Sécurité dans les environnements infonuagiques

Module 2 : Gestion des identitiés et des accès (Partie 2)

Armstrong Foundjem

Polytechnique Montréal

Automne 2023

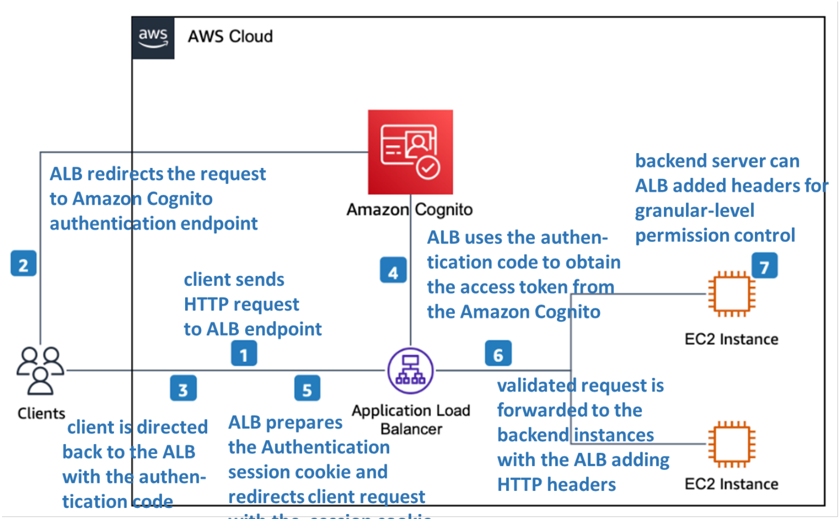
Plan

1. Identity and Access Management Security
2. Compliance

1 IAM security

2 IAM Compliance

Authentication



ALB redirects the request to Amazon Cognito authentication endpoint

client sends HTTP request to ALB endpoint

client is directed back to the ALB with the authentication code

ALB uses the authentication code to obtain the access token from the Amazon Cognito

ALB prepares the Authentication session cookie and redirects client request with the session cookie

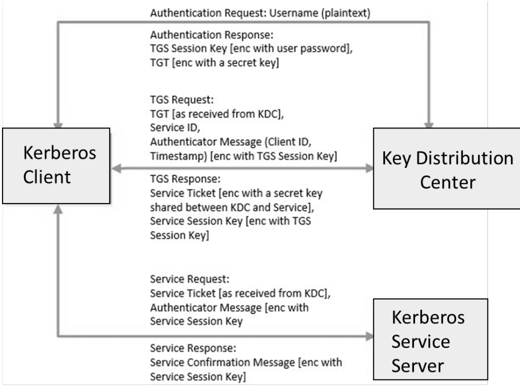
validated request is forwarded to the backend instances with the ALB adding HTTP headers

backend server can ALB added headers for granular-level permission control

Authentication

Kerberos protocol

- Client logins with username/pass
- KDC verifies credentials and sends a TGS session key and TGT.
- Client sends a TGS request using TGT received and a message encrypted with the TGS session key
- KDC decrypts the message with the TGS session key and verifies if it matches the client ID/timestamp and returns a service ticket and a service session key (SSK)
- Client authenticates and Service Server confirms it using the SSK
- **Disadvantages:** vulnerable to
  - pass-the-ticket via Windows LSASS memory extraction
  - pass-the-hash via Windows NT LAN Manager (NTLM) auth. protocol (mimikatz)



Kerberos Client

Key Distribution Center

Kerberos Service Server

Authentication Request: Username (plaintext)

Authentication Response:
TGS Session Key [enc with user password],
TGT [enc with a secret key]

TGS Request:
TGT [as received from KDC],
Service ID,
Authenticator Message (Client ID, Timestamp) [enc with TGS Session Key]

TGS Response:
Service Ticket [enc with a secret key shared between KDC and Service],
Service Session Key [enc with TGS Session Key]

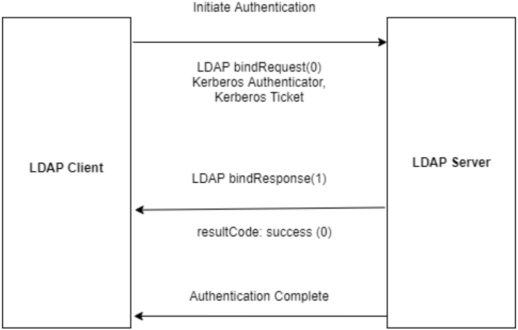
Service Request:
Service Ticket [as received from KDC],
Authenticator Message [enc with Service Session Key

