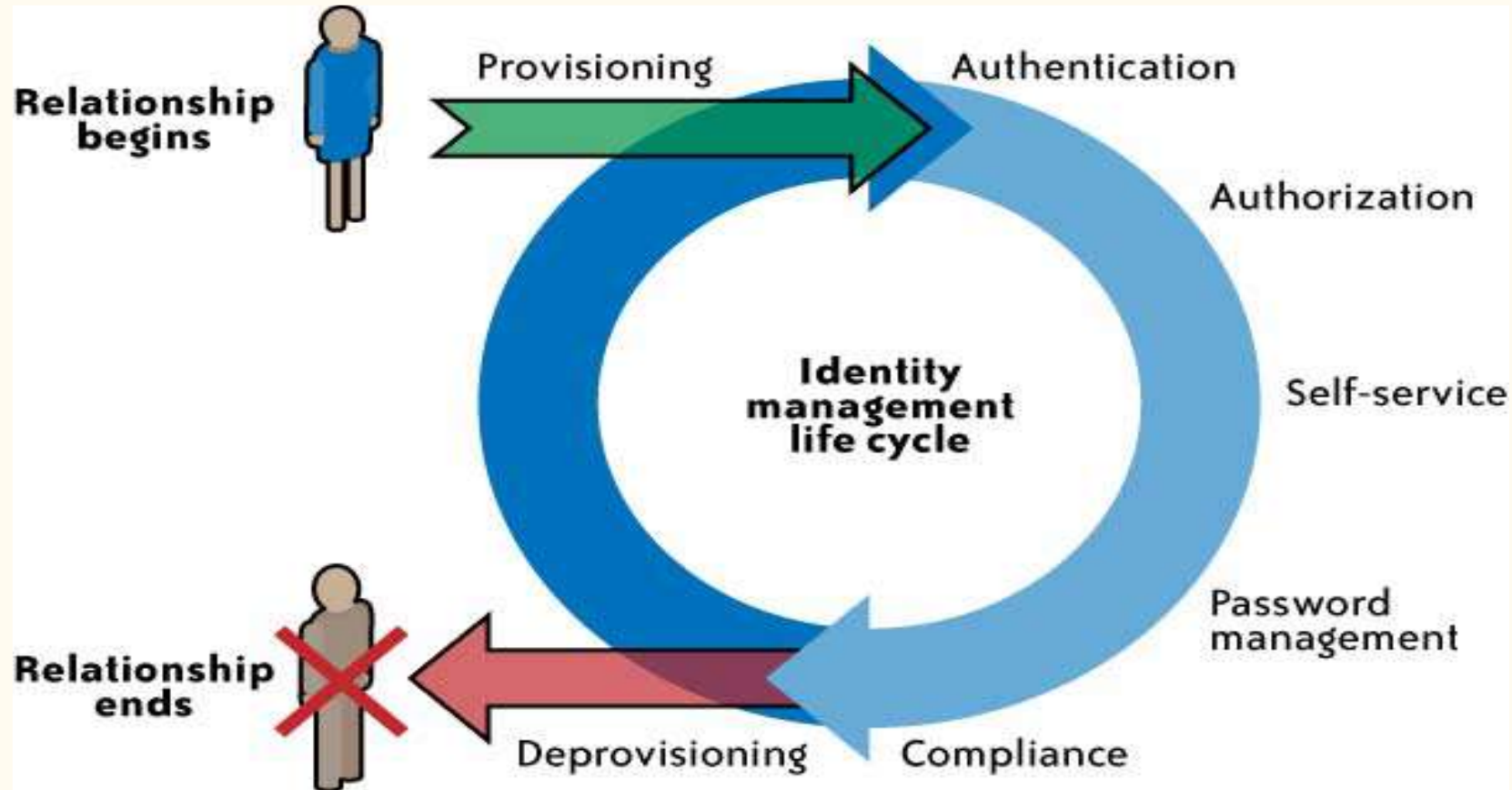
5. Identity Federation Management:

- **Federation** involves managing trust relationships between distinct organizations, enabling secure exchange of user and resource information.
- Supports **Single Sign-On (SSO)** to cloud services, allowing seamless access to external systems without re-authentication.

6. Centralization of Authentication (authN) and Authorization (authZ):

- A **centralized authentication and authorization infrastructure** removes the need for application developers to build custom security features.
- Promotes a decoupled architecture, where applications remain agnostic to authentication methods, enhancing flexibility and security through externalized authN and authZ.

Identity life cycle



Getting Ready for the Cloud

1. User Management Planning:

- Organizations need to establish user management processes, including provisioning and deprovisioning user accounts.
- Enterprises with existing Identity and Access Management (IAM) infrastructure can leverage it for cloud services.

2. Cloud-Based IAM Solutions:

- Companies without an IAM system can use cloud-based solutions from vendors like Symplified, Ping Identity, and TriCipher for identity management.

3. Federation and SSO:

- Identity federation allows secure Single Sign-On (SSO), reducing the need for multiple logins and minimizing security risks.
- It enables organizations to securely share user identities with trusted cloud service providers (CSPs) without sharing credentials.

4. **IAM Strategy and Architecture:**

- Organizations should implement an IAM strategy that includes foundational elements like user management and federation capabilities.
- A consistent user experience and secure access management are achieved through federation.

5. **Identity Provider (IdP):**

- Establishing an IdP enables secure identity sharing with CSPs using existing directory services or cloud-based identity management systems.
- IdPs can integrate with directories like LDAP or Active Directory to manage authentication securely.

6. **Federation Standards:**

- SAML 2.0 is the de facto standard for identity federation across industries, supporting secure identity sharing in private and public clouds.
- Other protocols like WS Federation and Liberty Alliance also play roles, but SAML 2.0 is widely adopted.

Relevant IAM Standards and Protocols for Cloud Services

- **IAM Standards and Specifications for Organizations**
 - **Security Assertion Markup Language (SAML)**
 - **Service Provisioning Markup Language (SPML)**
 - **eXensible Access Control Markup Language (XACML)**
 - **Open Authentication (OAuth)**

Four major challenges in user and access management faced by cloud users:

1. How can I avoid duplication of identity, attributes, and credentials and provide a single sign-on user experience for my users? SAML.
2. How can I automatically provision user accounts with cloud services and automate the process of provisioning and deprovisioning? SPML.
3. How can I provision user accounts with appropriate privileges and manage entitlements for my users? XACML.
4. How can I authorize cloud service X to access my data in cloud service Y without disclosing credentials? OAuth.

Security Assertion Markup Language (SAML)

1. Overview of SAML:

- SAML is a mature and widely adopted specification for browser-based federated sign-on.
- It allows users to access cloud services within a trusted domain after authenticating through an identity service.

2. Single Sign-On (SSO) Capabilities:

- SAML enables Single Sign-On (SSO), allowing users to bypass individual cloud service sign-ons.
- Once authenticated, users can freely access cloud services without needing separate credentials.

