신뢰기반 디지털 ID 관리기술

한국과학기술정보연구원 조진용
jiny92@kisti.re.kr

Audience

대상

- 통합인증과 인증연합에 관심 있는 누구나
- 인증연합에 참여 중인 학연 기관 및 서비스제공자

수준

- 초급+α
- IT 관련 기초지식 필요

참고

- CentOS 7 기준

KAFE

Agenda

1교시 (13:30PM - 14:45PM)

- 실습환경 구성
- SAML 표준 소개
- 아이디제공자의 구축
- 실습

2교시 (15:00PM - 16:30PM)

- MFA의 이해
- 실습

KAFE

실습환경 구성

목적

¶ SAML의 개념과 동작원리 이해
¶ 아이디제공자(Identity provider)의 관리운영

할일

① 계정 생성
② Chrome용 SAML 플러그인 설치
③ notepad++ 설치

실습자료

https://edu.kafe.or.kr

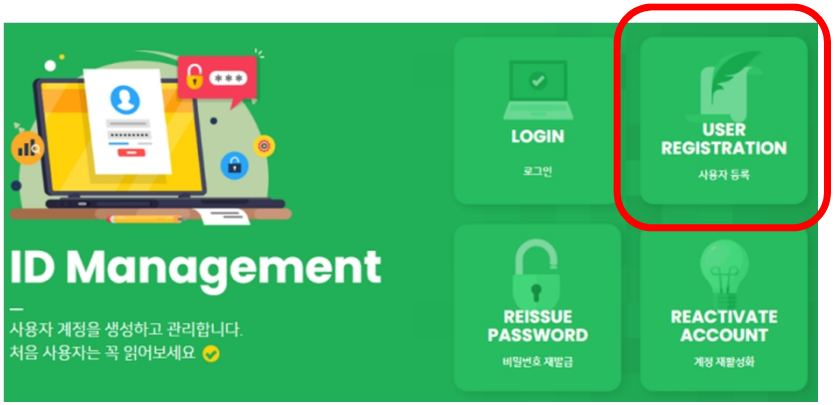
실습환경 구성 - 계정 생성

계정 생성이 필요하지 않음 (Home 계정)

¶ KAFE 참여기관(https://www.kafe.or.kr/participants_idp)

계정 생성이 필요 (Guest 계정)

① https://coreen-idp.kreonet.net 방문
② 사용자 등록 메뉴에서 계정 생성
  ＊ 본명 사용
  ＊ 연구기관이나 교육기관 구성원은 기관 이메일 주소 입력



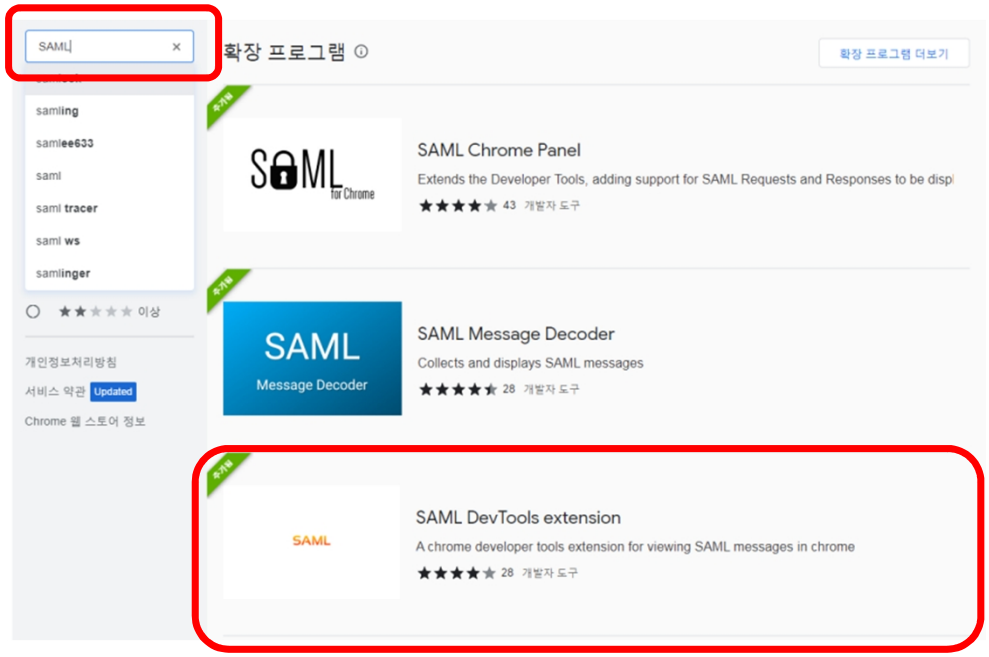
기억하세요!

¶ Guest 계정의 이름은 'COREEN Set.ID by KAFE'

실습환경 구성 - Chrome용 SAML Plugin

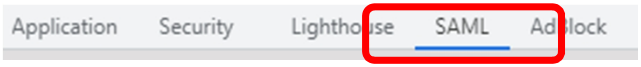
설치 방법 (Chrome 브라우저)

① https://chrome.google.com/webstore/category/extensions 또는 Google에서 'Chrome 확장프로그램 설치'로 검색
② 'SAML' 로 검색
③ 'SAML DevTools extension' 로 설치