Service Response:
Service Confirmation Message [enc with Service Session Key]

Authentication

LDAP protocol

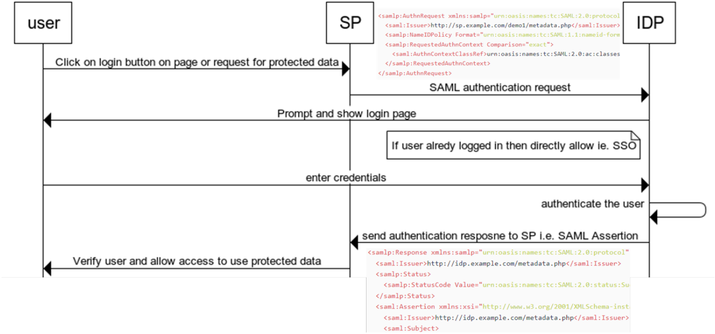
- refers to Lightweight Directory Access Protocol
- centrally manages authentication and access on Directories (e.g., Microsoft Active Directory)
- Client sends a bind request through Kerberos authentication challenge
- Server returns a bind response containing a success code allowing to Client to access Directories
- **Disadvantages:** vulnerable to
  - injection via query manipulation (e.g., special characters)
  - remote code execution via client redirection to a malicious LDAP server



Authentication

SAML2 Protocol

- provides XML-based authentication between client, identity provider (IDP), and service provider (SP)



user
SP
IDP
Click on login button on page or request for protected data
SAML authentication request
Prompt and show login page
If user alredy logged in then directly allow ie. SSO
enter credentials
authenticate the user
send authentication resposne to SP i.e. SAML Assertion
Verify user and allow access to use protected data

```xml
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol
  <saml:Issuer>http://sp.example.com/demo1/metadata.php</saml:Issuer>
  <samlp:NameIDPolicy Format="urn:oasis:names:tc:SAML:1.1:nameid-form
  <samlp:RequestedAuthnContext Comparison="exact">
    <saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes
  </samlp:RequestedAuthnContext>
</samlp:AuthnRequest>
```

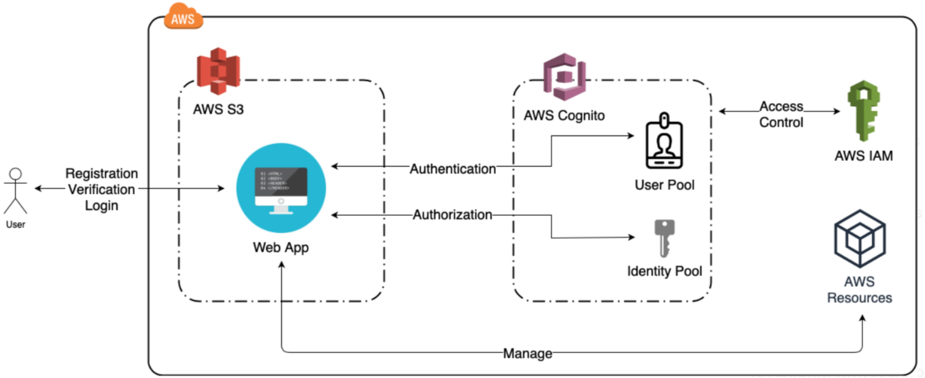
```xml
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  <saml:Issuer>http://idp.example.com/metadata.php</saml:Issuer>
  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Su
  </samlp:Status>
  <saml:Assertion xmlns:xsi="http://www.w3.org/2001/XMLSchema-inst
    <saml:Issuer>http://idp.example.com/metadata.php</saml:Issuer>
    <saml:Subject>
```

- **Disadvantages:** vulnerable to
  - XML injection via response modification
  - Golden SAML attack: use ADFSDump tool to extract private keys and bypass Windows AD FS

Sondage ☺: `https://app.wooclap.com/INAOAA`.