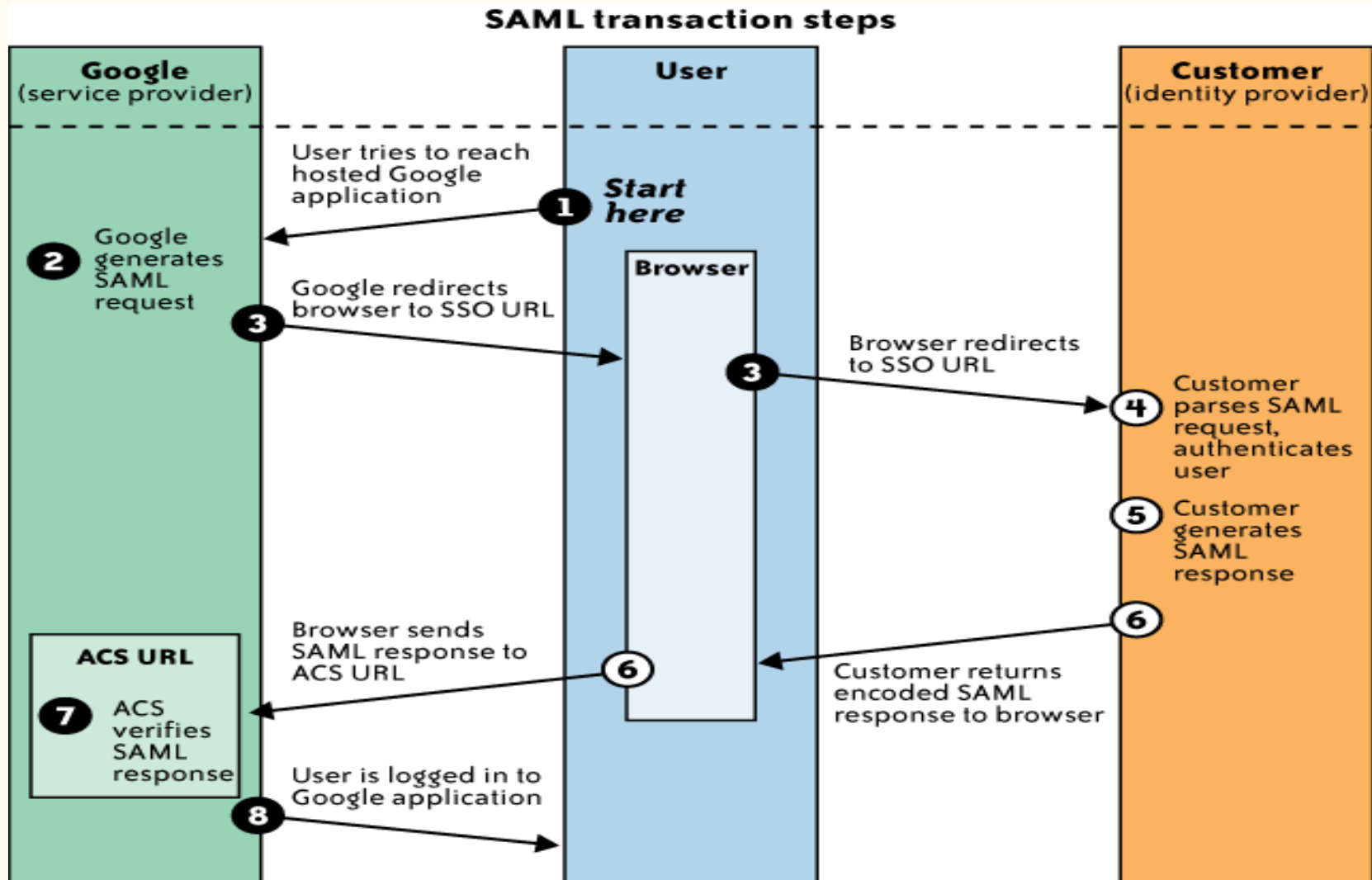
3. Delegated Authentication:

- SAML supports delegation, where the cloud service provider (CSP) delegates authentication to the customer organization's Identity Provider (IdP).
- This allows the organization to apply its own authentication policies, including risk-based and multifactor authentication.

4. Strong Authentication and Security:

- SAML facilitates strong authentication techniques, such as dual-factor authentication, reducing the risk of phishing attacks.
- Strong authentication protects user credentials from man-in-the-middle attacks, such as trojans or botnets compromising computers or browsers.

SSO transaction steps using SAML



1. User Initiates Access:

- The user attempts to access a hosted Google application (e.g., Gmail, Start Pages, etc.).

2. Google Generates SAML Request:

- Google creates a SAML authentication request, encodes it, and embeds it into the URL for the organization's Identity Provider (IdP).
- A **Relay State parameter**, containing the encoded URL of the Google application the user is trying to reach, is also included in the SSO URL.

3. Google Redirects User:

- Google redirects the user's browser to the organization's IdP service with the encoded SAML authentication request.

4. IdP Decodes SAML Request:

- The IdP decodes the SAML request, extracts the URLs for Google's Assertion Consumer Service (ACS) and the user's destination URL (Relay State parameter), and authenticates the user.
- The IdP can authenticate the user through login credentials or session cookies.

5. IdP Generates SAML Response:

- After successful authentication, the IdP generates a SAML response containing the authenticated user's username, digitally signed using the organization's public/private DSA/RSA keys.

6. IdP Returns SAML Response:

- The IdP encodes the SAML response and the Relay State parameter, then returns them to the user's browser.
- The browser forwards this information to Google's ACS, either through a form submission (automatically via JavaScript or manually).

7. Google Verifies SAML Response:

- Google's ACS verifies the SAML response using the organization's IdP public key.
- If verified, Google's ACS redirects the user to the final destination URL.

8. User is Logged into Google Apps:

- The user is now redirected to the destination URL and successfully logged into Google Apps.

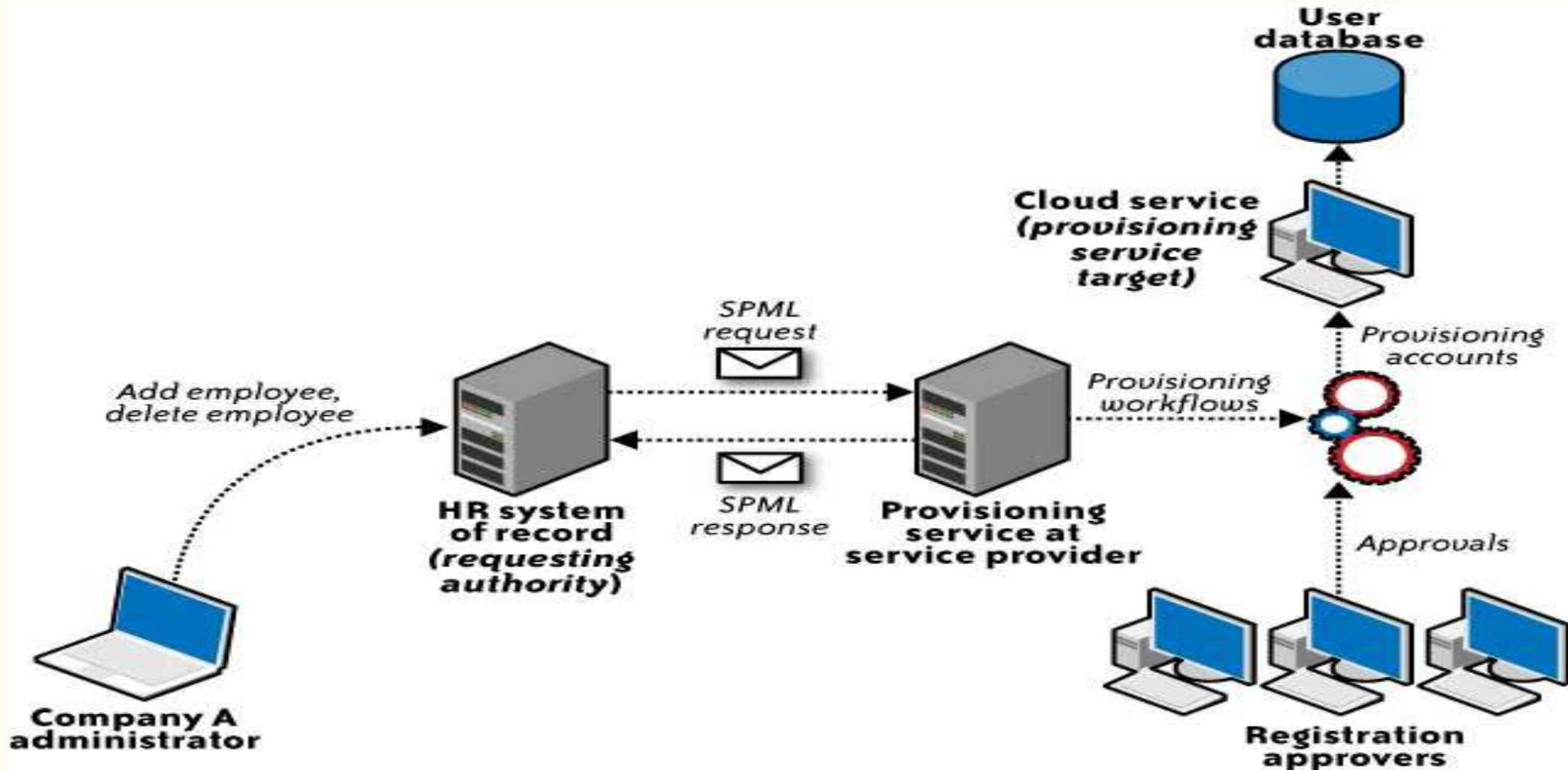
SPML Service Provisioning Markup Language