설치 확인

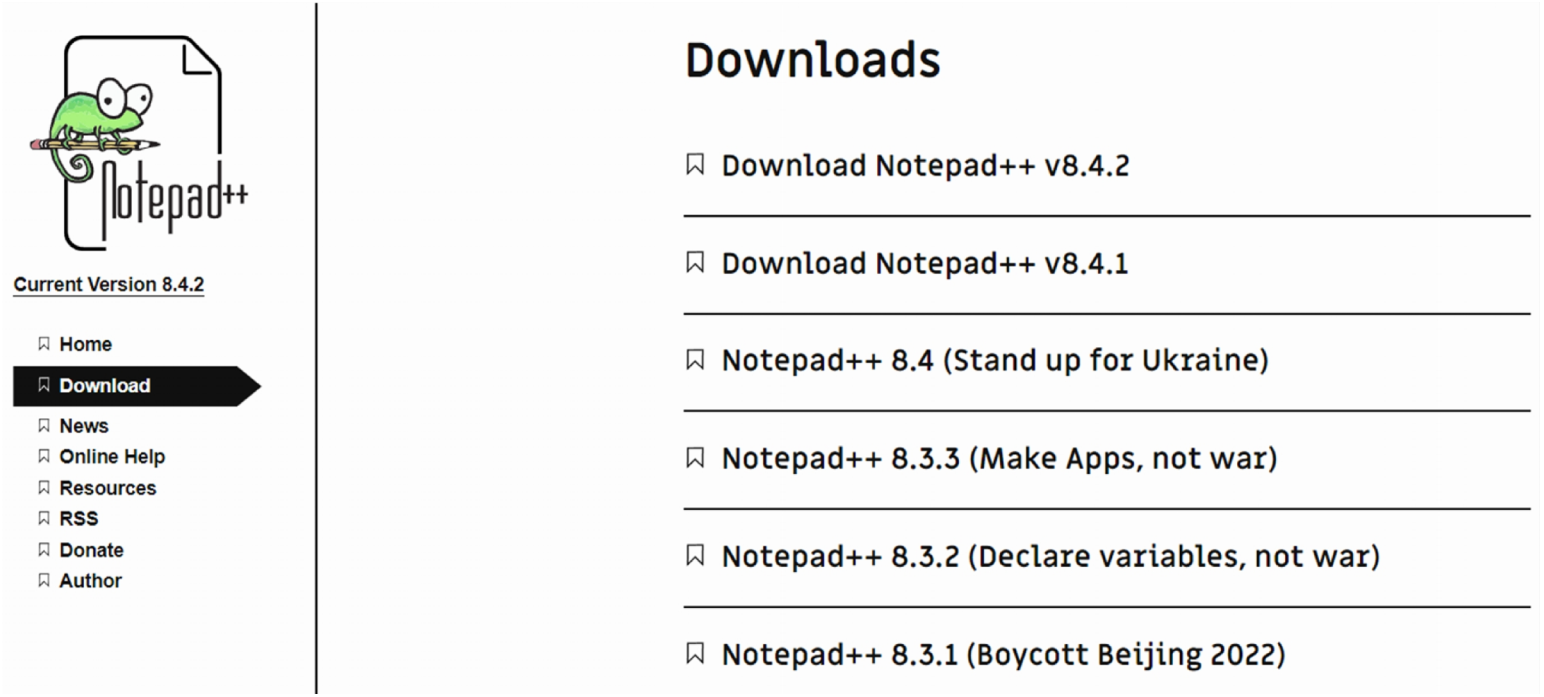
¶ Chrome 브라우저에서 F12 키 입력



실습환경 구성 - Notepad++

설치 방법

① Google에서 ‘notepad++’로 검색
② 최신버전을 다운로드 받아 설치

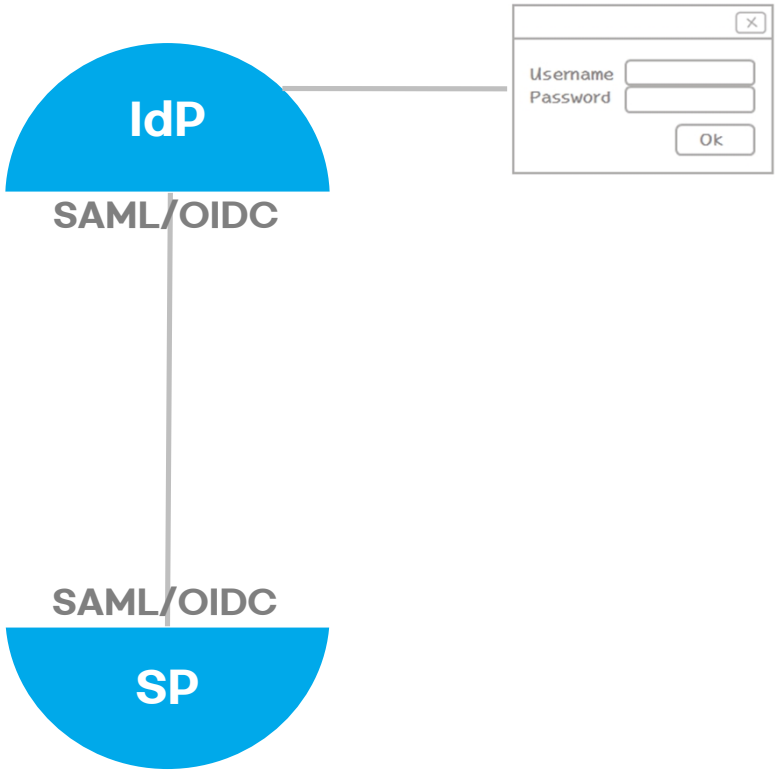


Part Ⅰ. SAML

KAFE

Background

Silo 모델



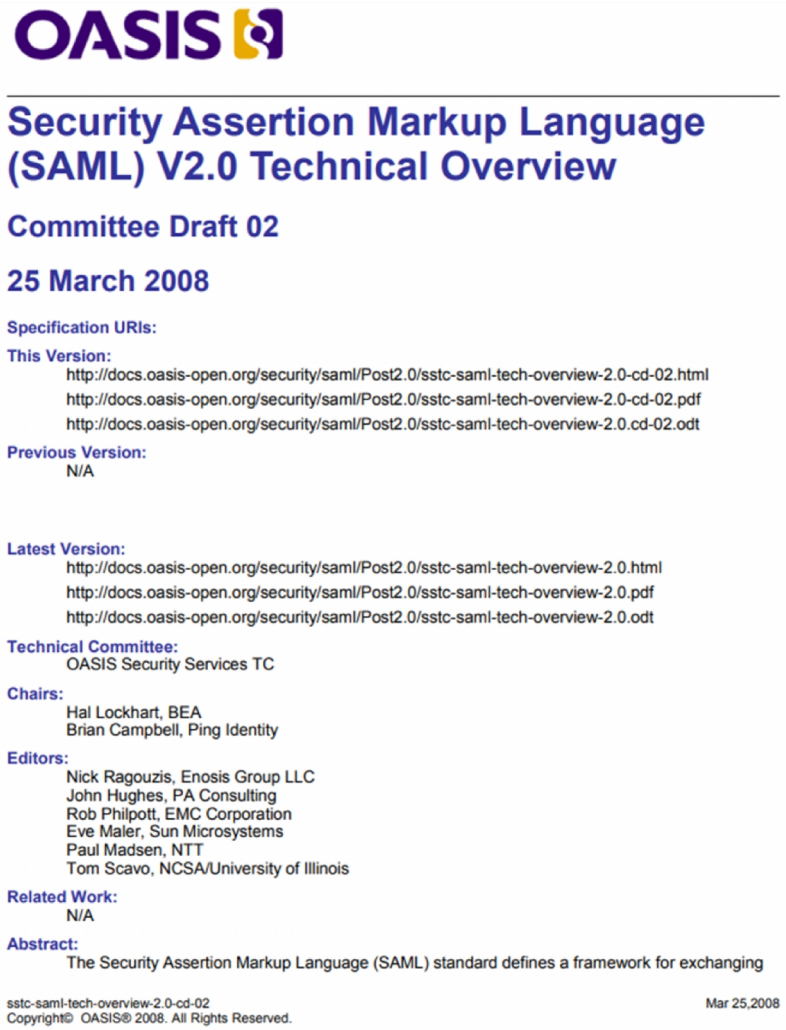
Federation 모델

SAML이란?

SAML [샘앨]

Security Assertion Markup Language
웹 통합인증(Single sign-on)을 위한 국제표준

보안 주장 표시 언어?



OASIS

Security Assertion Markup Language (SAML) V2.0 Technical Overview

Committee Draft 02

25 March 2008

Specification URIs:

This Version:
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0-cd-02.html
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0-cd-02.pdf
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.cd-02.odt

Previous Version:
N/A

Latest Version:
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.pdf
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.odt

Technical Committee:
OASIS Security Services TC

Chairs:
Hal Lockhart, BEA
Brian Campbell, Ping Identity

Editors:
Nick Ragouzis, Enosis Group LLC
John Hughes, PA Consulting
Rob Philpott, EMC Corporation
Eve Maler, Sun Microsystems
Paul Madsen, NTT
Tom Scavo, NCSA/University of Illinois

Related Work:
N/A

Abstract:
The Security Assertion Markup Language (SAML) standard defines a framework for exchanging

sstc-saml-tech-overview-2.0-cd-02 Mar 25,2008
Copyright© OASIS® 2008. All Rights Reserved.

주요 용어

주체(Subject)

¶ 인증의 대상이 되는 사람이나 사물

인증(Authentication)

¶ 디지털 식별자(예, 사용자 ID와 비밀번호)를 이용한 주체의 검증

"Verifying that the subject seeking access to a resource is the one previously identified and approved"

인가/권한부여(Authorization)

¶ 주체가 서비스/리소스에 접근 가능 여부를 결정

속성(Attribute)

¶ 사용자의 신원정보(예, 이메일, 전화번호 등)

단언(Assertion)

¶ 인증의 주체, 인증상태, 속성정보 등을 주장

SAML

웹 통합인증(싱글사인온)을 위한 XML 규약

SAML 1.1(`03), SAML2.0(`05, Errata(정오표)/`19)

¶ OASIS 보안서비스 기술위원회 제정

연구교육분야 활용(`05~)

¶ SWITCH(스위스 연구망)

강화된 보안 vs 적용 어려움

웹 환경 지원(모바일 미지원)

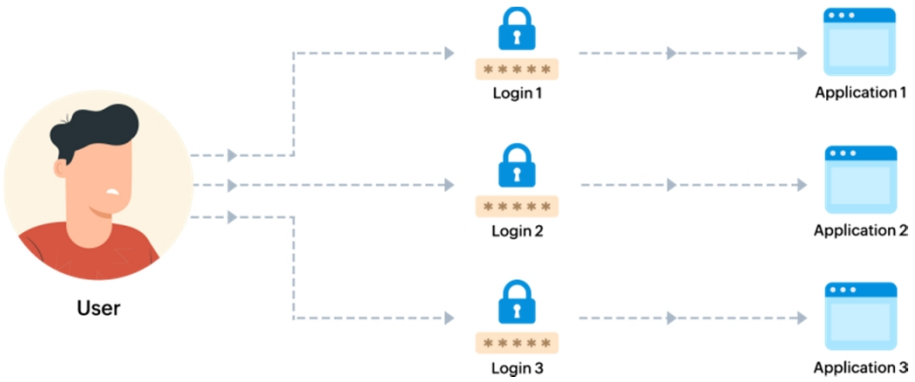
¶ OpenID Connect(OIDC) 출현

KAFE

통합인증(싱글사인온, SSO)

하나의 아이디와 한번의 로그인으로 다수 응용서비스 이용

¶ 보안성 ↑
¶ 편의성 ↑
¶ 관리비용 ↓



통합인증이 없는 경우

통합인증이 있는 경우

Image Source: ManageEngine.com

SAML 통합인증 시스템의 구성

아이디(신원정보) 제공자/Identity provider(IdP)

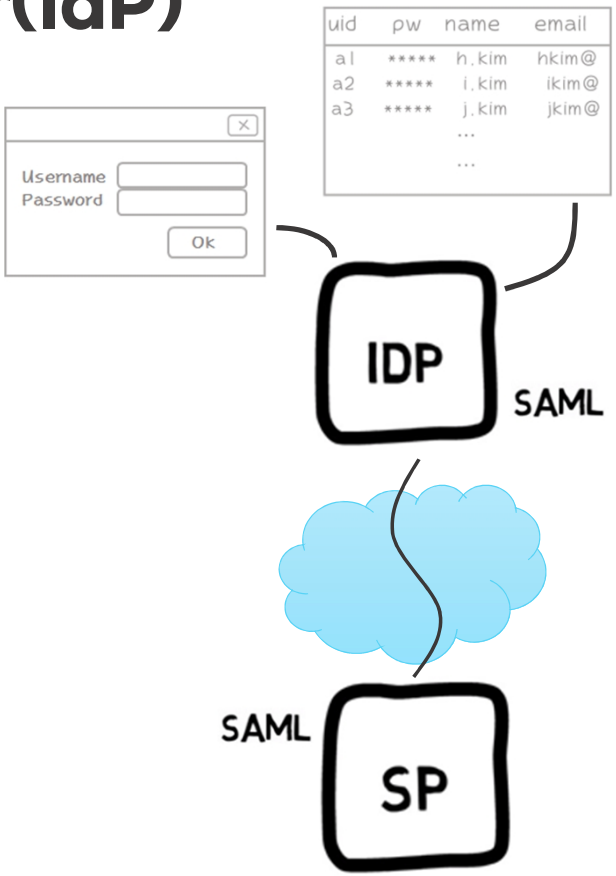
¶ 신원정보 관리
¶ 사용자 인증(로그인)
¶ SAML 메시지 처리(서명 등)

서비스 제공자/Service provider(SP)

¶ 사용자 인가(권한부여)
¶ SAML 메시지 처리(서명검증 등)