wooclap

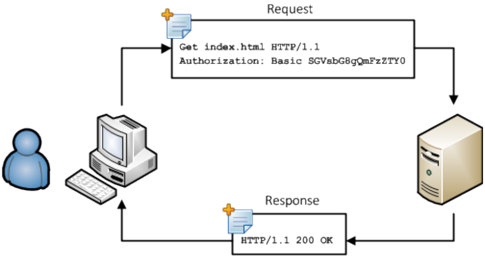
Authorization



Authorization

Basic Auth

- client sends a request with user name/password as encoded base64 (**unencrypted**) over HTTPS
- server decodes the user name/password and verify if it matches those in the database and returns a HTTP response
- Basic authorization header format: **Authorization: Basic <base64_encode_user_pass>**
- **Disadvantages:**
    - Easy to break
    - Very insecure even when used over HTTPS



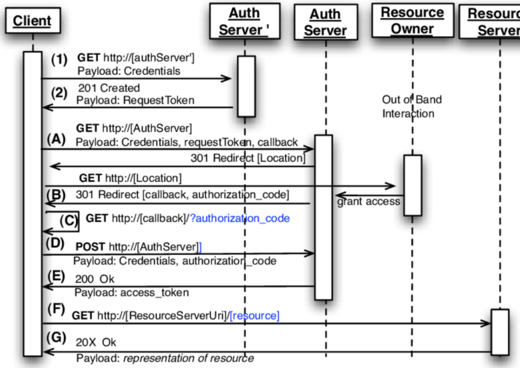
```
Get index.html HTTP/1.1
Authorization: Basic SGVsbG8gQmFzZTY0
```

```
HTTP/1.1 200 OK
```

Authorization

OAuth2

- User requests authorization (authorization request) from the Authorization server
- Authorization server authenticates User and verifies the requested scopes
- Resource owner interacts with the Authorization server to grant access
- Authorization server redirects back to User with either an Authorization Code or Access Token
- User requests access to the resource from the Resource server using the access token
- **Disadvantages:**
  - attacker can steal OAuth Token via URI redirection


Client
Auth Server '
Auth Server
Resource Owner
Resourc
Server
(1) GET http://[authServer']
Payload: Credentials
(2) 201 Created
Payload: RequestToken
Out of Band Interaction
(A) GET http://[AuthServer]
Payload: Credentials, requestToken, callback
301 Redirect [Location]
GET http://[Location]
(B) 301 Redirect [callback, authorization_code]
grant access
(C) GET http://[callback]/?authorization_code
(D) POST http://[AuthServer]]
Payload: Credentials, authorization_code
(E) 200 Ok
Payload: access_token
(F) GET http://[ResourceServerUri]/[resource]
(G) 20X Ok
Payload: representation of resource

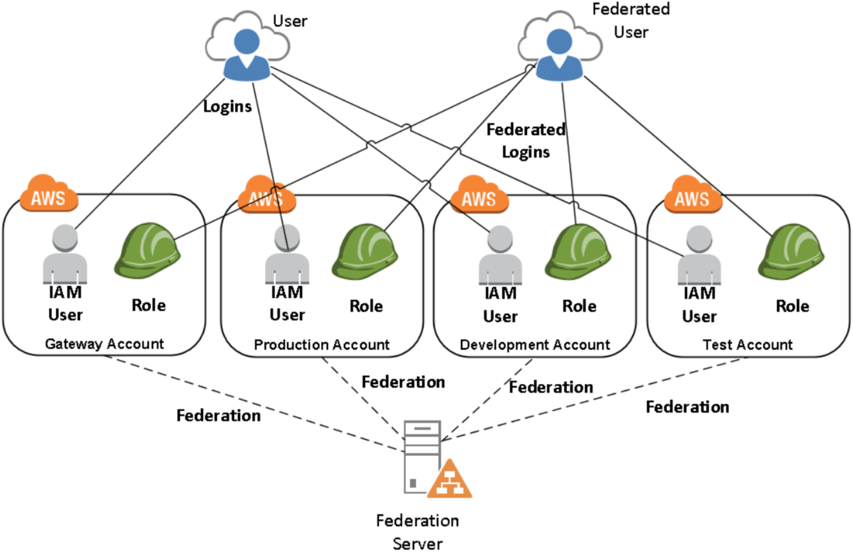
Sondage ☺: https://app.wooclap.com/NGGQAK.

wooclap

Access governance

- process of monitoring and controlling
- who within the organization
- has access to what,
- when and how
- **Components**
    - *Identity governance and administration* : create policies for users based on least privilege and RBAC, conduct access review, produce reports of authentication/ authorization activities
    - *Data access management*: identify who has access and permissions to given resources
    - *Reporting and compliance*: provide compliance reports that outline user access and permissions, adapt to data privacy laws and new regulations.

