

IDENTITY DAYS

24 octobre 2019 -
PARIS

 @IdentityDays
#identitydays2019

Merci à tous nos
partenaires !





Maîtrisez les accès à vos applications Web (Cloud et On Premise) avec LemonLDAP::NG

Clément OUDOT
Identity Solutions Manager, Worteks

Clément OUDOT

@clementoudot

Responsable du
département Identité
chez Wortexs
@wortexs_com

Contributeur des logiciels
LemonLDAP::NG, LDAP
Tool Box, LSC et
FusionIAM



AGENDA DE LA CONFÉRENCE

- Définition du SSO
- Le logiciel LemonLDAP::NG
- Les protocoles CAS, SAML et OpenID Connect
- Stratégies Cloud / On Premise



Single Sign On

Imagine SSOng

Imagine there are no passwords
Or maybe just only one
A single secured form
To access our applications
Imagine all the users
Loving security

You may say
I'm a hacker
But I'm not the only one
I hope one day
You will log in
Using the Single Sign On

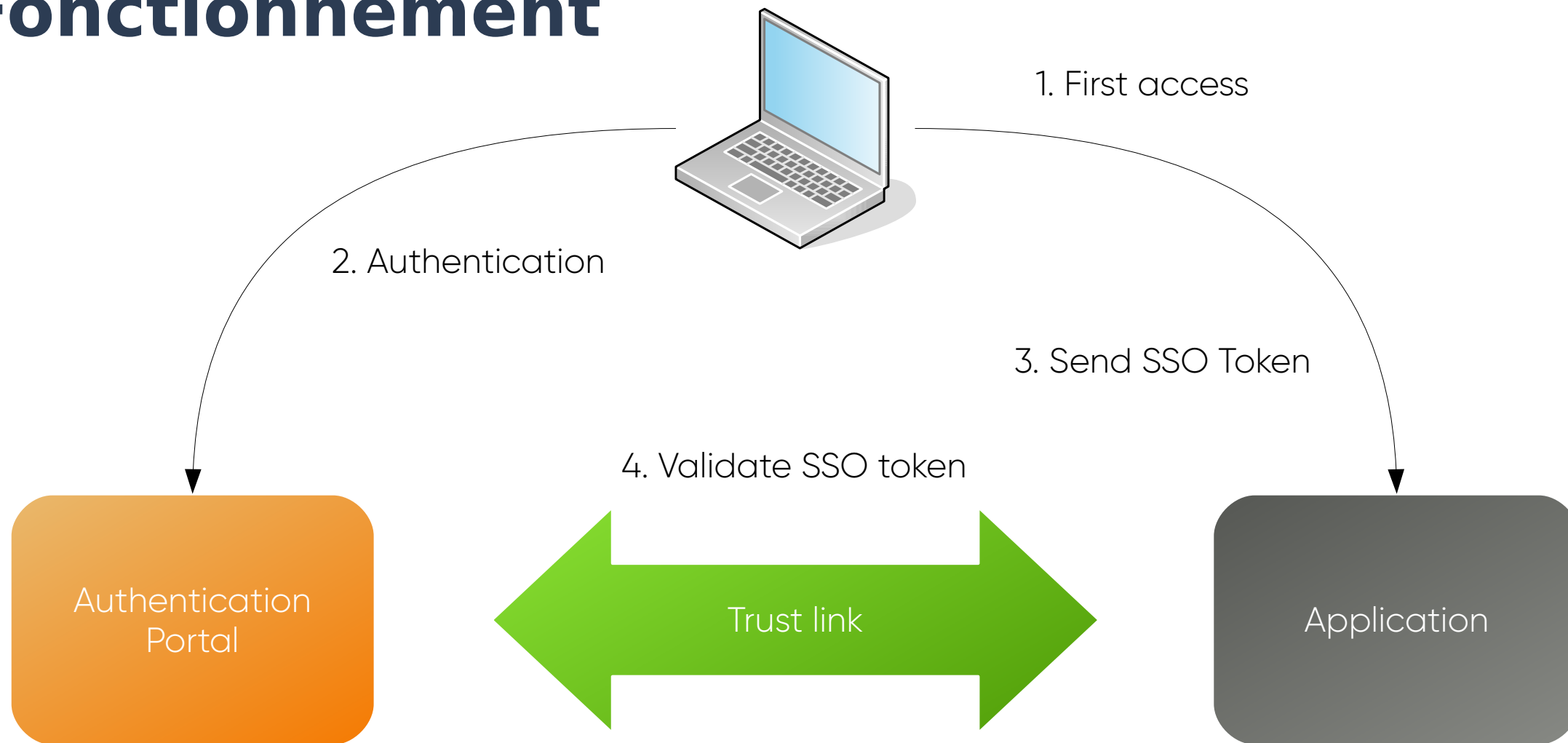
Imagine applications
No more storing passwords
Relying on a token
Even for authorizations
Imagine all developers
Loving security

Imagine some protocols
Made by clever people
CAS, OpenID or SAML
Even WS Federation
Imagine authentication
Interoperability