특징

¶ IdP와 SP가 분리되어 있음
¶ 일반적으로 HTTP를 전송규약으로 이용



KAFE

SAML 통합인증 절차

① 메타데이터 교환
② 인증요청(로그인요청)
③ 인증응답

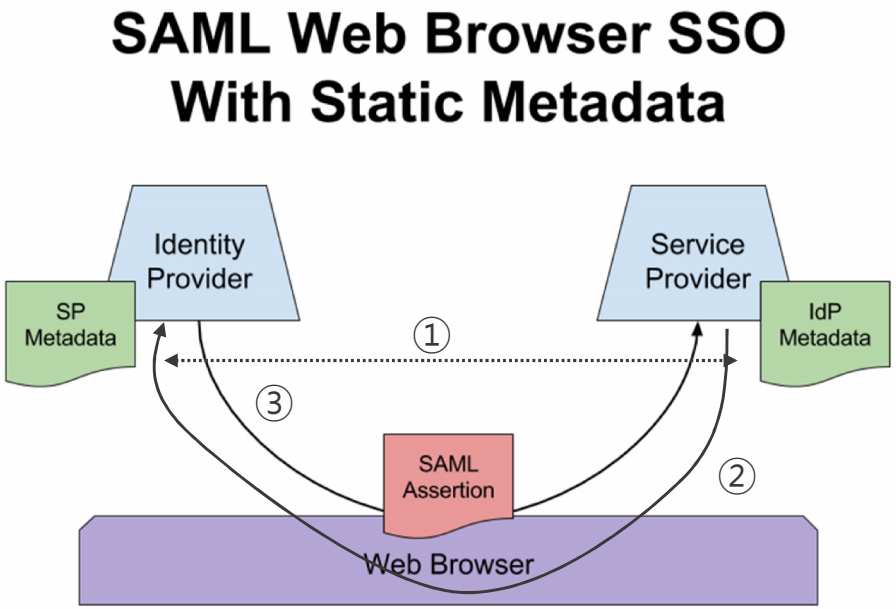


Image Source: https://en.wikipedia.org/wiki/SAML_metadata

SAML Metadata

아이디제공자/서비스제공자의 시스템 정보를 저장한 XML 파일

내용

- ¶ 개체식별자(EntityID)
- ¶ 기관도메인(Scope)
- ¶ 암호키(서명/암호화)
- ¶ 서비스 주소
  - ＊ Protocol endpoints
- ¶ 부가정보

SAML Request

서비스제공자가 전송하는 인증요청 메시지

메시지 내용

- ¶ SSO 서비스 주소
- ¶ ACS 서비스 주소
  - * Assertion Consumer Service
- ¶ 서비스제공자 식별자
- ¶ 서명
- ¶ 인증방법(ID/PW, OTP 등)

등등

```
<samlp:AuthnRequest
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="_bc6335d0970e40d7d40c161ca9eladcffd2d47c3d4"
    Version="2.0"
    IssueInstant="2022-11-14T08:17:21Z"
    Destination="https://saml.kafe.or.kr/simplesaml/saml2/idp/SSOService.php"
    AssertionConsumerServiceURL="https://webinar.kafe.or.kr/simplesaml/module.php/saml/sp/saml2-acs.php/default-sp"
    ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST">
    <saml:Issuer>https://webinar.kafe.or.kr/sp/simplesamlphp</saml:Issuer>
    <samlp:NameIDPolicy
        Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"
        AllowCreate="true"/>
    <samlp:Scoping>
        <samlp:RequesterID>https://webinar.kafe.or.kr/sp/python</samlp:RequesterID>
    </samlp:Scoping>
</samlp:AuthnRequest>
```

KAFE

SAML Response with Assertion

인증요청에 대한 응답으로 아이디제공자가 전송

메시지 내용

¶ 발행자(Issuer)

¶ 서명(Signature)

＊ digest(간추림)를 개인키로 암호화

¶ X.509 인증서

＊ 서명검증 및 복호화

¶ 인증정보

＊ 누가, 언제, 어떤 서비스제공자를 위해
어떤 인증방식(예, 비밀번호)으로 로그인했고

＊ 메시지가 언제까지 유효한지를 기록

¶ 속성값

<참고> SAML에서 서명과 검증은 **매우** 중요한 보안요소입니다.

Digital Signature

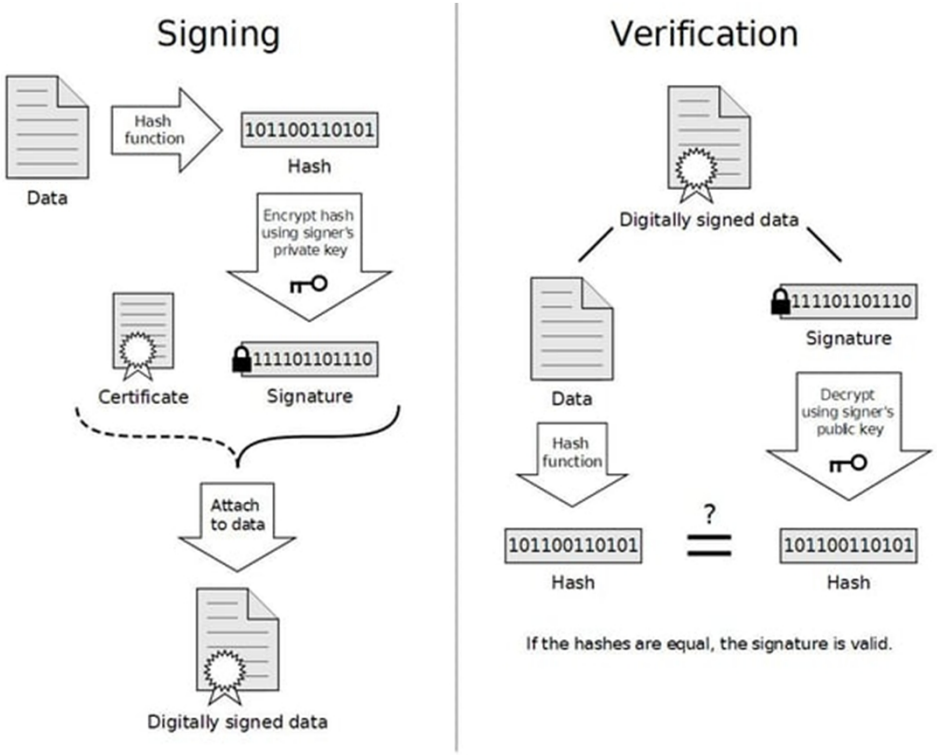


Image Source: https://www.identityfusion.com/

```
<saml:Assertion                                                                    ①
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    ID="_4ee1121c24771a9cf31ef208923f12f5d9e24d8722"
    Version="2.0"
    IssueInstant="2022-11-14T08:17:35Z">
    <saml:Issuer>https://saml.kafe.or.kr/idp/simplesamlphp</saml:Issuer>
    <ds:Signature                                                                  ②
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
            <ds:CanonicalizationMethod
                Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
            <ds:SignatureMethod
                Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
            <ds:Reference
                URI="#_4ee1121c24771a9cf31ef208923f12f5d9e24d8722">
                <ds:Transforms>
                    <ds:Transform
                        Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
                    <ds:Transform
                        Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                </ds:Transforms>
                <ds:DigestMethod
                    Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                <ds:DigestValue>
                    whUztIpvkU4o9p3kJbm3aVVylx1biAyZLfWzYyj6vK8=
                </ds:DigestValue>
            </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>M3Hj+Vzab43p3JWluUUIH1aVZNIDFfbZn+U32CmbAUAjCCVh4yJnMwkvsz6G8iGiOrUA+N1LO8N24oa1VjnpTsBUBCkhRW69UgdDf/V0
            ZpOAYuR1USWIQw==
        </ds:SignatureValue>
        <ds:KeyInfo>
            <ds:X509Data>
                <ds:X509Certificate>MIIDOTCCArmgAwIBAgIJAOMgLpVea1i4MA0GCSqGSIb3DQEBCwUAMH8xCzAJBgNVBAYTAktSMRAwDgYDVQQHDAdEYWVqZW9
            </ds:X509Data>
        </ds:KeyInfo>
    </ds:Signature>
    <saml:Subject>
        <saml:NameID
            SPNameQualifier="https://webinar.kafe.or.kr/sp/simplesamlphp"
            Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient">_9c25d611ebb697df5b96798d37f029a347b7586685
        </saml:NameID>
        <saml:SubjectConfirmation
            Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
            <saml:SubjectConfirmationData
```

Attributes in SAML Assertion

Assertion에 속성(Attributes) 포함

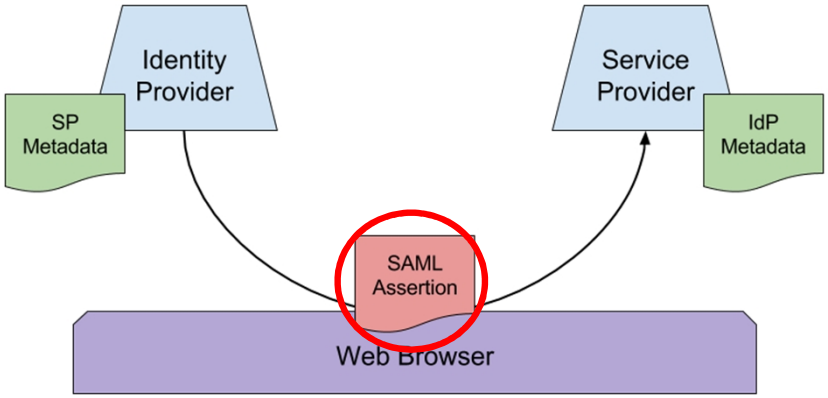
Friendly name/OID 형식 표기

¶ Friendly name

(예): cn/commonName, sn/surName,

¶ OID(Object Identifier)

(예): URN:OID:2.5.4.3



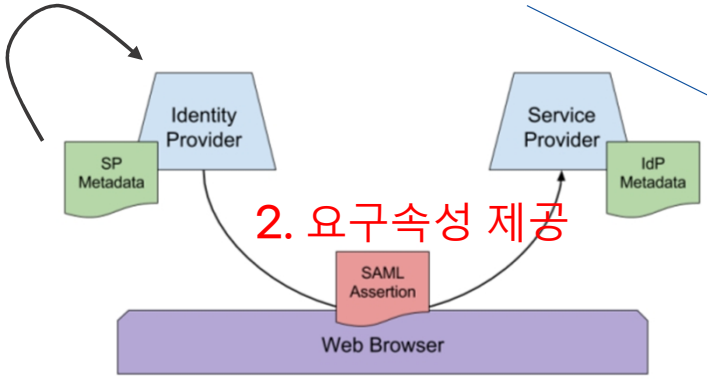
displayName
- SAML Name: urn:oid:2.16.840.1.113730.3.1.241
- LDAP source attribute: suDisplayname
- Example: Prof. John Doe

```
<saml:AttributeStatement>
   <saml:Attribute
      Name="urn:oid:2.16.840.1.113730.3.1.241"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
         xsi:type="xs:string">Jinyong JO
      </saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute
      Name="urn:oid:0.9.2342.19200300.100.1.3"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
         xsi:type="xs:string">jinyong.jo@gmail.com
      </saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute
      Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
         xsi:type="xs:string">staff
      </saml:AttributeValue>
      <saml:AttributeValue
         xsi:type="xs:string">member
      </saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute
      Name="urn:oid:2.5.4.10"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
         xsi:type="xs:string">KISTI
      </saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute
      Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
         xsi:type="xs:string">jiny92@coreen.or.kr
      </saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute
      Name="urn:oid:1.3.6.1.4.1.25178.1.2.9"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
         xsi:type="xs:string">coreen.or.kr
      </saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute
      Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.10"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue>
         <saml:NameID
            NameQualifier="https://coreen-idp.kreonet.net/idp/simplesamlphp"
            SPNameQualifier="https://filesender.kreonet.net/sp/simplesamlphp"
            Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">d74f17c949df30493dc8cf7171959c01716f1d03
         </saml:NameID>
      </saml:AttributeValue>
   </saml:Attribute>
</saml:AttributeStatement>
```