- **SPML Overview:**
 - XML-based framework developed by OASIS.
 - Aims to standardize the exchange of user, resource, and service provisioning information between different systems and organizations.
 - Supports integration and interoperability between various cloud service providers (CSPs).
- **Purpose:**
 - Automates the provisioning, updating, and de-provisioning of user identities for cloud services.
 - Essential for organizations managing large-scale or cross-organizational environments to streamline user access management.
 - Reduces manual intervention in managing user accounts and resource provisioning.

- **Process:**

- A Cloud Service Provider (CSP) extracts user attributes from a Security Assertion Markup Language (SAML) token.
- Based on this information, an SPML message is generated in real time.
- The SPML message is sent to a provisioning service that handles account creation or updates in the cloud service.
- The provisioning service adds the user identity to the cloud user database, granting access rights automatically.

SPML USE CASE



- **HR System of Record:**
 - Acts as the **requesting authority**.
 - It is an SPML web services client that initiates the provisioning request for a user account (e.g., a new employee or an update to an existing user's account).

- **SPML Provisioning Request:**

- The HR system generates an SPML request, which contains the necessary user data and provisioning details (e.g., roles, permissions).
- The request is sent via SPML to the **SPML provisioning service provider** hosted at the cloud service provider.

- **Cloud Service Provider (CSP):**
 - The cloud service provider acts as the **SPML provisioning service provider**.
 - Upon receiving the SPML request, the CSP processes the request and provisions the necessary user accounts on its cloud services.
- **Provisioning Service Target:**
 - The CSP interacts with its internal provisioning system, the **provisioning service target**, to create or modify user accounts as specified in the SPML request.
 - Once the user account is provisioned, it is recorded in the CSP's **cloud user database**, making the new user eligible for accessing cloud services.

XAML-eXtensible Access Control Markup Language

- **XML-based Access Control Language:**
 - XACML defines an XML schema to express general-purpose access control policies.
 - The policies can be applied to protect any kind of resource, whether it's data, applications, or services.
- **Policy Language and Processing Environment:**
 - XACML provides a **policy language** for writing access control rules, which govern the access rights of users based on roles, attributes, or other criteria.
 - It also includes a **processing model**, which specifies how policies should be managed and how access decisions should be reached.
- **Request/Response Protocol:**
 - XACML specifies a protocol for communication between the **Policy Enforcement Point (PEP)**, which requests access control decisions, and the **Policy Decision Point (PDP)**, which evaluates the policies and responds with an access decision.
 - Both the request for access and the response (allow/deny) are communicated using XML.

eXensible Access Control Markup Language (XACML)

- XACML is an OASIS-ratified, general-purpose, XML-based access control language for policy management and access decisions.
- It provides an XML schema for a general policy language which is used to protect any kind of resource and make access decisions over these resources.
- The XACML standard not only gives the model of the policy language, but also proposes a processing environment model to manage the policies and to conclude the access decisions.

XACML

XACML-eXtensible Access Control Markup Language

- The XACML context also specifies the **request/response protocol** that the application environment can use to **communicate with the decision point**.
- The response to an access request is also specified using XML.
- In a centrally **managed IAM architecture**, application-specific authorization models (silos) make it difficult to state the **access rights of individual users** across all applications.
- Hence, **the goal of XACML is to provide a standardized language, a method of access control, and policy enforcement across all applications that implement a common authorization standard.**
- These authorization decisions are based on various authorization policies and rules centered on the user role and job function. In short, XACML allows for unified authorization policies.

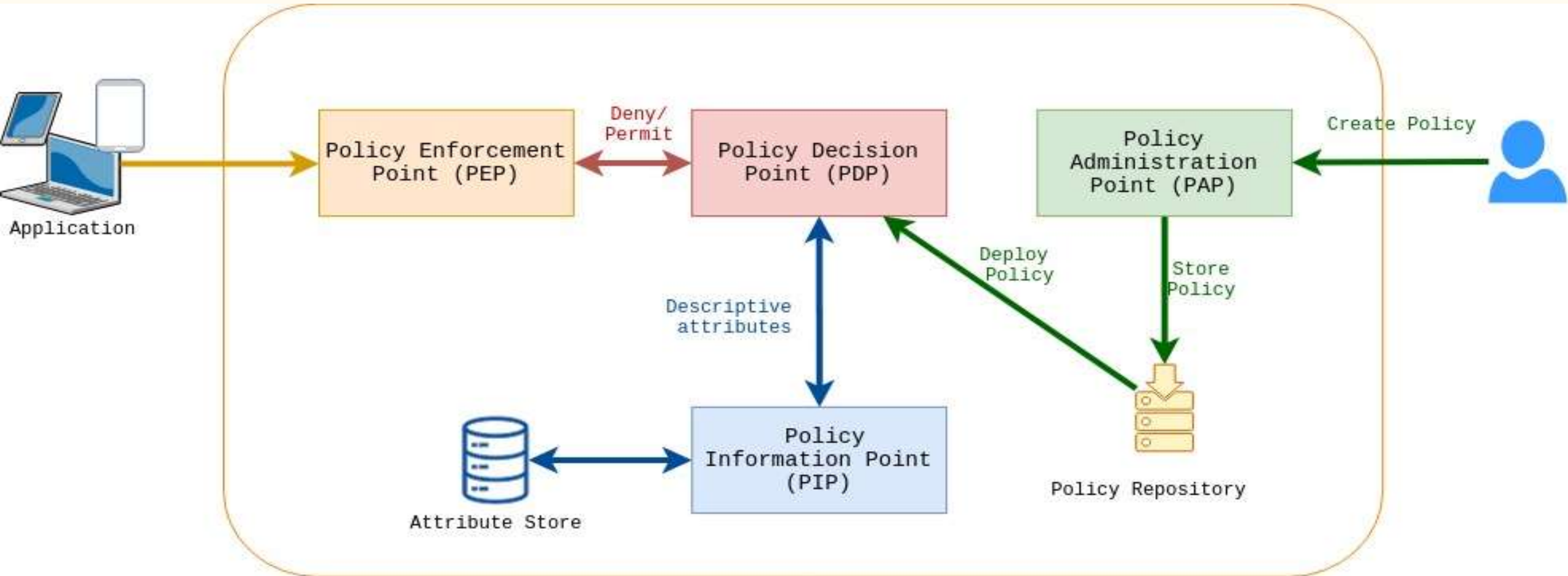
Terminology

- **PAP** – Policy Administration Point Point which manages access authorization policies
- **PDP** – Policy Decision Point Point which evaluates access requests against authorization policies before issuing access decisions
- **PEP** – Policy Enforcement Point Point which intercepts a user's access request to a resource, makes a decision request to the PDP to obtain the access decision
- **PIP** – Policy Information Point, the system entity that acts as a source of attribute values (i.e. a resource, subject, environment)
- **PRP** – Policy Retrieval Point Point where the XACML access authorization policies are stored, typically a database or the filesystem.