© John Lennon

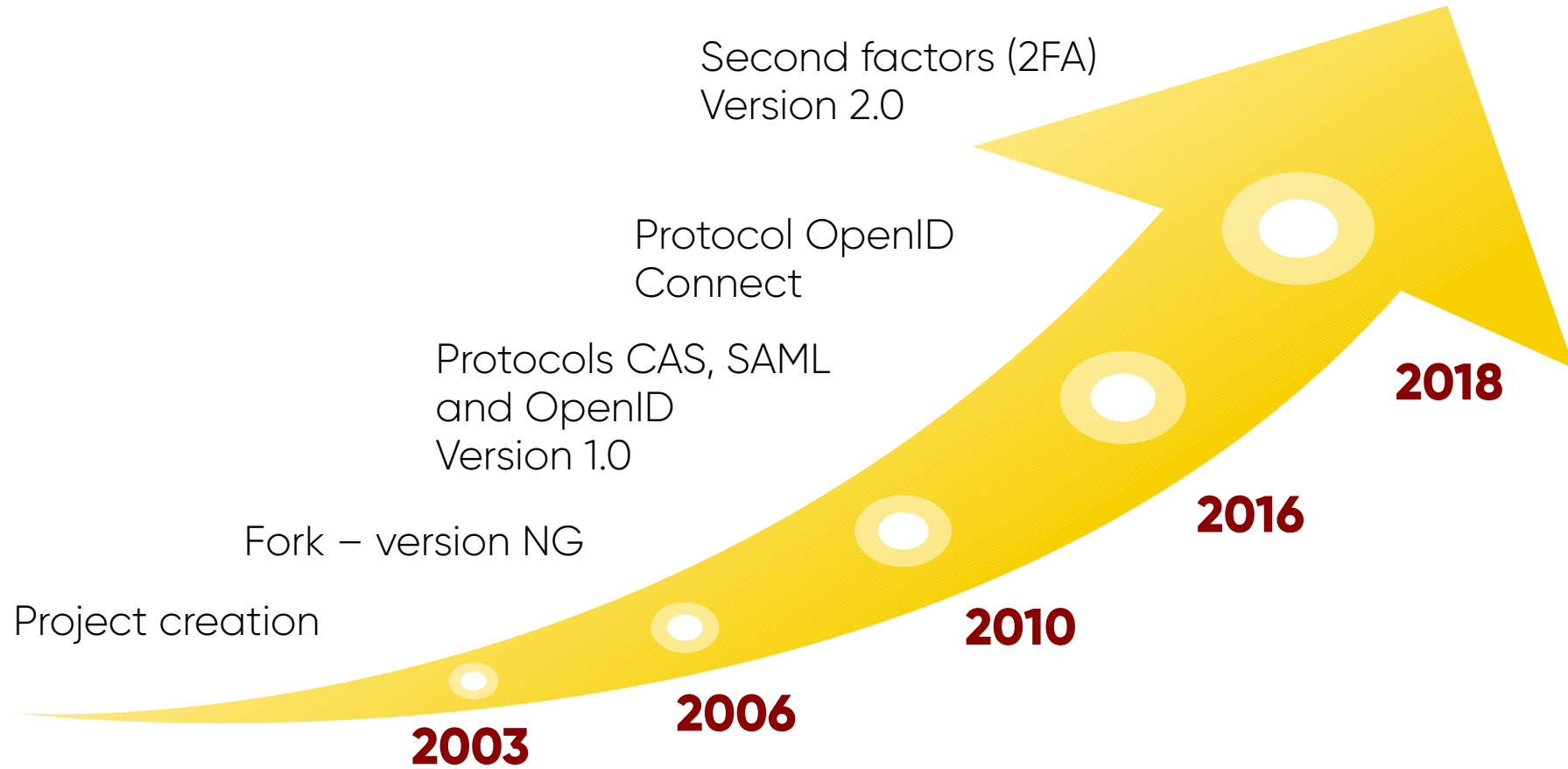
Fonctionnement





Le logiciel LemonLDAP::NG

Historique



Logiciel libre et communautaire

- Licence GPL (GNU Public License) :
 - Liberté 0. La liberté d'exécuter le logiciel, pour n'importe quel usage ;
 - Liberté 1. La liberté d'étudier le fonctionnement d'un programme et de l'adapter à ses besoins, ce qui passe par l'accès aux codes sources ;
 - Liberté 2. La liberté de redistribuer des copies ;
 - Liberté 3. L'obligation de faire bénéficier la communauté des versions modifiées.
- Projet OW2 : <https://gitlab.ow2.org/lemonldap-ng/lemonldap-ng>
- Développé principalement par la Gendarmerie Nationale et Worteks
- Déployé dans la majorité des Ministères et collectivités locales