Attributes provided

서비스제공자는 요구속성을 SP 메타데이터에 기록

1. 개체식별자(entityID) 및 요구속성 확인




```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata" xmlns:md="urn:oasis:n
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:saml="urn:oasis:names:tc:SAML:2.0:a
xmlns:shibmd="urn:mace:shibboleth:metadata:1.0" xmlns:mdui="urn:oasis:names:tc:SAML:met
xmlns:mdattr="urn:oasis:names:tc:SAML:metadata:attribute" xmlns:mdrpi="urn:oasis:names:
xmlns:idpdisc="urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol" xmlns:init=
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xi="http://www.w3.org/2001/
entityID="https://snu.bookcube.biz/sp/shibboleth">
  <md:Extensions>
   <mdrpi:RegistrationInfo registrationAuthority="http://kafe.kreonet.net" registrationI
  </md:Extensions>
```

...,

```
<md:AttributeConsumingService index="0">
 <md:ServiceName xml:lang="en">Bookcube for Seoul National University</md:ServiceName>
 <md:ServiceDescription xml:lang="en">It is a service that allows you to use e-books online. Users can
 <md:RequestedAttribute FriendlyName="eduPersonPrincipalName" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6" N
 <md:RequestedAttribute FriendlyName="eduPersonAffiliation" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1" Name
 <md:RequestedAttribute FriendlyName="email" Name="urn:oid:0.9.2342.19200300.100.1.3" NameFormat="urn:o
 <md:RequestedAttribute FriendlyName="displayName" Name="urn:oid:2.16.840.1.113730.3.1.241" NameFormat="
</md:AttributeConsumingService>
```

SAML SP Metadata

Attributes 활용

응용서비스가 사용자 권한부여에 사용

예시

¶ kisti.re.kr에 소속된 학생만 서비스 이용을 허용하라!

```
<saml:Attribute
            Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.9"
           NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:basic">
            <saml:AttributeValue
               xsi:type="xs:string">student@kisti.re.kr
            </saml:AttributeValue>
</saml:Attribute>
```

SAML Assertion 일부

<참고>
urn:oid:1.3.6.1.4.1.5923.1.1.1.9 == eduPersonScopedAffiliation

서비스제공자 != 응용서비스

KAFE

실습 #1

SP 메타데이터의 내용 확인

¶ 다운로드 받은 파일을 notepad++로 읽어오기

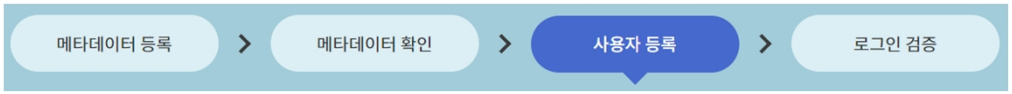
https://testssp.kreonet.net/simplesaml/module.php/saml/sp/metadata.php/default-sp

URL은 edu.kafe.or.kr를 참조

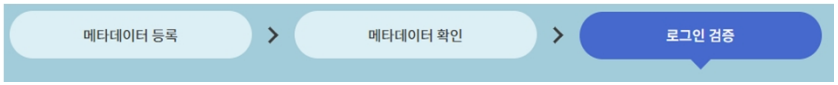
실습 #2

Chrome Plugin으로 SAML 메시지 내용 확인하기
¶ AuthnRequest/Response 메시지

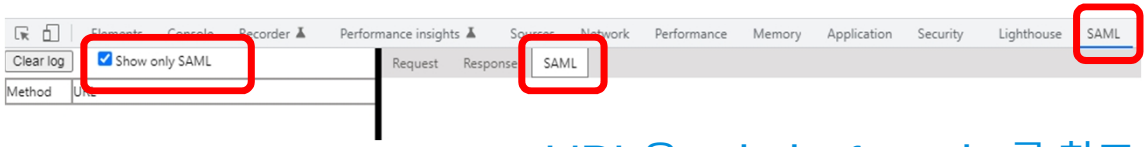
① [검증용 아이디제공자] https://testidp.kreonet.net로 이동
② 사용자 등록 메뉴에서 계정 생성
＊개인정보는 입력하지 마세요! 계정은 24시간 이후에 자동삭제 됩니다.



③ [검증용 서비스제공자] https://testssp.kreonet.net으로 이동
④ 로그인 검증 메뉴로 이동



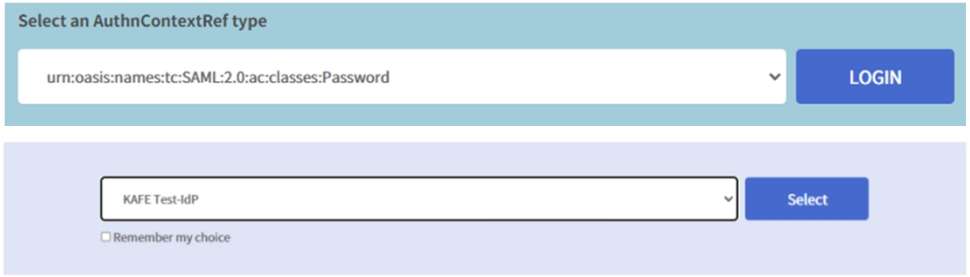
⑤ Chrome 브라우저에서 F12 키 입력
＊우측 그림 참조



URL은 edu.kafe.or.kr를 참조

실습 #2 (계속)

⑥ [검증용 서비스제공자] 로그인버튼 클릭



⑦ KAFE test-idp를 선택

⑧ SAML Request 메시지 확인
⑨ 검증용 아이디제공자에서 만든 사용자 계정으로 로그인
⑩ SAML Response 메시지 확인

URL은 edu.kafe.or.kr를 참조

[참고] Identity federation

KAFE

Identity federation(신원연합)

동일한 정책프레임워크 공유
협약에 의한 참여
연합운영자에 의해 관리

※ 우리나라의 신원연합은 KAFE (Korean Access Federation), 국가과학기술연구망에서 운영

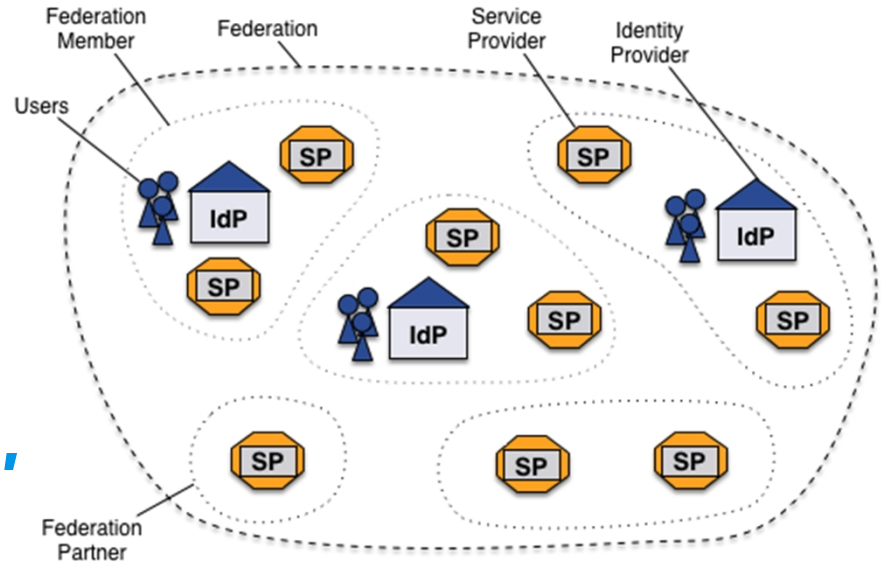


Image Source: Geant

KAFE

Federation operation

정책프레임워크 제공
메타데이터 관리
공통도구 제공
메시지중계(Optional)