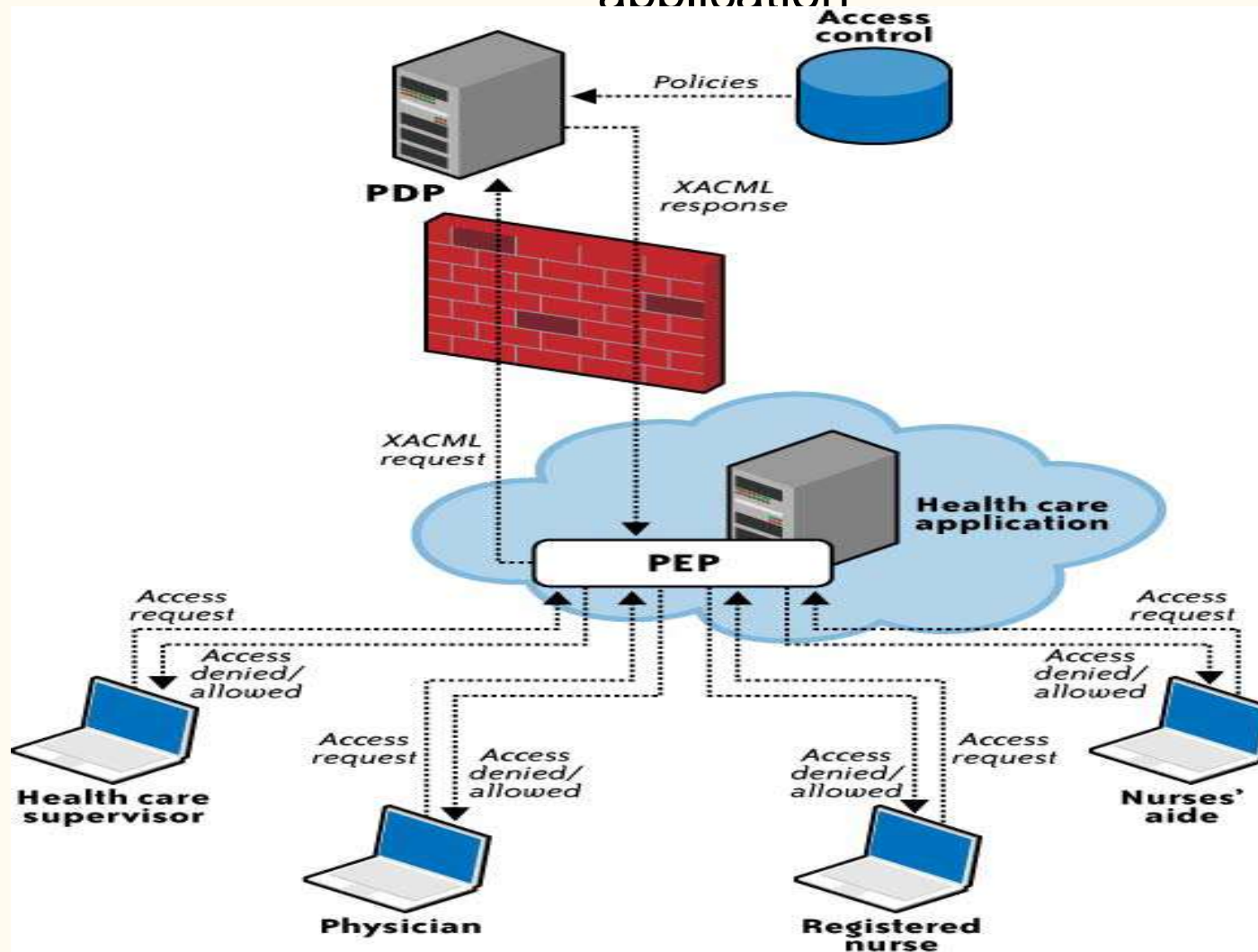
STEPS-

1. The PEP reads a request that was sent by a user and rejects it there.
2. The PEP alters the request such that it is now an XACML request.
3. The request is communicated to the PDP by the PEP.
4. The PDP compares the permission request to the access rules it has defined and decides whether or not the request should be granted.
5. These guidelines are administered by the PAP. Attribute values stored in Policy Information Points may also be accessed if it becomes required.
6. In the end, a judgment is taken at the PDP, and it either returns the value "Permit," "Deny," "NotApplicable," or "Indeterminate" to the PEP in the form of a boolean.

XACML Architecture Diagram



Illustrates the interaction among various health care participants with unique roles (authorization privileges) accessing sensitive patient records stored in a health care application



The figure illustrates the following steps involved in the XACML process:

1. The health care application manages various hospital associates (the physician, registered nurse, nurses' aide, and health care supervisor) accessing various elements of the patient record. This application relies on the policy enforcement point (PEP) and forwards the request to the PEP.
2. The PEP is actually the interface of the application environment. It receives the access requests and evaluates them with the help of the policy decision point (PDP). It then permits or denies access to the resource (the health care record).

3. The PEP then sends the request to the PDP. The PDP is the main decision point for access requests. It collects all the necessary information from available information sources and concludes with a decision on what access to grant. The PDP should be located in a trusted network with strong access control policies, e.g., in a corporate trusted network protected by a corporate firewall.
4. After evaluation, the PDP sends the XACML response to the PEP.
5. The PEP fulfills the obligations by enforcing the PDP's authorization decision.

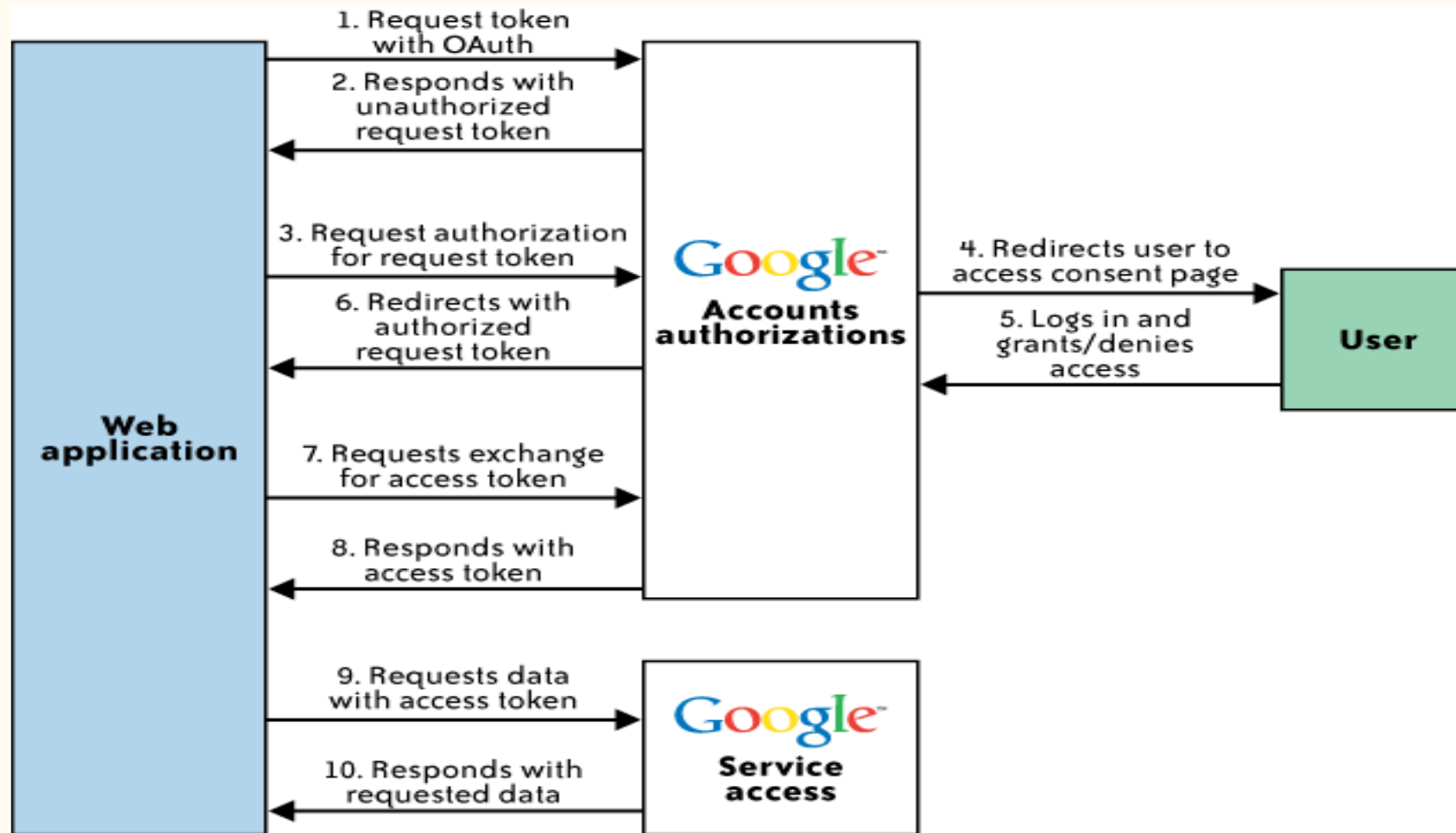
Open Authentication (OAuth)