Fonctionnalités principales



Single Sign On



Contrôle d'accès



Authentication
multi-facteurs

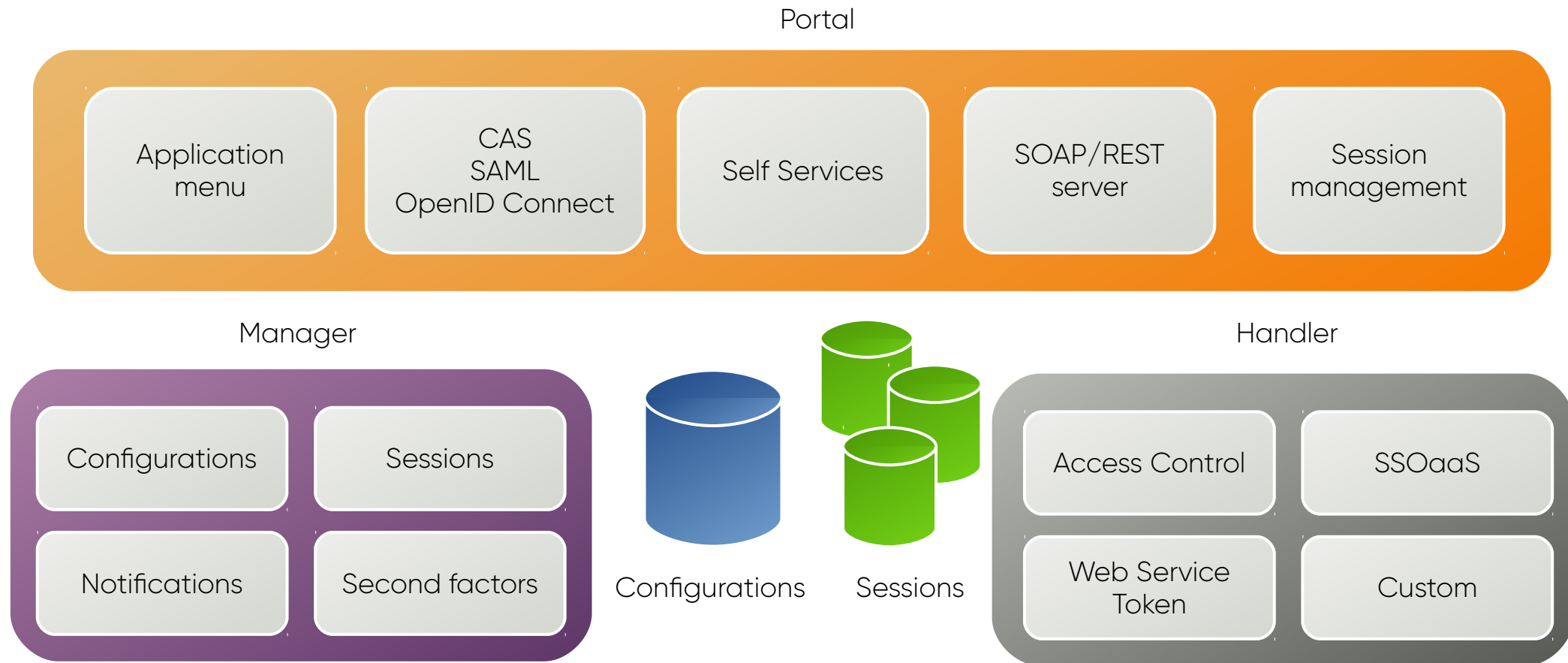


Multi-protocoles



Applications Web et API

Rôles des composants





Authentication required

 Login

 Password

☐ Check my last logins

 Connect

 Reset my password

 Create an account




[⚙ Configuration](#)
[📄 Sessions](#)
[🔔 Notifications](#)
[🔑 Second Factors](#)
[Menu ▾](#)
[> General Parameters](#)
[> Variables](#)
[> Virtual Hosts](#)
[> SAML2 Service](#)
[> SAML Identity Providers](#)
[> SAML Service Providers](#)
[> OpenID Connect Service](#)
[> OpenID Connect Providers](#)
[> OpenID Connect Relying Parties](#)
[> CAS Service](#)
[> CAS Servers](#)
[> CAS Applications](#)

[Save](#)
[⚙ Browse ▾](#)
[👁 Show help](#)
[📄 Download it](#)
[🔄 Restore](#)