Federation

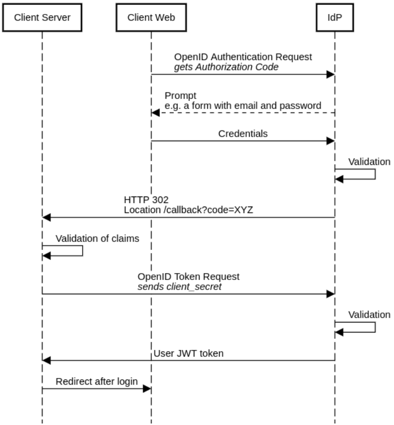

User
Federated User
Logins
Federated Logins
AWS
IAM User
Role
Gateway Account
Production Account
Development Account
Test Account
Federation
Federation
Federation
Federation
Federation Server

Federation

What are the federated identity protocol ?

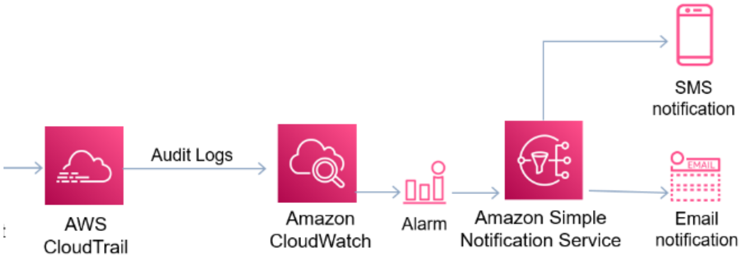
- **OAuth2 SSO, SAML2 SSO**
- **OpenID Connect SSO**
  - helps to check the identity of the End-User based on OAuth2

OpenID authorization flow



Audit and Monitoring

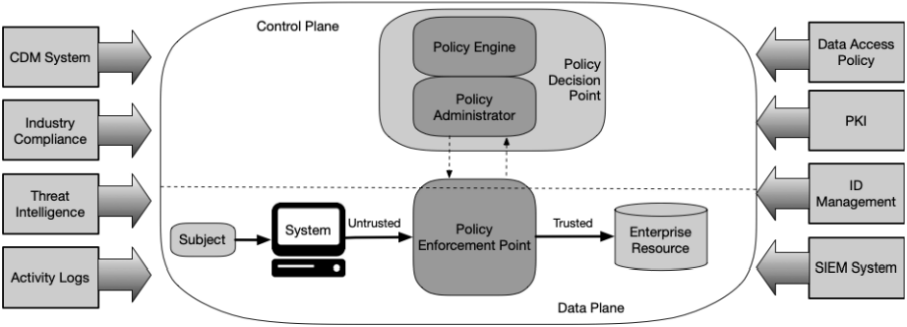
- Record IAM activities using AWS CloudTrail
- Monitor IAM activities using AWS CloudWatch
- Alert unusual behaviors using AWS Simple Notification Service



Zero-Trust Access

- Never trust users/resources/assets/network, always verify
- continuously monitor trust and update the security policies on the system
- The trust information are collected from
  - activity logs, threat intelligence
  - industry compliance, data access policies
  - IAM, system and information event management (SIEM), etc.
- The policy enforcement point (PEP) control accesses (e.g., deny, allow) based on instructions from the policy decision point (PDP)

Zero-Trust Access



- In the PDP, a policy administrator takes ultimate decision (e.g., grant, deny, revoke) based on the policy engine

1 IAM security

2 IAM Compliance

Sarbanes-Oxley (SOX)

- Standard protecting the integrity of the financial information in banking and insurance companies
- IAM must enforce separation of duties (SoD) policies
- IAM must provide a centralized system for managing user access rights and authentication.
- IAM must allow regular audits of access rights and privileges
- IAM must revoke user access after termination

Health Insurance Portability and Accountability Act (HIPAA)

- Standard protecting the privacy of health data
- IAM must support
  - least privileges
  - multifactor authentication
  - RBAC
  - regular key rotation
  - SSO

Payment Card Industry Data Security Standard (PCI)

- Standard protecting credit card information and access
- IAM must revoke user access after termination
- IAM must remove inactive users after a given period
- IAM must ensure a proper user identification management

General Data Protection Regulation (GDPR)

- European standard for data protection and privacy
- IAM must provide monitoring or analytics of activities manipulating data
- IAM must ensure proper user identification management (e.g., Identity Federation, SSO)
- IAM must allow regular audits of access rights and privileges