※ 메시지중계가 아닐 경우,
아이디제공자와 서비스제공자가
브라우저를 통해 직접 SAML
메시지를 교환

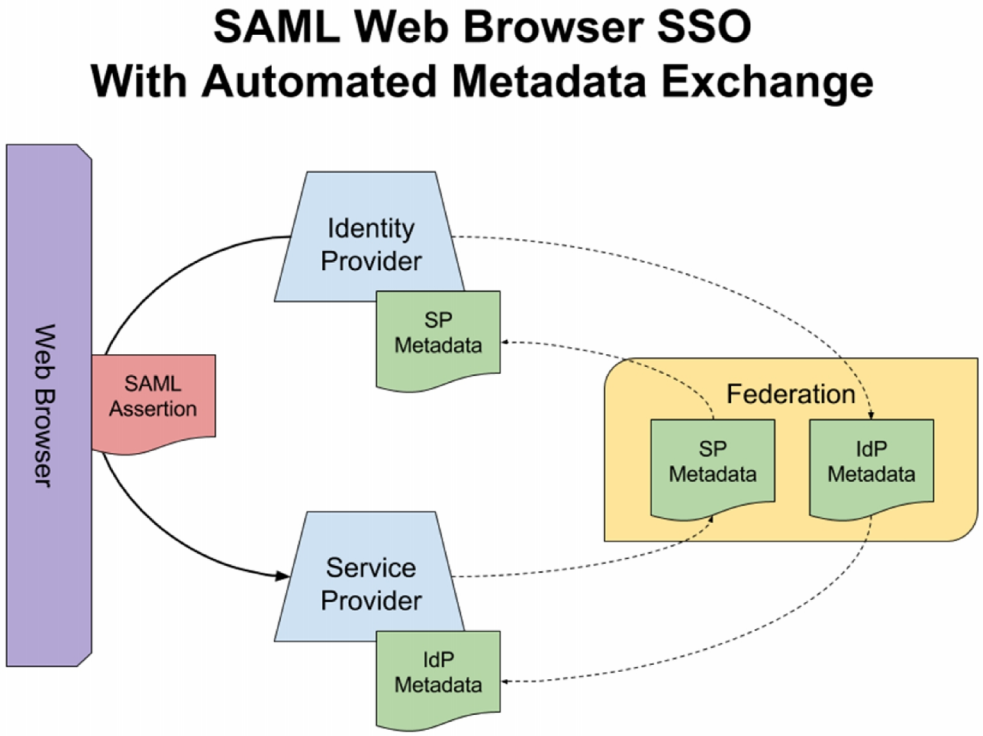


Image Source: https://en.wikipedia.org/wiki/SAML_metadata

Part Ⅱ. 아이디제공자의 구축

KAFE

공개소스 SAML 소프트웨어

분류	Shibboleth	simpleSAMLphp
인증 플로우	비밀번호, X.509, SPNEGO/Kerberos, IP Address, etc	비밀번호(SQL/LDAP/Radius), X.509, Social media
표준지원	SAML 1.1/2.0, X.509, Kerberos, LDAP, SQL	SAML 1.1/2.0, OpenID, Oauth 2.0, Kerberos, VOOT, SQL, LDAP, RADIUS
구현 언어	Java(Spring) for IdP, Apache daemon for SP	PHP for IdP and SP
고가용성(HA) 지원	Yes	Yes (다수의 memcached를 통해 가능)
License	GNU GPL v2.0	GNU LGPL v2.1
지원	Shibboleth 컨소시엄	사용자 커뮤니티

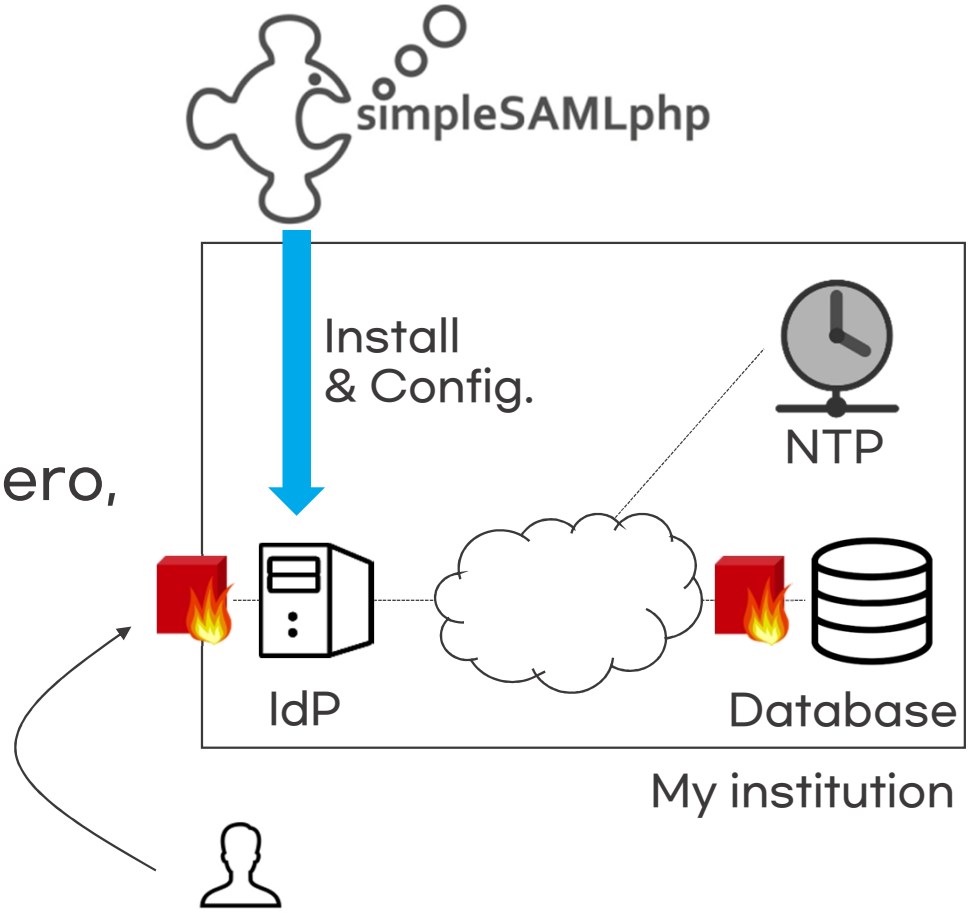
아이디제공자의 구축

¶ 아이디제공자(IdP)

* KAFE installer 이용
* NTP 시간동기화 및 TLS 통신 필수
* Apache 웹 서버에서 동작

¶ 데이터베이스

* MySQL, Oracle, LDAP, AD, Tibero, MSSQL, RESTful API, 등등



Why NTP?

¶ SAML response message

```
<saml:Conditions
    NotBefore="2022-11-21T02:14:44Z"
    NotOnOrAfter="2022-11-21T02:20:14Z">
    <saml:AudienceRestriction>
        <saml:Audience>                                                    </saml:Audience>
    </saml:AudienceRestriction>
</saml:Conditions>
<saml:AuthnStatement
    AuthnInstant="2022-11-21T02:02:26Z"
    SessionNotOnOrAfter="2022-11-21T10:15:14Z"
    SessionIndex="_f70c5e79d90cb0bbc119920844a3831194 0f28058 ">
    <saml:AuthnContext>
        <saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:Password</saml:AuthnContextClassRef>
    </saml:AuthnContext>
</saml:AuthnStatement>
```

¶ SAML Metadata

```
<md:EntitiesDescriptor ID="kafe-20221128022233" Name="urn:mace:kisti.re.kr:kafe:testfed" validUntil="2022-11-28T02:22:33Z"
 <md:Extensions>
  <mdrpi:PublicationInfo creationInstant="2022-11-21T02:22:33Z" publisher="http://kafe.kreonet.net"/>
 </md:Extensions>
 <!--
https://proxy.kreonet.net/sp/simplesamlphp
-->
 <md:EntityDescriptor entityID="https://proxy.kreonet.net/sp/simplesamlphp">
  <md:Extensions>
   <mdrpi:RegistrationInfo registrationAuthority="http://kafe.kreonet.net" registrationInstant="2015-11-10T08:32:00Z"/>
  </md:Extensions>
```

KAFE

데이터베이스 연동

①아이디제공자 서버에 데이터베이스(예, Oracle) 라이브러리 설치

I. http://www.oracle.com에서 Oracle instant client 다운로드: oracle instantclient basic, oracle instant client devel
II. Instant client 설치 rpm -ivh xxxx.rpm
III. http://pecl.php.net/package/oci8 에서 oci8 다운로드
※ oci8 버전은 php 버전에 의존성 있으므로 php 버전에 맞는 oci8버전을 다운로드
IV. oci8 설치

```
$ tar xvzf oci8-1.4.9.tgz
$ cd oci8-1.4.9
$ phpize
$ ./configure --with-oci8=shared,instantclient,/usr/lib/oracle/11.2/client64/lib
$ ln -s /usr/include/oracle/11.2/client64/ /usr/lib/oracle/11.2/client64/lib/include
$ make all install
```

V. php 연동: echo 'extension=oci8.so' > /etc/php.d/oci8.ini; service httpd restart

데이터베이스 연동(계속)

② 검증용 PHP 코드 작성

```
<?php
    $user_id = "USER_ID";
    $user_pw = "USER_PW";
    $tns =
"(DESCRIPTION=(ADDRESS_LIST=(LOAD_BALANCE=ON)(FAILOVER=ON)(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.10.10)(PORT=5055))(A
DDRESS=(PROTOCOL=TCP)(HOST=192.168.10.11)(PORT=5505)))(CONNECT_DATA=(SERVICE_NAME=HELLO_WORLD)))";
    $conn = oci_connect("DB_ID", "DB_PW", $tns, ' AL32UTF8 ');
    if(!$conn){
        $e = oci_error();
        var_dump($e);
    }else echo "success";

    $query = "SELECT NAME, EMAIL, AFFLI FROM TABLE(ORACLE_FUNC('$user_id', '$user_pw' ))";
    $stid = oci_parse($conn, $query);
    oci_execute($stid);
    if(!$row = oci_fetch_assoc($stid)){
        $e = oci_error();
        var_dump($e);
    }
    var_dump($row);
    oci_free_statement($stid);
    oci_close($conn);
?>
```

데이터베이스 연동(계속)

③simpleSAMLphp 소스코드 수정

＊ /var에 simplesamlphp가 설치된 경우

```
[root@localhost ~]# nano /var/simplesamlphp/modules/kafe/lib/Auth/Source/CoreAuth.php
```

＊ protected function login($username, $password) 함수 수정

☞ 아래 그림의 예시 임

```
protected function login($username, $password) {
    $db = new PDO($this->dsn, $this->username, $this->password);
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $db->exec("SET NAMES 'utf8'");

    $st = $db->prepare("SELECT username, password, name, email, affi from ex_users where username = '$username'");
    if(!$st->execute()){
        throw new Exception('Failed to Query database for the user');
    }
```

데이터베이스 연결부

데이터베이스 질의부

데이터베이스 연동(계속)

②simpleSAMLphp 소스코드 수정(contd.)