Current configuration

Number	1
Author	The LemonLDAP::NG team
Author IP address	127.0.0.1
Date	04/04/2015 à 11:13:28
Configuration version	2.0.0
Resume	Default configuration provided by LemonLDAP::NG team

```

root@ader-worteks:~# /usr/share/lemonldap-ng/bin/lemonldap-ng-cli info

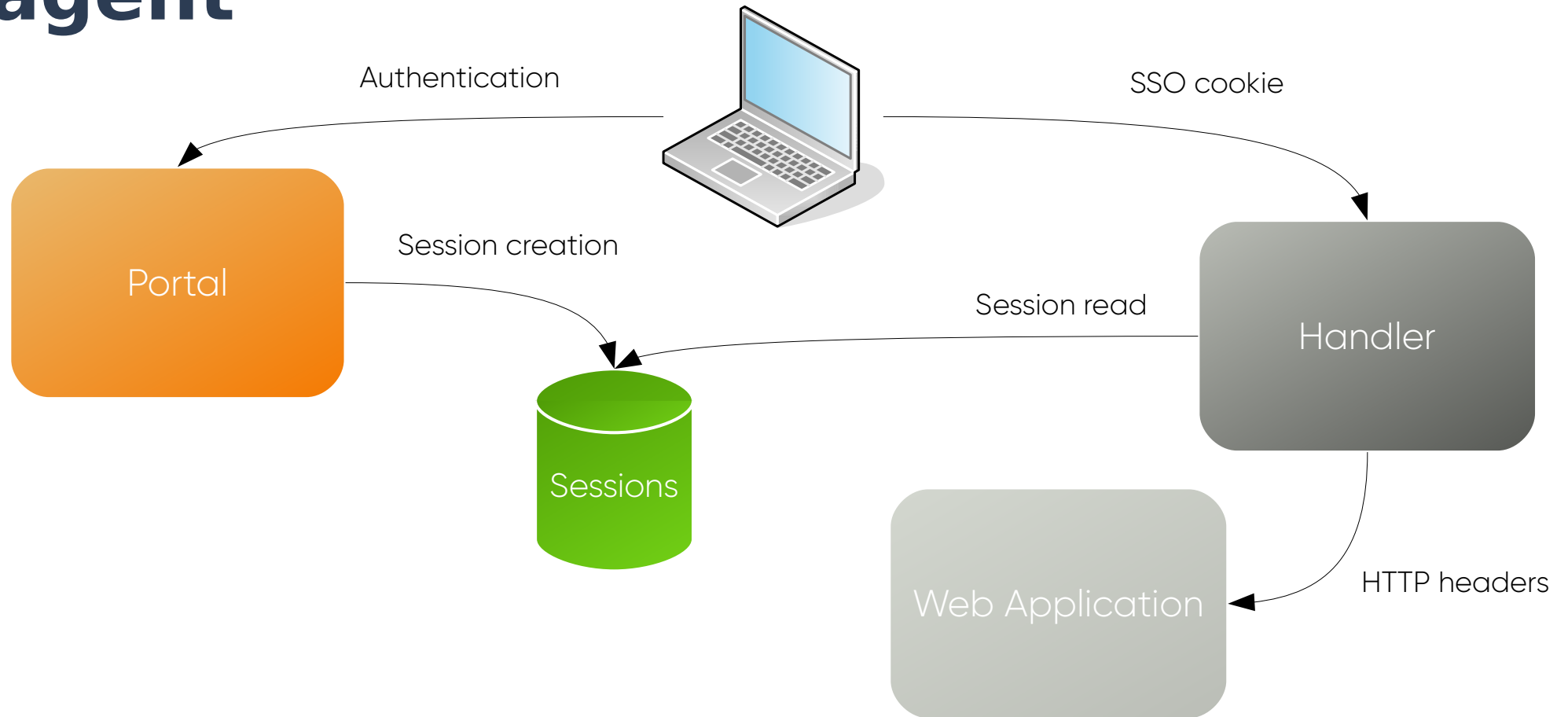
Num      : 88
Author   : clement
Author IP: localhost
Date     : Tue Dec 18 09:57:58 2018
Log      : Edited by lmConfigEditor
root@ader-worteks:~# /usr/share/lemonldap-ng/bin/lemonldap-ng-cli help
Usage: /usr/share/lemonldap-ng/bin/lemonldap-ng-cli <options> action <parameters>