- OAuth is an emerging authentication standard that allows consumers to share their private resources (e.g., photos, videos, contact lists, bank accounts) stored on one CSP with another CSP without having to disclose the authentication information (e.g., username and password).
- OAuth is an open protocol and it was created with the goal of enabling authorization via a secure application programming interface (API)—a simple and standard method for desktop, mobile, and web applications.

- OAuth is a technological standard that allows you to authorize one app or service to sign in to another without divulging private information, such as passwords.
- If you've ever received a message such as, "Sign in with Facebook?" or "Allow this application to access your account?" you've seen OAuth in action.
- OAuth stands for Open Authorization—not authentication, as it's sometimes assumed to be.
- Authentication is a process that verifies your identity. OAuth does involve your identity, but its purpose is to grant permission to seamlessly connect to you with different apps and services without requiring you to create a new account.
- OAuth provides that simplicity of experience by giving you the option to authorize two apps to share some of your data without revealing your credentials. It strikes a balance between convenience and security.

- For application developers, OAuth is a method for publishing and interacting with protected data.
- For CSPs, OAuth provides a way for users to access their data hosted by another provider while protecting their account credentials.
- Google released a hybrid version of an OpenID and OAuth protocol that combines the authorization and authentication flow in fewer steps to enhance usability.
- Google's GData API recently announced support for OAuth.

Illustrates the sequence of interactions between customer or partner web application, Google services, and end user:



1. Customer web application contacts the Google Authorization service, asking for a request token for one or more Google service.
2. Google verifies that the web application is registered and responds with an unauthorized request token.
3. The web application directs the end user to a Google authorization page, referencing the request token.
4. On the Google authorization page, the user is prompted to log into his account (for verification) and then either grant or deny limited access to his Google service data by the web application.
5. The user decides whether to grant or deny access to the web application. If the user denies access, he is directed to a Google page and not back to the web application.
6. If the user grants access, the Authorization service redirects him back to a page designated with the web application that was registered with Google. The redirect includes the now-authorized request token.

7. The web application sends a request to the Google Authorization service to exchange the authorized request token for an access token.
8. Google verifies the request and returns a valid access token.
9. The web application sends a request to the Google service in question. The request is signed and includes the access token.
10. If the Google service recognizes the token, it supplies the requested data.

IAM Standards, Protocols, and Specifications for Consumers

- **OpenID**
- **Information cards**
- **Open Authentication (OATH)**
- **Open Authentication API (OpenAuth)**

OpenID

- OpenID is an open, decentralized standard for user authentication and access control, allowing users to log on to many services with the same digital identity—i.e., a single sign-on user experience with services supporting OpenID.
- As such, it replaces the common logon process that uses a logon username and password, by allowing a user to log on once and gain access to the resources of multiple software systems

- OpenID is primarily targeted for consumer services offered by Internet companies including Google, eBay, Yahoo!, Microsoft, AOL, BBC, PayPal, and so on.
- OpenID adoption for enterprise use (e.g., non-consumer use) is almost nonexistent due to trust issues; some researchers have revealed that OpenID could accelerate phishing attacks that can result in compromising user credentials.

Information cards

What is an Information Card?

- An **Information Card (InfoCard)** is a **digital identity representation** that works like an electronic “ID card” stored on your computer or device.
- It was introduced by **Microsoft’s Identity Metasystem (CardSpace)** around 2005–2011.
- Instead of typing usernames and passwords, users select an InfoCard to authenticate to a service.

How It Works

- **User selects an InfoCard** (like choosing a credit card from a wallet).
- The InfoCard contains **claims** (pieces of identity information such as name, email, role).
- These claims are issued by an **Identity Provider (IdP)** in a digitally signed token (e.g., SAML).
- The InfoCard is presented to the **Relying Party (RP)** (the website or service).
- The RP verifies the token and grants access.

Types of Information Cards

- **Self-issued Cards**

- Created and managed by the user.
- Typically hold personal info (name, email).

- **Managed Cards**

- Issued by an organization or IdP.
- Carry stronger security guarantees and trusted attributes.

Benefits

- **User-friendly** → Selecting a card is easier than remembering multiple passwords.
- **Stronger Security** → Reduces password reuse and phishing.
- **Claims-based Identity** → Only necessary identity info is shared (privacy-friendly).

Limitations

- Required Microsoft **CardSpace** or compatible software.
- Never became widely adopted → replaced by modern standards like **OpenID Connect (OIDC)** and **OAuth2**.

IdP (Identity Provider) Definition

An **Identity Provider (IdP)** is a trusted system that **creates, stores, and manages digital identities** and provides **authentication services** to other applications or service providers. Think of it as the “**login authority**” that vouches for your identity.

What IdP Does

Authentication – Verifies *who you are* (via username/password, MFA, biometrics, etc.).

Attribute/Claim Sharing – Passes user info (name, email, roles, group membership) to apps.

Federation – Allows single sign-on (SSO) across multiple apps and cloud services using standards like **SAML, OAuth2, OpenID Connect**.

Examples of IdPs

- **Enterprise IdPs:** Microsoft Active Directory Federation Services (ADFS), OpenLDAP, Ping Identity, Okta.
- **Consumer IdPs:** Google, Facebook, Apple ID, LinkedIn.
- **Cloud-native IdPs:** AWS IAM, Azure AD (now Entra ID), Google Cloud IAM.

IAM Practices in the Cloud

- Dynamic nature of IAM users, systems, and applications in the cloud and addresses the four key components of the IAM automation process:
- • User Management, New Users
- • User Management, User Modifications
- • Authentication Management
- • Authorization Management

User management functions in the cloud can be categorized as follows:

- Cloud identity administration
- Federation or SSO
- Authorization management
- Compliance management

Cloud Identity Administration

1. Focus on Lifecycle Management

- Provisioning and deprovisioning of user identities.
- Identity federation and Single Sign-On (SSO).
- Password/credential management and profile management.
- Administrative management of **user accounts**.

2. Federation and Identity Management Services

- Organizations unable to support federation internally should explore **cloud-based identity management services**.
- These services typically synchronize **internal directories with a multitenant directory and act as a proxy Identity Provider (IdP)**.

Federated Identity (SSO)

- Organizations planning to implement identity federation that enables SSO for users can take one of the following two paths (architectures):
 1. • **Implement an enterprise IdP within an organization perimeter.**
 2. • **Integrate with a trusted cloud-based identity management service provider.**
- Both architectures have pros and cons.