＊ 질의(Query) 결과를 SAML attributes로 변환

```
$attributes = array(
    'uid' => array($username),
    'displayName'    => array($row['name']),
    'cn'             => array($row['name']),
    'sn'             => $sn,
    'givenName'      => $givenname,
    'mail'           => array($row['email']),
    'eduPersonAffiliation'     => $this->eAffiliation,
    'eduPersonPrincipalName' => $eppName,
    'eduPersonScopedAffiliation' => $epsAffiliation,
    'organizationName'     => array('YOUR ORG FULL NAME'),
    'schacHomeOrganization' => array('yourorg.ac.kr'),
    // for future use
    'eduPersonEntitlement'     => array('urn:mace:dir:entitlement:common-lib-terms'),
);
```

＊ 이용 가능한 속성명 확인

```
[root@localhost ~]# nano /var/simplesamlphp/attributemap/name2oid.php
```

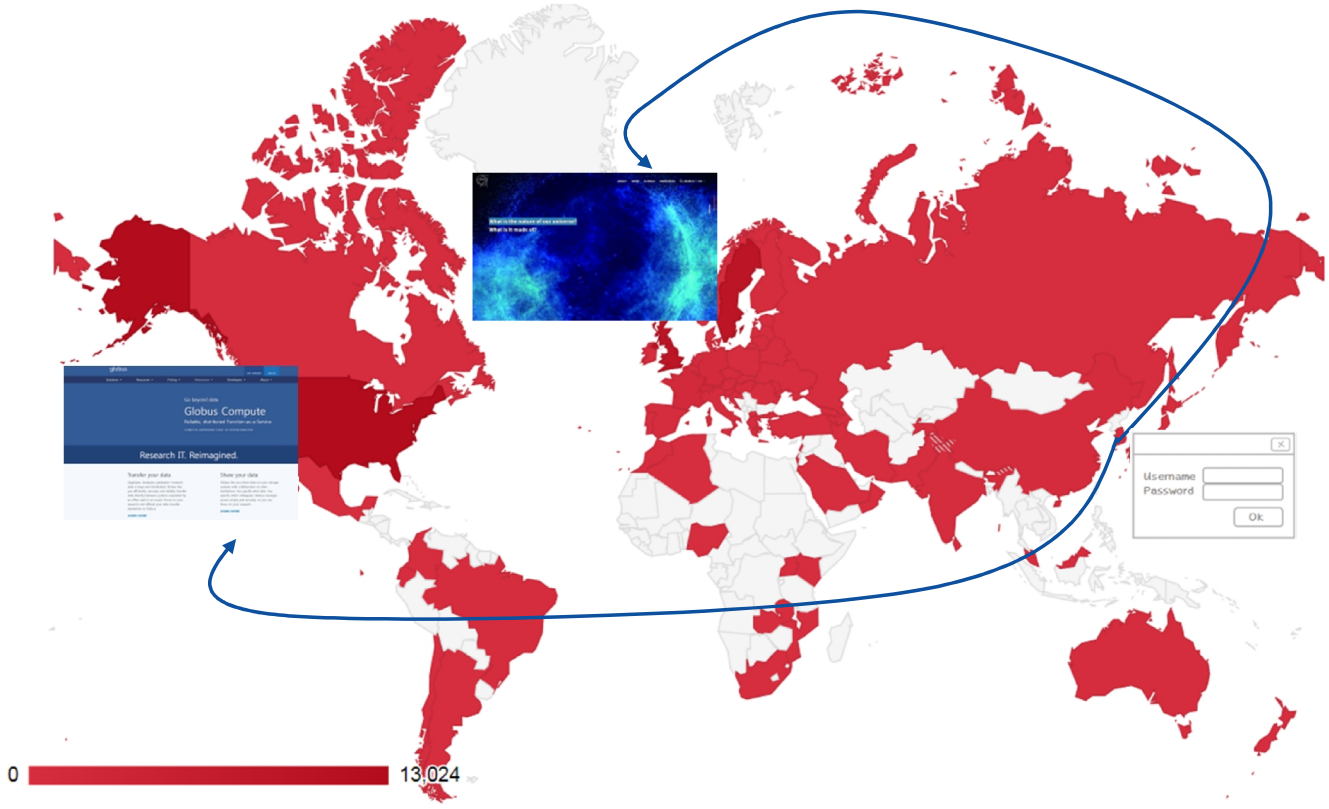
```
$attributemap = [
    'aRecord'                       => 'urn:oid:0.9.2342.19200300.100.1.26',
    'aliasedEntryName'              => 'urn:oid:2.5.4.1',
    'aliasedObjectName'             => 'urn:oid:2.5.4.1',
    'associatedDomain'              => 'urn:oid:0.9.2342.19200300.100.1.37',
    'associatedName'                => 'urn:oid:0.9.2342.19200300.100.1.38',
    'audio'                         => 'urn:oid:0.9.2342.19200300.100.1.55',
    'authorityRevocationList'       => 'urn:oid:2.5.4.38',
```

[참고] Inter-Federation

KAFE

Federations Map

¶ 아이디 페더레이션의 국가 간 확장



국외 응용서비스 이용

¶ 국제 신원연합 eduGAIN을 통해 연합 메타데이터 교환

＊ 개인정보처리방침을 제공하는 아이디제공자와 서비스제공자만 수용

＊ 여과된 eduGAIN 메타데이터 배포 주소(서비스제공자 확인 가능)
https://fedinfo.kreonet.net/signedmetadata/federation/KAFE-eduGAIN/eduGAIN-SP.xml

Federations in eduGAIN	
Participants	80
Voting-only Members	1
Candidates	5

Entities in eduGAIN	
All entities	8801
IdPs	5185
SPs	3631
Standalone AAs	2

¶ 조건

＊ eduGAIN 가입(KAFE 가입 시 선택)

＊ 아이디제공자의 경우, SIRTFI(보안사고 대응프레임워크) 및 Research & scholarship category(개인정보최소사용) 준수

```
<mdattr:EntityAttributes>
 <saml:Attribute Name="http://macedir.org/entity-category-support" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
  <saml:AttributeValue>http://refeds.org/category/research-and-scholarship</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="urn:oasis:names:tc:SAML:attribute:assurance-certification" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
  <saml:AttributeValue>https://refeds.org/sirtfi</saml:AttributeValue>
 </saml:Attribute>
</mdattr:EntityAttributes>
```

SIRTFI 중 TLP 규정

¶ TLP(Traffic Light Protocol)

* 민감정보의 공유범위를 지정
* 참조: https://www.cisa.gov/tlp

181.1.253.234, IPV4ADDR, ,IP_WATCHLIST, C2, 12/28/2017 00:10:25Z, TLP:WHITE, According t o DHS and FBI analysis, this IP address is compromised infrastructure. This IP is geolocated in Ar gentina.

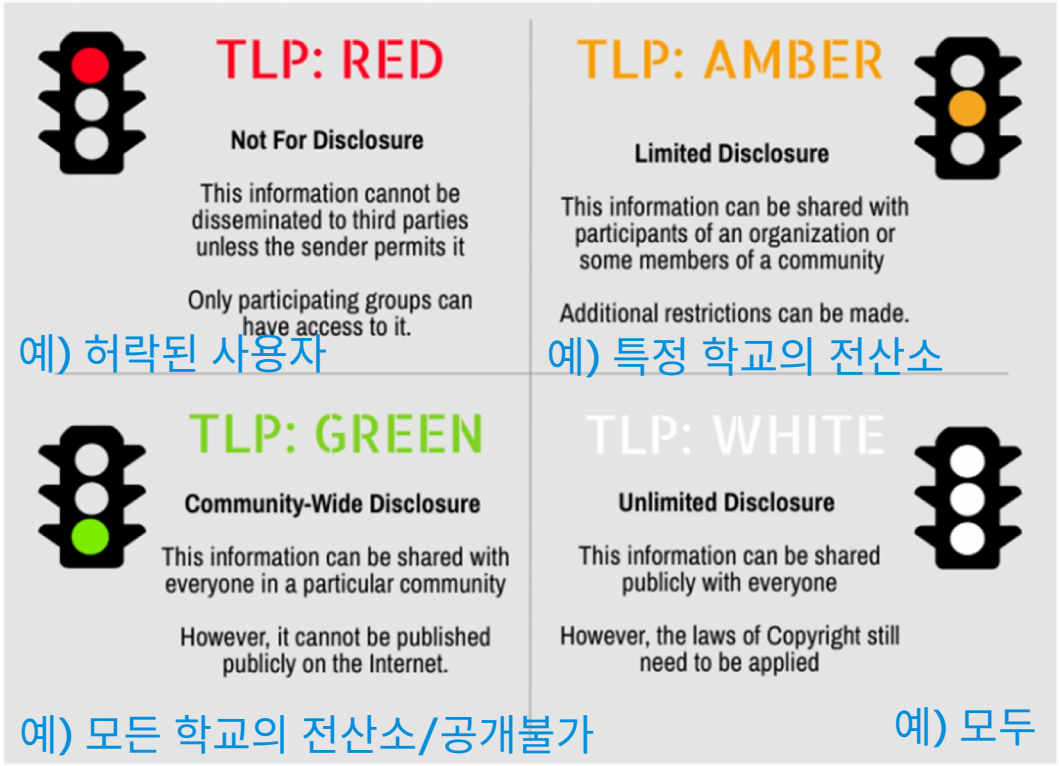


Image Source: https://cyberhoot.com/cybrary/traffic-light-protocol/

KAFE

Part Ⅲ. MFA

Multi-Factor Authentication

KAFE

MFA(Multi-Factor Authentication)

- **Something you know(지식)**
  eg., Password or PIN
- **Something you have(소유)**
  eg., token(smartphone, key, smart card)
- **Something you are(생체)**
  eg., 지문, 홍채, 안면인식

Image Source: https://www.rsa.com/multi-factor-authentication/what-is-mfa/

KAFE

Why MFA?

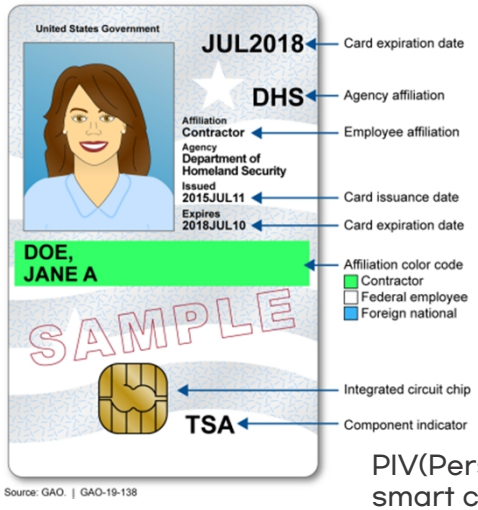
¶ 150억 개의 크리덴셜(예, 아이디/비밀번호 등)이 유출
¶ Google의 경우, 주 당 250,000개의 웹 로그인이 탈취
¶ 해킹 관련 데이터 유출의 약 80%가 탈취된/취약한 비밀번호로 인함

GDPR and NIST guidelines →

Strong authentication

Types of MFA

¶ Secure?