Available actions:
- help           : print this
- info          : get currentconfiguration info
- update-cache   : force configuration cache to be updated
- get <keys>     : get values of parameters
- set <key> <value> : set parameter(s) value(s)
- addKey <key> <subkey> <value> : add or set a subkey in a parameter
- delKey <key> <subkey> : delete subkey of a parameter

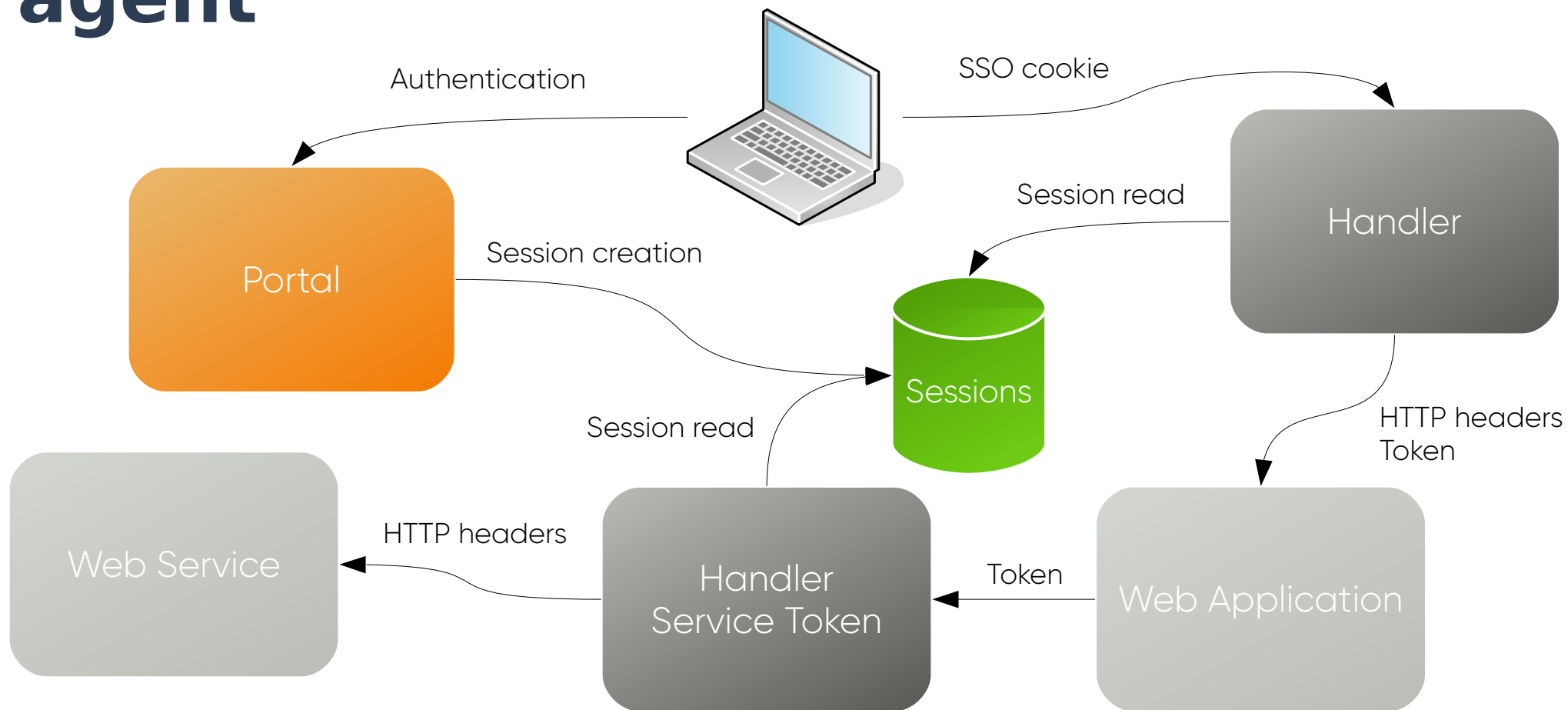
See Lemonldap::NG::Common::Cli(3) or Lemonldap::NG::Manager::Cli(3) for more
root@ader-worteks:~# /usr/share/lemonldap-ng/bin/lemonldap-ng-cli set ldapServer 'ldap://ldap.example.com'