Enterprise Identity Provider (IdP) Deployment Architecture

- Cloud services delegate **authentication** to the organization's **Identity Provider (IdP)**.
- The organization **federates identities** within a trusted circle of CSP (Cloud Service Provider) domains.
- A circle of trust is established: only authorized CSP domains can delegate authentication to the IdP.
- This approach gives the organization greater control over:
 - **User identities,**
 - **Attributes,**
 - **Credentials,**
 - **Policies for authentication and authorization**

Federated Identity

Definition

- A **federated identity** is a way of **linking a person's digital identity across multiple systems, organizations, or domains** so they can use **one set of credentials** to access many services.
- It's the foundation of **Single Sign-On (SSO)** across different platforms.

How It Works

- Instead of each application having its **own username/password** for the user, a **trusted Identity Provider (IdP)** authenticates the user.
- The IdP then shares a **security token** (via SAML, OAuth2, or OpenID Connect) with the other application (the **Service Provider / Relying Party**).
- The user gains access without logging in again

Example (Everyday Use Case)

- You log in to a **university portal** with your campus credentials.
- Using the same login, you can access:
 - Online library (external provider)
 - Email (Google Workspace / Microsoft 365)
 - Research databases (Springer, IEEE, etc.)
- You don't create separate accounts for each service -----that's **federated identity** in action.

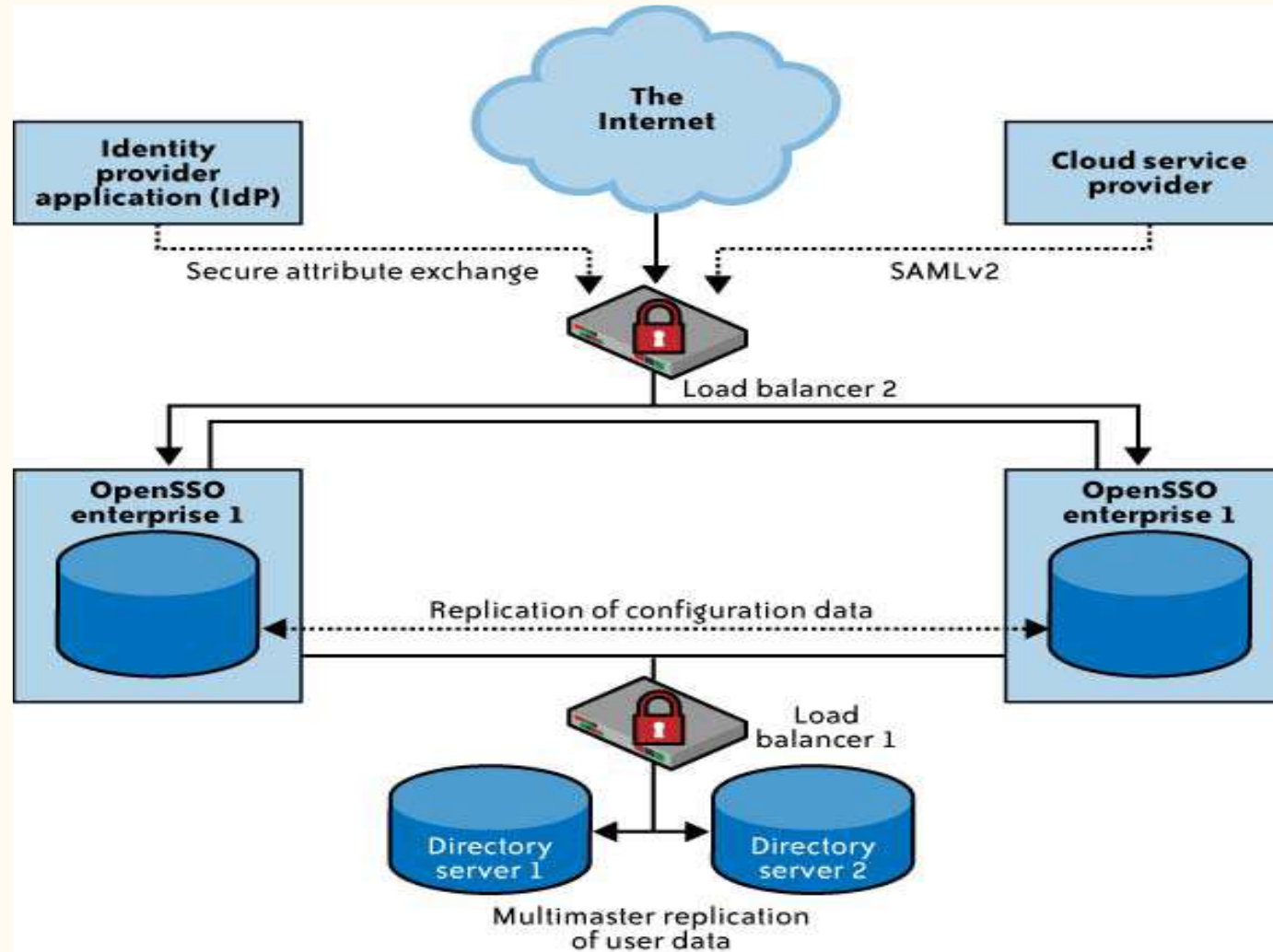
Pros

- **Leverage Existing IAM Investments**
 - Extend current IAM infrastructure (e.g., Active Directory, OpenSSO) to the cloud.
- **Consistent User Experience**
 - Enables **Single Sign-On (SSO)** across on-premises and cloud services.
- **Centralized Policy Control**
 - Organization retains full control of **authentication rules, credentials, and attributes**.
- **Stronger Security Posture**
 - Authentication does not rely solely on CSP; the IdP enforces the enterprise's security policies.
- **Reduced Identity Duplication**
 - Avoids storing multiple sets of user credentials with different CSPs.

Cons

- **Operational Complexity**
 - The organization must maintain and secure the IdP infrastructure (servers, load balancers, replication).
- **High Availability Requirements**
 - IdP becomes a **single point of failure**; must ensure redundancy and disaster recovery.
- **Integration Challenges**
 - Different CSPs may have **inconsistent support** for federation standards (e.g., SAML, OIDC).
- **Cost Overhead**
 - Additional infrastructure, licensing, and maintenance increase costs.
- **Scalability Concerns**
 - Supporting multiple CSPs and large-scale user bases requires advanced capacity planning.

Identity provider deployment architecture



Note

- Provides **federated authentication**-----one identity, multiple cloud services.
- Ensures **high availability** using load balancers and replication.
- Enterprise maintains **full control** over identities, policies, and credentials.

1. Identity Provider (IdP) and Cloud Service Provider (CSP)

- The **IdP application** (on the left) is the enterprise-controlled identity system (e.g., OpenSSO, Active Directory Federation Services).
- The **CSP** (on the right) is the cloud service (e.g., Salesforce, AWS, Office 365).
- Authentication requests from the CSP are **delegated to the IdP**.

2. Secure Attribute Exchange & Federation (SAMLv2)

- The IdP and CSP communicate using **federation protocols** like **SAMLv2**.
- User attributes (identity info, permissions, roles) are securely exchanged.
- This enables **Single Sign-On (SSO)** for users.

3. Load Balancer 2

- Between the Internet and the IdP servers, a **load balancer** ensures high availability and distributes authentication requests.
- If one server fails, requests are automatically routed to another.

4. OpenSSO Enterprise Servers

- Two **OpenSSO enterprise servers** handle authentication and authorization.
- They **replicate configuration data** between them for consistency (so policies, rules, and settings are always synchronized).

How it Works – Flow

- A user tries to access a **CSP service**.
- The CSP redirects the request to the **IdP** using SAMLv2.
- The **IdP authenticates** the user using its **directory servers**.
- If successful, the IdP sends a **token/assertion** back to the CSP.
- The CSP grants access to the cloud service — without needing a separate CSP password.