PIV(Personal Identity Verification) smart card

IVR: Interactive Voice Response

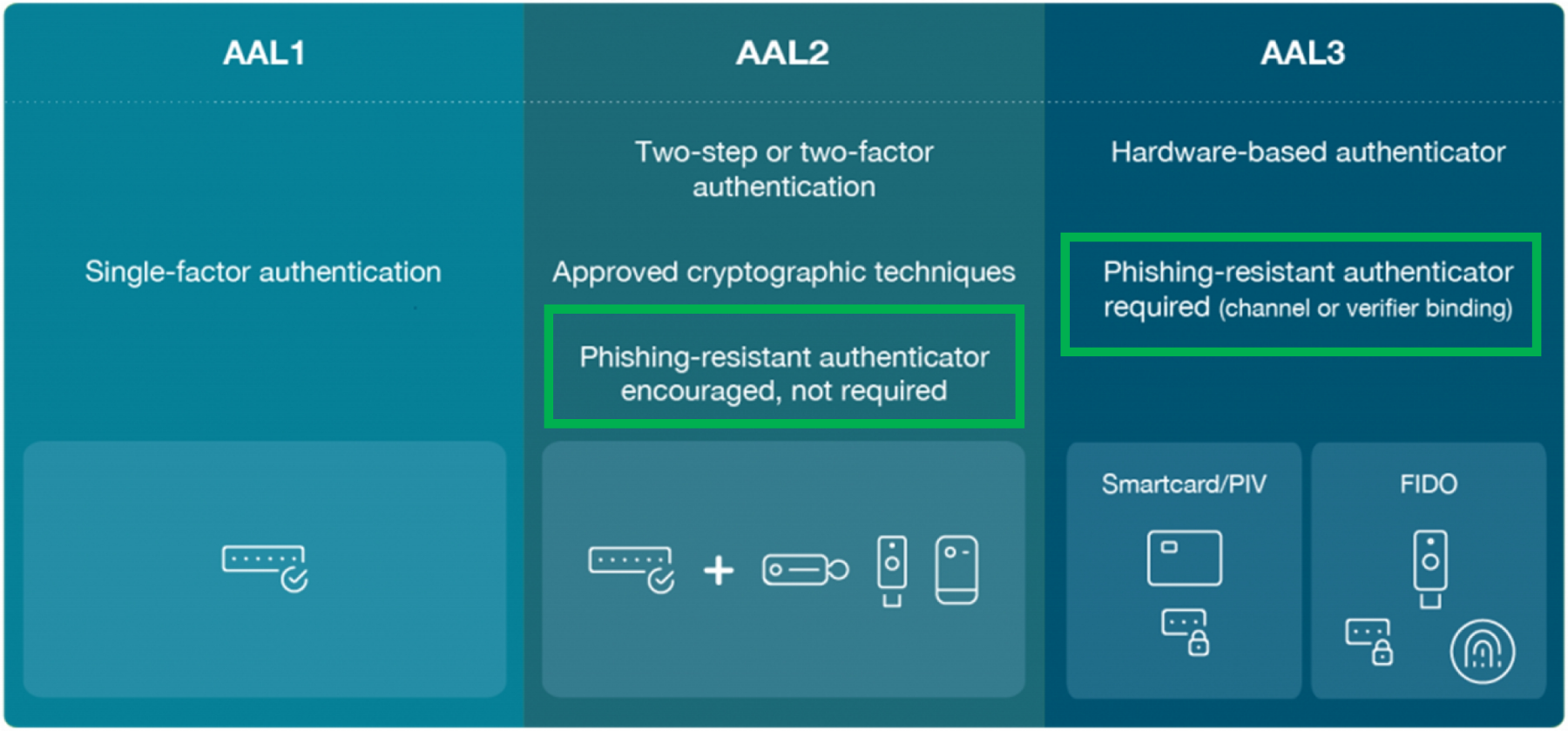
Type of MFA		NIST-Compliant	Phishing-Resistant	% Targeted Phishing Attacks Resisted[1]	Notes
Other	Last Sign-In Location	No	No	0%	
	Knowledge Questions	No	No	40%	
	Email OTP	No	No	75%	
	Desktop OTP	Yes	No	85%	
Mobile Authenticators	SMS OTP	No	No	75%	Lower cost and complexity, but mobile devices not always allowed or available
	IVR OTP	No	No	75%	
	App OTP (i.e. Google Authenticator)	Yes	No	80%	
	Push Notification	Yes	No	90%	
	FIDO2/WebAuthn mobile token	Yes	Yes	100%	
Hardware Authenticators	OTP hardware token	Yes	No	100%	Higher cost and complexity, but sometimes required
	FIDO2/WebAuthn security key	Yes	Yes	100%	
	PIV smart card	Yes	Yes	100%	
	PIV-derived authenticator	Yes	Yes	100%	

Image Source: https://surepassid.com/blog/2023/01/11/types-of-mfa-compared

KAFE

Authenticator Assurance Level

¶ NIST AAL



예) 소프트웨어 인증서, 비밀번호

예) 스마트폰 OTP, 하드웨어 OTP 토큰

예) 스마트카드, FIDO2 키

Image Source: https://www.yubico.com/

When apply?

¶ FIPS 201 (Federal Information Processing Standard Publication 201) - 3: 개인정보검증 사양서(연방정부 관련자)

불편함, 고통 또는 명예나 평판에 대한 손해

재정적 손실 또는 연방기관의 법적 책임

연방기관 프로그램이나 공익에 대한 피해

민감 정보의 무단 공개
개인의 안전
민사적 또는 형사적 법률 위반

Image Source: https://www.intercede.com/

When apply? (Contd.)

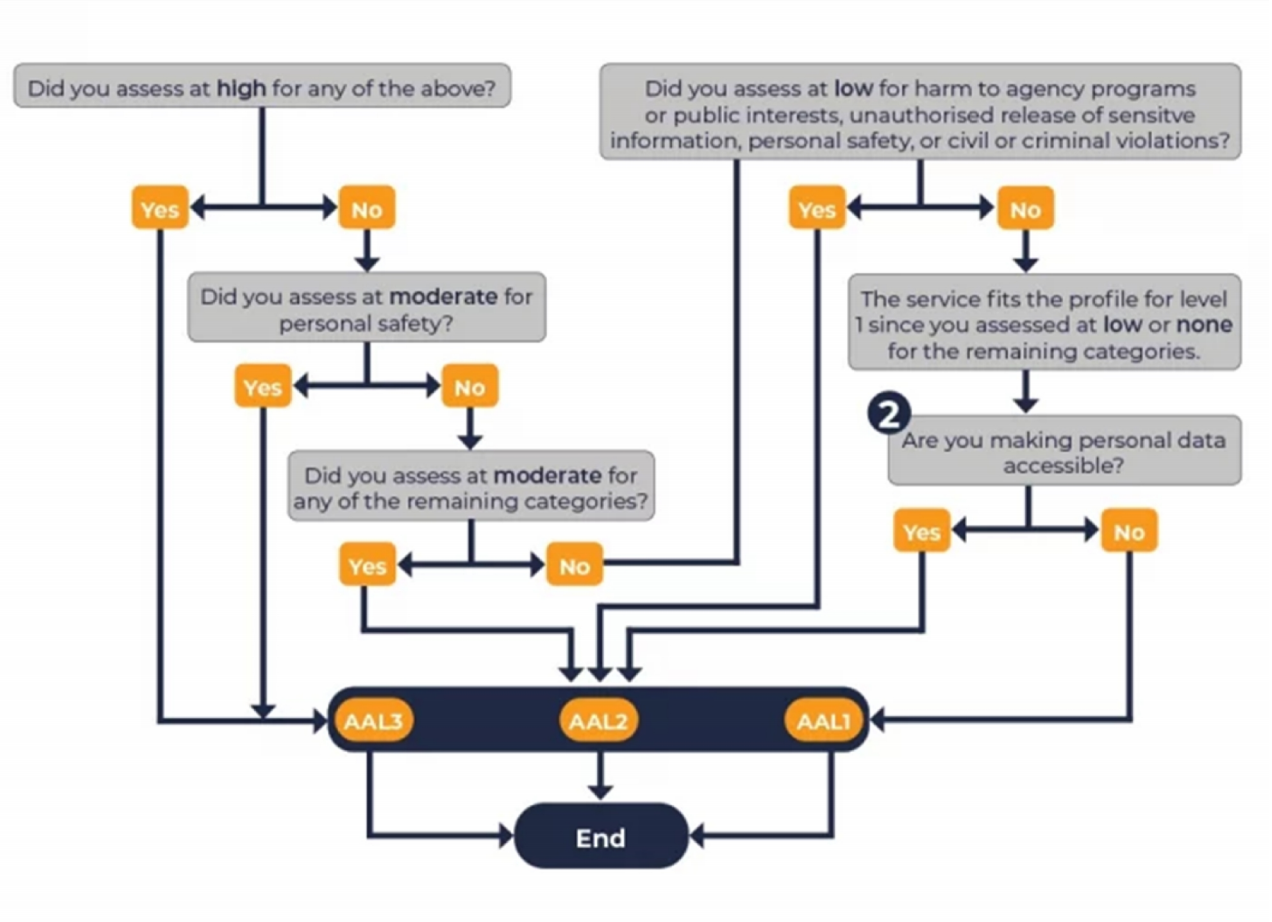
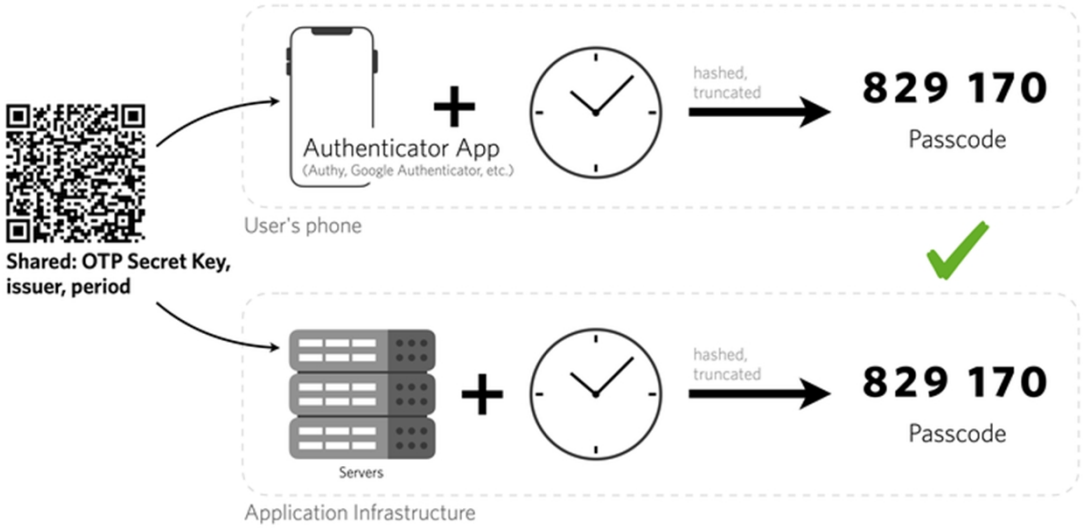