```

Protection d'une application Web par agent



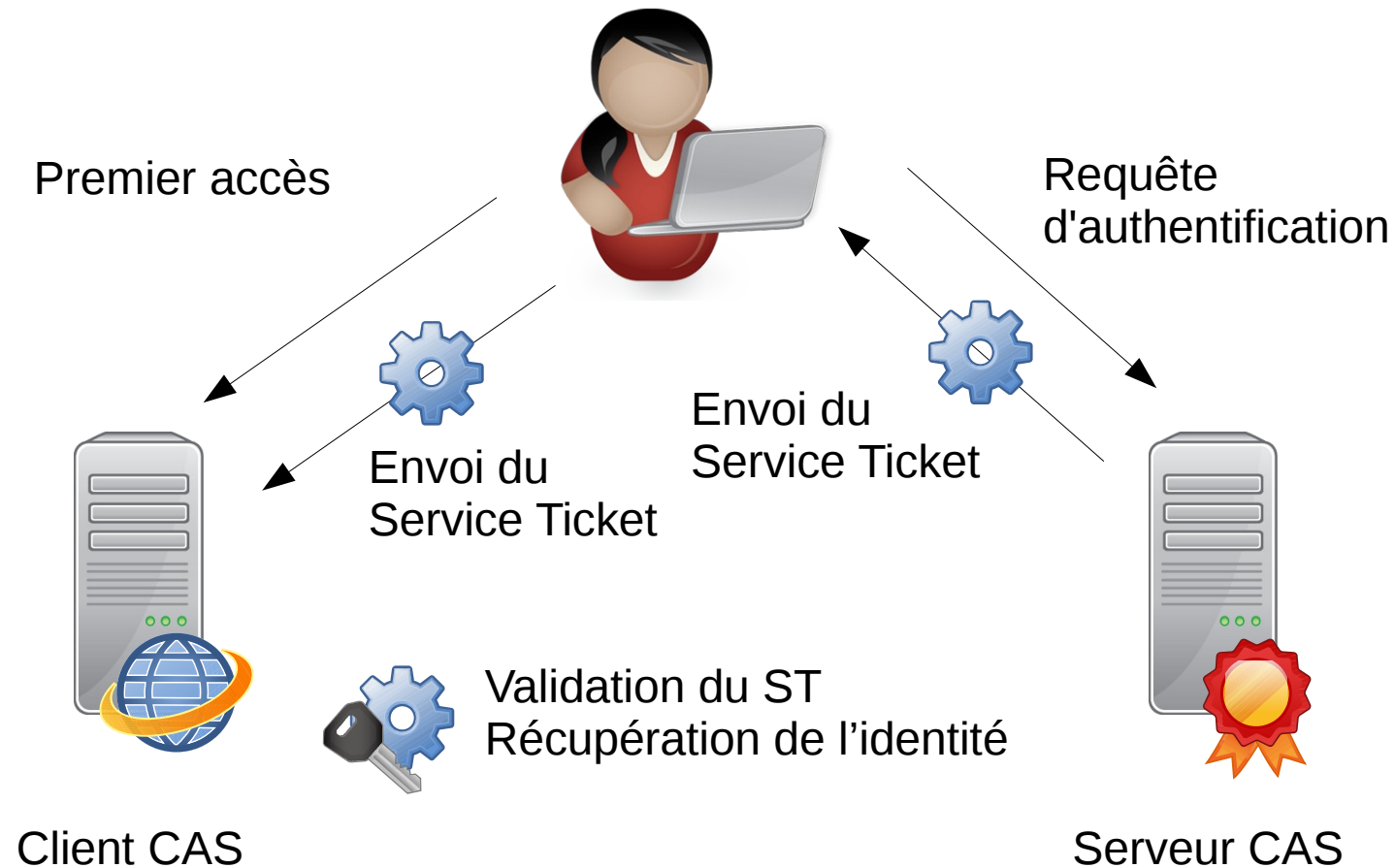
Protection d'un Web Service par agent





Protocoles CAS, SAML et OpenID Connect

Le protocole CAS



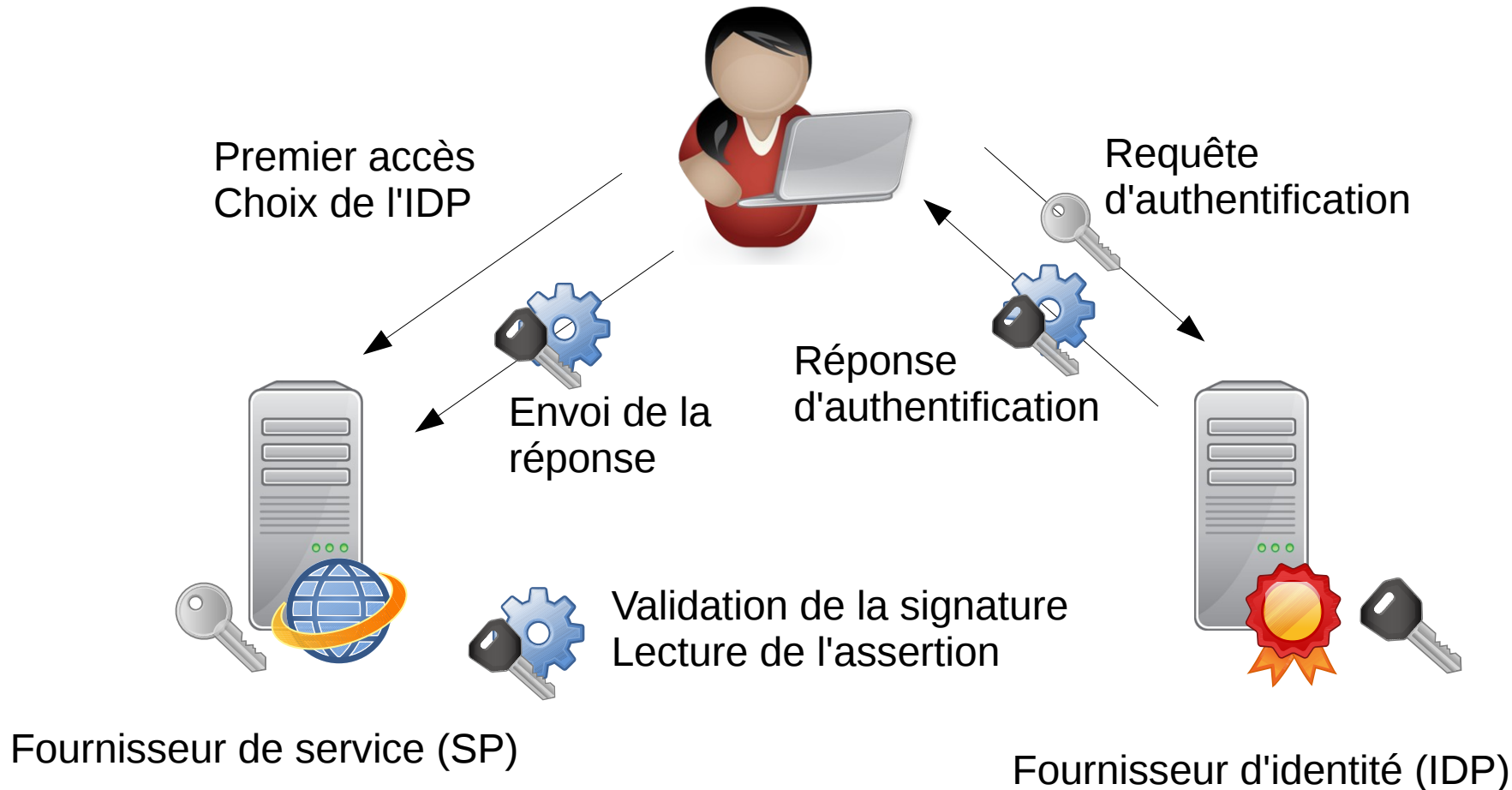
- **Requête validation ticket CAS :**

```
https://auth.example.com/cas/serviceValidate?
service=http://appli.example.com/&ticket=ST-
6096f5d3ddb33df6fd79529e2d626a6d
```

- **Réponse validation ticket CAS :**

```
<cas:serviceResponse
xmlns:cas='http://www.yale.edu/tp/cas'>
  <cas:authenticationSuccess>
    <cas:user>coudot</cas:user>
  </cas:authenticationSuccess>
</cas:serviceResponse>
```