5. Load Balancer 1

- A second **load balancer** manages traffic to the backend **directory servers** (user databases).
- Ensures requests are distributed evenly and avoids bottlenecks.

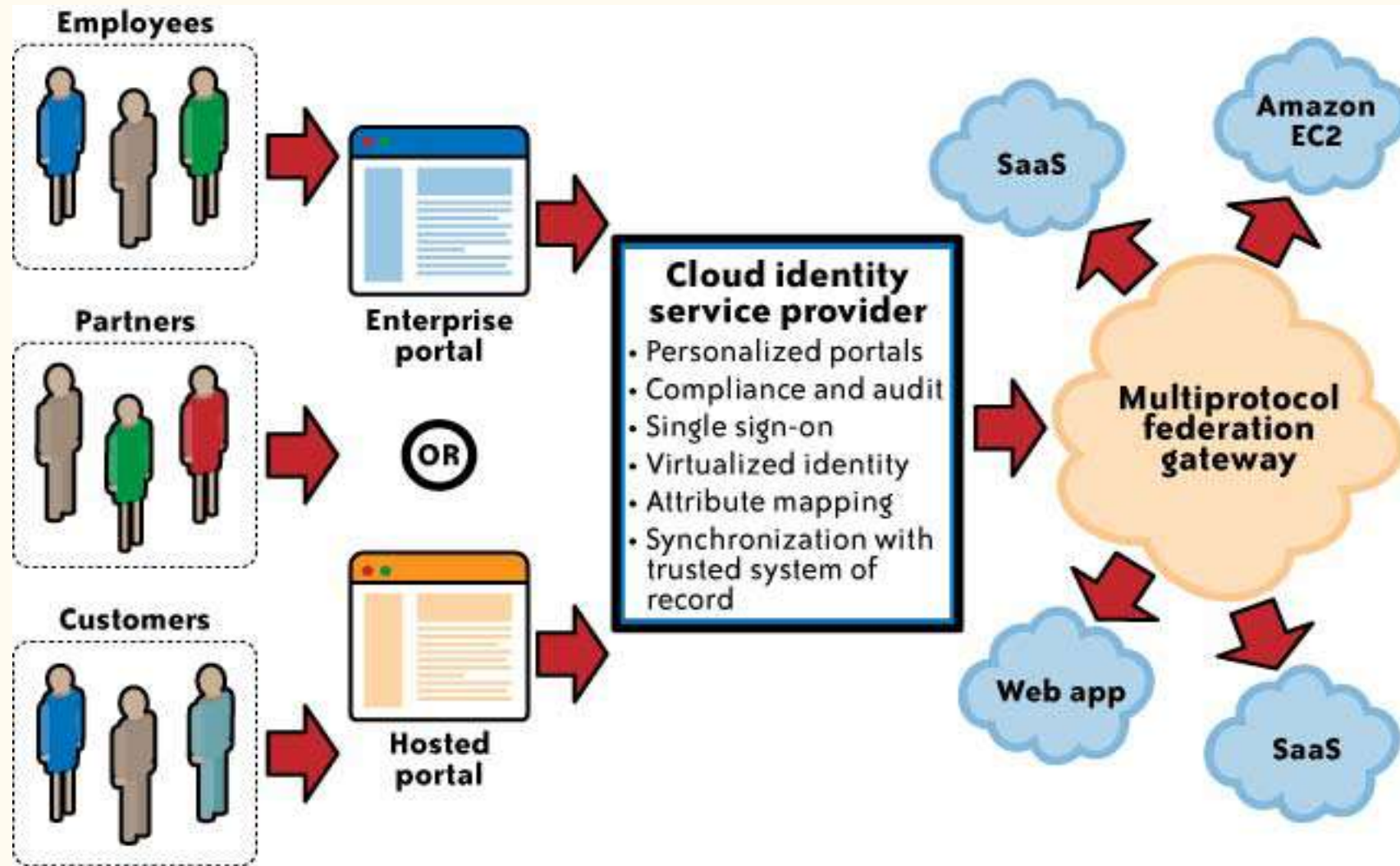
6. Directory Servers

- The **Directory Servers (1 & 2)** store **user data** (credentials, attributes, roles, policies).
- They perform **multimaster replication**, meaning updates in one directory automatically replicate to the other.
- This ensures **data redundancy** and **real-time synchronization**.

Identity Management as a Service (IDaaS) Architecture

- Cloud services can **delegate authentication** to a **cloud-based IdP (IDaaS provider)**.
- Organizations **outsource identity management** (federation, lifecycle, user processes) to third-party providers like **Ping Identity, Myonelogin, Symplified**.
- This creates a **SaaS model for identity management**.

Cloud Identity Service Provider (IDaaS) with Multiprotocol Federation Gateway model.



1. Users

- **Employees, Partners, Customers** need access to cloud services.
- They log in through:
 - **Enterprise portal** (for internal users like employees/partners), OR
 - **Hosted portal** (for external users like customers).

2. Cloud Identity Service Provider (IDaaS)

- This is the **third-party identity management service** (e.g., Ping Identity, Okta, Symplified).
It provides:
 - **Personalized portals** – Tailored login experiences for different user groups.
 - **Compliance & audit** – Logging and reporting for regulatory needs.
 - **Single Sign-On (SSO)** – One login works for multiple apps.
 - **Virtualized identity** – Centralized digital identity in the cloud.
 - **Attribute mapping** – Mapping user info (roles, department, etc.) to apps.
 - **Synchronization** – **Keeps cloud identity store in sync with enterprise's directory.**

3. Multiprotocol Federation Gateway

- Many cloud providers (CSPs) use **different federation standards** (e.g., SAML 1.1, SAML 2.0, OAuth2, OpenID Connect).
- **The gateway acts as a translator so organizations don't need to reconfigure IAM for each provider.**
- It standardizes identity exchange across multiple CSPs.

4. Cloud Services

- Once authentication is done, users can access:
 - **SaaS apps** (e.g., Salesforce, Google Workspace, Office 365)
 - **Web apps**
 - **Infrastructure services** (e.g., Amazon EC2)

How it Works (Flow)

- User (employee, partner, or customer) logs into the **enterprise or hosted portal**.
- Authentication request goes to the **Cloud Identity Service Provider (IDaaS)**.
- The IDaaS authenticates the user (using synced corporate directory + policies).
- The request passes through the **Multiprotocol Federation Gateway**, which translates federation standards if needed.
- User gains **seamless SSO access** to cloud apps (SaaS, web apps, IaaS).

Cloud Authorization Management[3]

1. Enterprise Needs for Authorization

- Medium and large organizations usually require **fine-grained authorization features** to align with their internal security policies.
- This involves assigning privileges (also called **entitlements**) to users depending on their **job roles** or responsibilities.
- A common method is **Role-Based Access Control (RBAC)**, where access rights are tied to organizational roles (e.g., HR staff, finance officers, IT administrators).

2. Current Limitations in Cloud Services

- Many **cloud service providers (CSPs)** still offer only **basic authorization features**, often too **coarse-grained** for enterprise needs.
- In practice, most CSPs support only **two main roles**:
 - **Administrator** – can manage user accounts, set limited access policies, and control basic system configurations.
 - **End User** – has access only to the services they use, with no administrative capabilities.

3. Typical Administrative Privileges

- Administrators usually have the ability to:
 - Provision and deprovision identities (create or remove user accounts).
 - Manage basic user profiles (e.g., attributes like name, email, department).
 - Configure basic policies, such as password strength rules or trusted network restrictions.

4. XACML (eXtensible Access Control Markup Language)

- XACML is an industry standard for expressing fine-grained access control policies (who can access what, under which conditions).
- It enables policy-driven authorization and supports complex organizational requirements.
- However, as of now, most cloud providers do not support XACML for user authorization, leaving enterprises with limited policy enforcement options.