Image Source: https://www.intercede.com/

[참고] How TOTP works?

¶ TOTP: Time-based One-Time Password
RFC 6238



기준시간(Unix time)

현재시간(Unix time)

비밀키(Secret key)

$$TOTP_v = Truncate(HMAC_H(K, \frac{(T_c - T_o)}{T_x}))$$

갱신간격(Time step)

HMAC: Hash-based Message Authentication Code

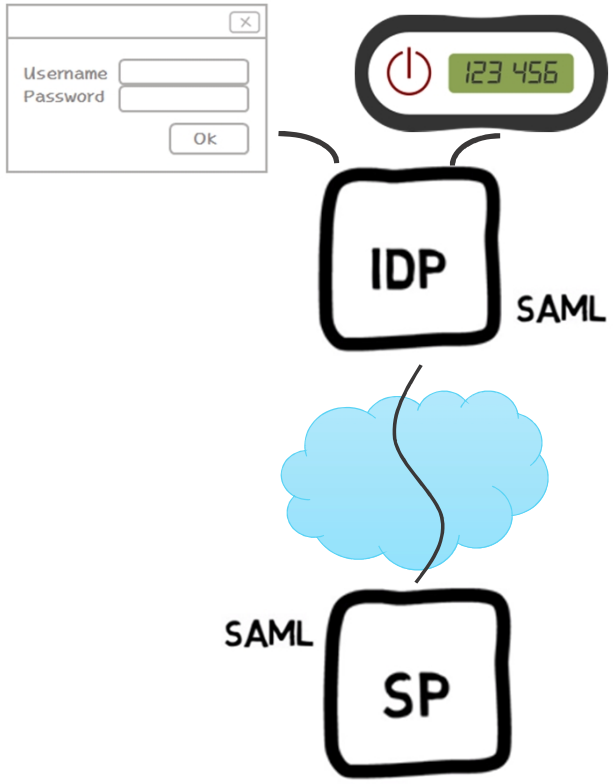
Image Source: https://www.twilio.com/

Federation 환경에서 MFA 실행 주체

¶ 아이디제공자

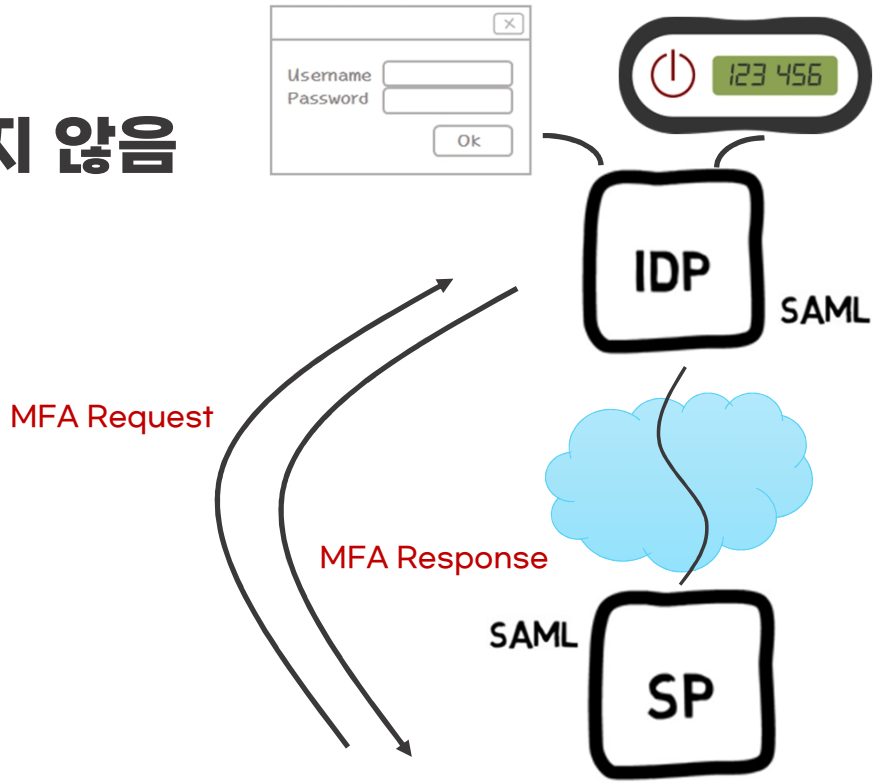
서비스제공자 제공 시,
사용자는 다수의 토큰을 보유해야 함

서비스제공자에서
MFA의 실행여부 확인 방법?



REFEDS MFA Profile

- ¶ 단체표준
- ¶ MFA의 실행 요청과 실행 확인
- ¶ MFA의 보장등급(AAL)은 지정하지 않음



REFEDS MFA Profile (Contd.)

¶ SAML Request & Response messages

MFA syntax: https://refeds.org/profile/mfa

MFA request message

```
<samlp:AuthnRequest
    Version="2.0"
    IssueInstant="2023-06-09T06:37:11Z">
    <samlp:RequestedAuthnContext>
        <saml:AuthnContextClassRef>
            https://refeds.org/profile/mfa
        </saml:AuthnContextClassRef>
    </samlp:RequestedAuthnContext>
        <samlp:RequesterID>https://oidc.kafe.or.kr/Saml2/sp</samlp:RequesterID>
    </samlp:Scoping>
</samlp:AuthnRequest>
```

MFA response message

```
<samlp:Response
    Version="2.0"
    IssueInstant="2023-06-09T06:37:40Z">
    <saml:Issuer>
        https://saml.kafe.or.kr/idp/simplesamlphp
    </saml:Issuer>
    <samlp:Status>
        <samlp:StatusCode
            Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
    </samlp:Status>
    <saml:Assertion>
        <saml:AuthnStatement>
            <saml:AuthnContext>
                <saml:AuthnContextClassRef>
                    https://refeds.org/profile/mfa
                </saml:AuthnContextClassRef>
                <saml:AuthenticatingAuthority>
                    https://coreen-idp.kreonet.net/idp/simplesamlphp
                </saml:AuthenticatingAuthority>
            </saml:AuthnContext>
        </saml:AuthnStatement>
    </saml:Assertion>
</samlp:Response>
```

실습 #3

MFA Request 확인

① 스마트폰에 Google authenticator 설치
② Chrome 브라우저를 이용해 https://testssp.kreonet.net 으로 이동
③ F12 클릭 후, SAML 메시지 캡쳐
④ '로그인 검증' 버튼 클릭
select an authncontextref type을 아래와 같이 설정한 후 'LOGIN' 클릭

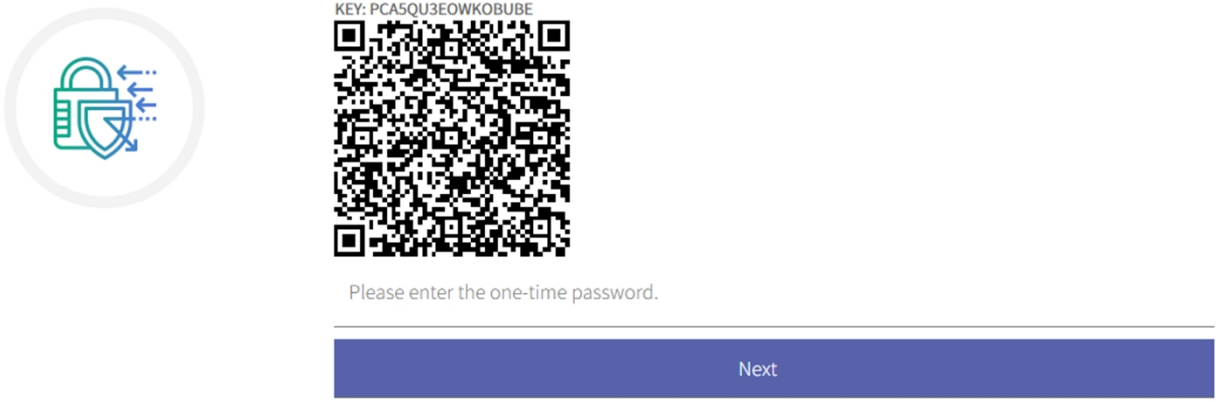
⑤ 아이디제공자 선택화면에서 'TEST ORG'를 선택
⑥ 캡쳐된 SAML Request 메시지 확인

URL은 edu.kafe.or.kr를 참조

실습 #3 (Contd.)

MFA Response 확인

⑦ 로그인 창에 아이디와 비밀번호를 입력: student11 ~ student19/student1234
⑧ 스마트폰의 Google authenticator로 QR 코드를 스캔; 생성된 OTP 입력



⑨ SAML response 메시지 확인

Korean Access Federation

https://www.kafe.or.kr/
support@kafe.or.kr