Le protocole SAML

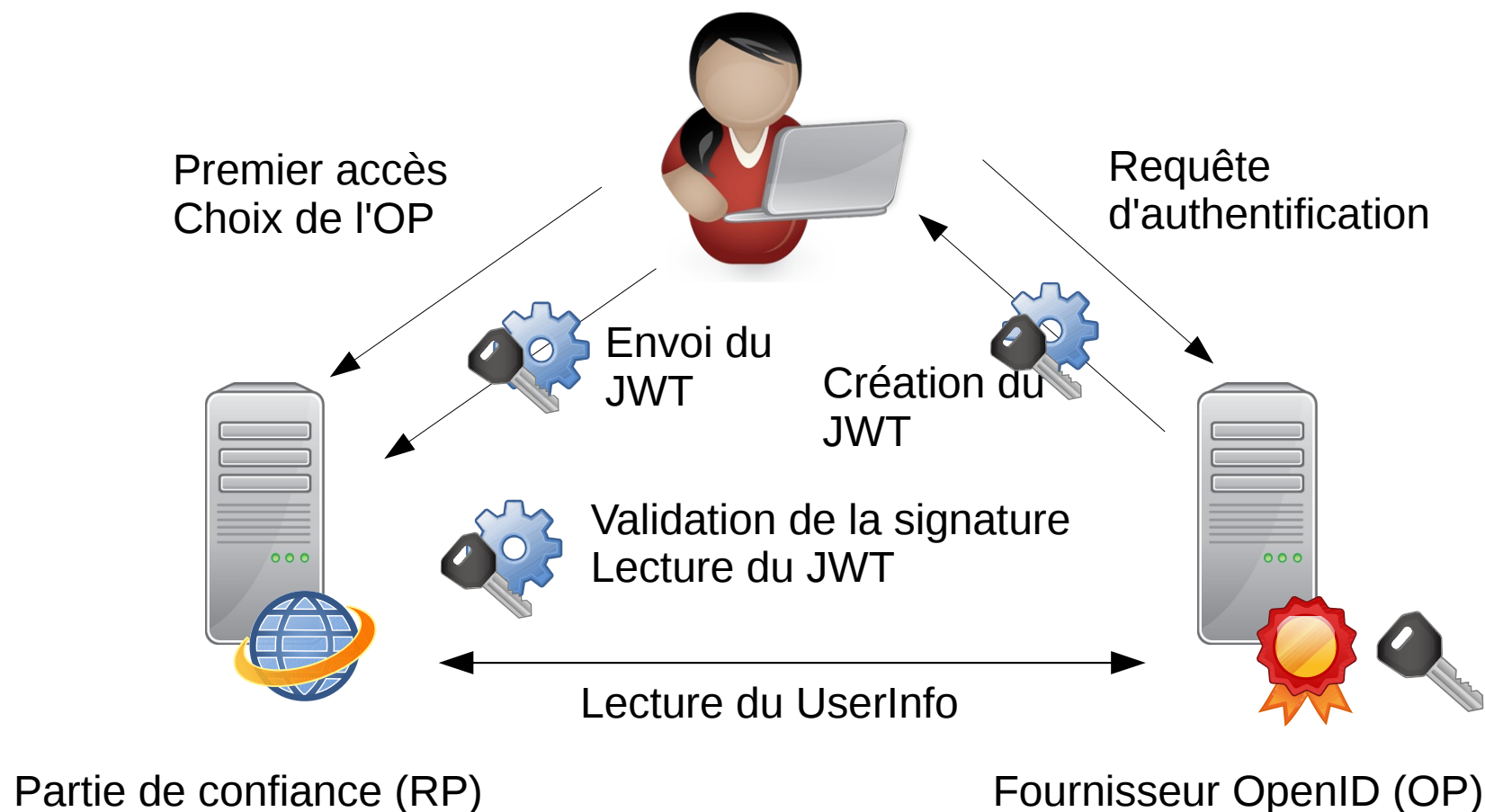


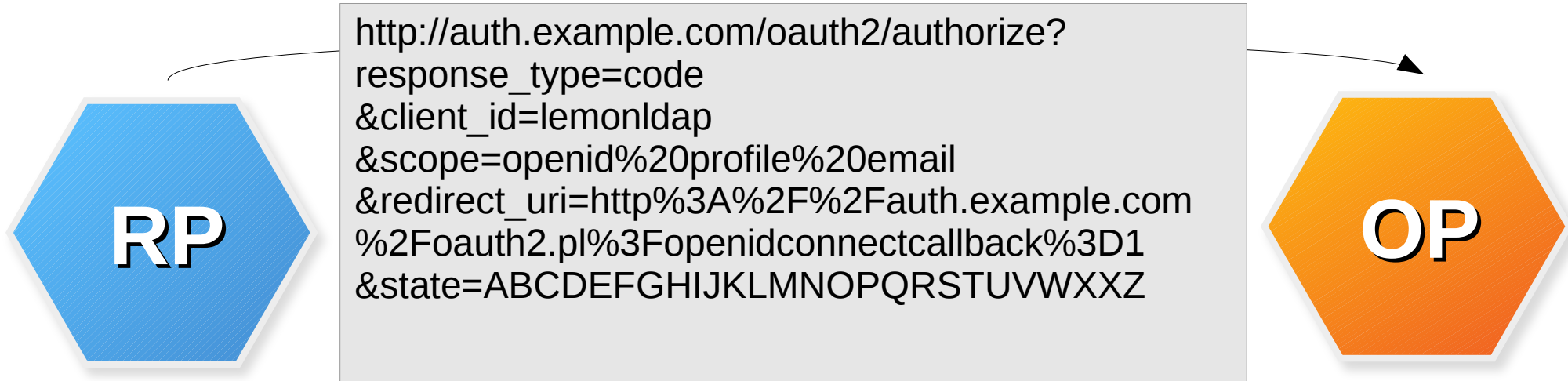
Requête SAML

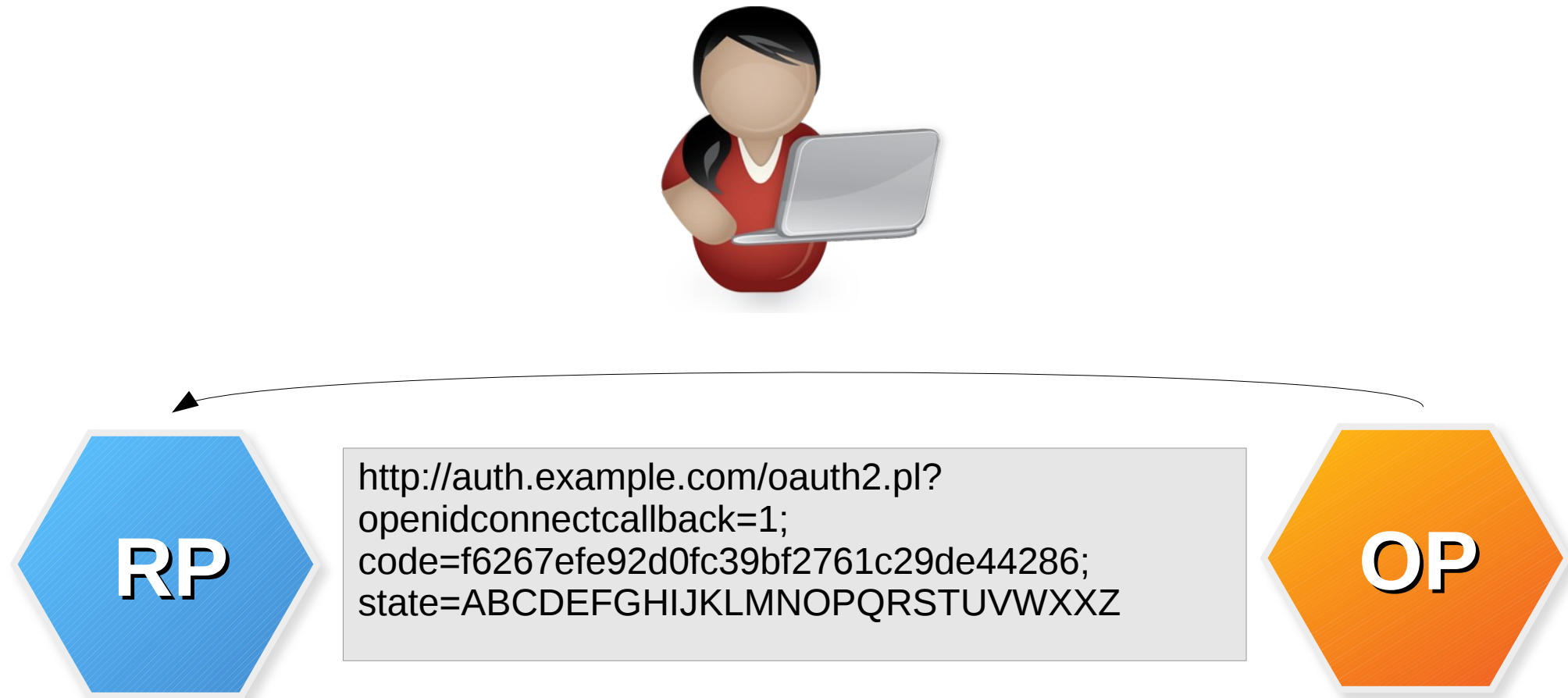
```
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" ID="ONELOGIN_809707f0030a5d00620c9d9df97f627afe9dcc24"
Version="2.0" ProviderName="SP test" IssueInstant="2014-07-16T23:52:45Z"
Destination="http://idp.example.com/SSOService.php"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

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol" xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" ID="_8e8dc5f69a98cc4c1ff3427e5ce34606fd672f91e6"
Version="2.0" IssueInstant="2014-07-17T01:01:48Z" Destination="http://sp.example.com/demo1/index.php?acs" InResponseTo="ONELOGIN_4fee3b046395c4e751011e97f8900b5273d56685">
  <saml:Issuer>http://idp.example.com/metadata.php</saml:Issuer>
  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
  </samlp:Status>
  <saml:Assertion xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xs="http://www.w3.org/2001/XMLSchema" ID="_d71a3a8e9fcc45c9e9d248ef7049393fc8f04e5f75"
Version="2.0" IssueInstant="2014-07-17T01:01:48Z">
    <saml:Issuer>http://idp.example.com/metadata.php</saml:Issuer>
    <saml:Subject>
      <saml:NameID SPNameQualifier="http://sp.example.com/demo1/metadata.php" Format="urn:oasis:names:tc:SAML:2.0:nameid-
format:transient">_ce3d2948b4cf20146dee0a0b3dd6f69b6cf86f62d7</saml:NameID>
      <saml:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
        <saml:SubjectConfirmationData NotOnOrAfter="2024-01-18T06:21:48Z" Recipient="http://sp.example.com/demo1/index.php?acs"