IAM Support for Compliance Management[5]

1. IAM's Role in Compliance

- **Identity and Access Management (IAM)** isn't just about controlling access—it is also a **compliance enabler**.
- Many compliance frameworks (like **GDPR, HIPAA, SOX, ISO 27001**) require strong access controls and auditability.
- A well-implemented IAM system helps ensure that the **right people** have the **right access** at the **right time**—and that this can be proven during an audit.

2. Automation for Compliance

- Timely provisioning and deprovisioning of users is critical:
 - When an employee joins → provision access quickly.
 - When they leave or change roles → deprovision/adjust access immediately.
- Automating this reduces the risk of **orphaned accounts** (ex-employees still having access) or **over-privileged users**, which are both compliance and security risks.
- This also supports **privacy requirements** by ensuring that personal and sensitive data is only accessible by authorized users.

3. Focus on Identity & Attribute Management

- Compliance rules emphasize **identity attributes** (e.g., department, location, clearance level).
- Managing these attributes centrally allows organizations to enforce access control policies more effectively.
- Example: A finance role attribute should automatically restrict a user to financial data only, supporting **least privilege access**.

4. Insider Threat Protection

- IAM gives a **centralized view of user activities**.
- Automated controls can detect unusual or risky access patterns and stop insider threats before they cause damage.
- Example: If a user tries to access data outside their role, IAM can flag or block it.

5. Challenges with Cloud IAM Standards

- Common IAM standards:
 - **SAML** → federation (single sign-on, cross-domain authentication).
 - **SPML** → provisioning (automated account management).
 - **XACML** → fine-grained authorization policies.
 - Problem: Many **Cloud Service Providers (CSPs)** have **limited support** for these standards.
 - This means enterprises must carefully **assess each CSP** for IAM capability gaps and put in place **custom processes** to handle compliance requirements.
-

Cloud Service Provider IAM Practices

1. IAM as a Design Requirement for CSPs

- **CSPs (SaaS, PaaS, IaaS)** should **embed IAM features** into the service design from the start.
- Goal: Allow customers to manage **user authentication and authorization** through federation and user management standards.
- Early IAM integration helps both:
 - **Customers** – easier adoption (single sign-on, account provisioning).
 - **CSPs** – smoother billing, accounting, and resource usage tracking.
- If IAM is added later → costly retrofits.
- Best practice: **Externalize authentication** from applications (e.g., use federation standards instead of custom login modules).

IAM Requirements in the Cloud[4]

1. Provisioning

- Users (including admins) must have cloud accounts provisioned automatically.
- Service-to-service integration (e.g., private cloud ↔ public cloud) must also support provisioning.

2. Single Sign-On (SSO)

- Should be based on **federation standards** (e.g., **SAML**, **OpenID Connect**).
- Enables seamless user experience across multiple apps/clouds.

3. Compliance & Access Control

- IAM must support **regulatory compliance** (SOX, PCI, HIPAA).
- **RBAC (Role-Based Access Control)** → grants minimum required privileges (least privilege model).
- **Claims-based authentication** → only **entitlements** (not full user identity) are passed.
 - Example: Instead of sending “Alice Smith, HR Manager,” the system sends “Role: HR, Permission: View Payroll.”
 - This supports **privacy** and **fine-grained authorization**.

4. User Activity Monitoring & Auditing

- Logging, monitoring, and reporting of user activity are mandatory.
- Helps meet internal security policies and external regulatory requirements.
- Example: Audit trails for HIPAA or PCI compliance.

SaaS Identity and Access Management (IAM) Responsibilities

[1] Key Security Concerns in SaaS

- **Security Management** ----A top concern for IT and business leaders when adopting SaaS.
- **Operational Security vs. Access Control**---While most SaaS vendors ensure operational security, organizations must address **access control** to align with internal policies and compliance.
- **Shadow IT Risk**---Business units may bypass IT and directly purchase SaaS, causing loss of visibility and control.
- **Access Management Requirement**----IT must ensure **right users** → **right access** → **right time**.

[2] IAM Challenges for Organizations

- **Provisioning & Life Cycle Extension**
 - Can the organization extend its existing IAM practices to SaaS services?
- **SaaS Provider IAM Capabilities**
 - Does the SaaS provider support automated provisioning and life cycle management, or is a **custom solution** needed?

Customer Responsibilities in SaaS IAM

- **User Provisioning**

- Methods vary by provider (manual uploads, bulk spreadsheets, XML, just-in-time provisioning via SPML).

- **Profile Management**

- Creation of roles (e.g., *user*, *manager*) to manage entitlements. Often basic and less flexible.

- **Evaluating IAM Features**

- Check for support of **SSO** and federation (SAML 1.1 vs. 2.0).
- Example: Salesforce supports SAML 1.1, Google Apps supports SAML 2.0.

- **Investigation & Monitoring**

- Logs and audit trails may be internal to the provider.
- Customers must negotiate **log access** for compliance (e.g., PCI DSS requires forensic investigation support).

- **Compliance Management**
 - Customers are responsible for ensuring compliance with **SOX, GLBA, HIPAA, PCI DSS**.
 - Assess SaaS provider's **access control, logging, auditing** capabilities.
- SOX (Sarbanes–Oxley Act)= Financial reporting accuracy and integrity.
- GLBA (Gramm–Leach–Bliley Act)= Protecting consumers' financial information.
- HIPAA (Health Insurance Portability and Accountability Act)
- PCI DSS (Payment Card Industry Data Security Standard)

CSP Responsibilities in IAM (SaaS Context)

1. Authentication Services

- If **federated authentication** is not supported, CSPs authenticate users directly.
- Typically done via **web forms over HTTPS** with a **user ID + password**.
- CSPs may enhance security by:
 - Pre-registering **trusted IP addresses/ranges** (home, office, etc.).
 - Blocking access from suspicious/untrusted networks.
 - Preventing credential theft from **keyloggers** or compromised devices.
- **Key responsibility:** Maintain a **highly available authentication service**, since it is the **entry point** for SaaS usage.

2. Account Management Policies

- CSPs must define and communicate policies such as:
 - **Account lockout** rules (e.g., after multiple failed login attempts).
 - **Account provisioning methods** (manual, automated, delegated).
 - **Privileged account management** roles and restrictions.
- Ensures customers understand **how accounts are created, suspended, or terminated**.

3. Federation Support

- CSPs should support **identity federation** to enable **Single Sign-On (SSO)**.
- Must clearly publish:
 - **Federation standards supported** (e.g., **SAML 1.1, SAML 2.0**).
 - **Implementation examples** for customer integration.
 - **Technical details** for APIs (REST, SOAP) used in federation.
- Enables enterprises to integrate SaaS authentication with **corporate IAM systems**.

PaaS IAM (Identity and Access Management)

- **Key Considerations for Organizations**
- Extending enterprise **IAM practices** to PaaS providers is **more limited** than in SaaS.
- PaaS typically offers **fewer standardized IAM options**, requiring adaptation to provider-specific methods.

IaaS IAM (Identity and Access Management)

- **IaaS = Compute & Storage as a Service** (e.g., AWS EC2).
- Providers do **not** have visibility into the applications hosted on their infrastructure.
- IAM support is **basic and infrastructure-focused**, not application-level.
- Access typically via **SSH** (for admins) and **API keys/certificates** (for customers).

IaaS IAM Responsibilities & Challenges

1. User Provisioning

- Provisioning developers/administrators on **dynamic IaaS systems**.
- Large-scale workloads → provisioning must be **automated at image creation**.
- Best practice: integrate with **corporate directories (LDAP/Active Directory)**.
- Challenge: **cloud virtual networks** and **security policies** may block directory-based authentication.

2. Privileged User Management

- Admins often use **SSH private keys**.
- Keys must be **secured, rotated, and revoked** when admins leave.
- Risk: **lost or compromised keys** → **full system access**.