InResponseTo="ONELOGIN_4fee3b046395c4e751011e97f8900b5273d56685"/>
      </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Conditions NotBefore="2014-07-17T01:01:18Z" NotOnOrAfter="2024-01-18T06:21:48Z">
      <saml:AudienceRestriction>
        <saml:Audience>http://sp.example.com/demo1/metadata.php</saml:Audience>
      </saml:AudienceRestriction>
    </saml:Conditions>
    <saml:AuthnStatement AuthnInstant="2014-07-17T01:01:48Z" SessionNotOnOrAfter="2024-07-17T09:01:48Z" SessionIndex="_be9967abd904ddcae3c0eb4189adbe3f71e327cf93">
      <saml:AuthnContext>
        <saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:Password</saml:AuthnContextClassRef>
      </saml:AuthnContext>
    </saml:AuthnStatement>
    <saml:AttributeStatement>
      <saml:Attribute Name="uid" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">test</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="mail" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">test@example.com</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="eduPersonAffiliation" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">users</saml:AttributeValue>
        <saml:AttributeValue xsi:type="xs:string">examplerole1</saml:AttributeValue>
      </saml:Attribute>
    </saml:AttributeStatement>
  </saml:Assertion>
</samlp:Response>
```

Le protocole OpenID Connect







```
POST /oauth2/token HTTP/1.1
Host: auth.example.com
Authorization: Basic xxxx
Content-Type: application/x-www-form-urlencoded

grant_type=authorization_code
&code=f6267efe92d0fc39bf2761c29de44286
&redirect_uri=http%3A%2F%2Fauth.example.com
%2Foauth2.pl%3Fopenidconnectcallback%3D1
```



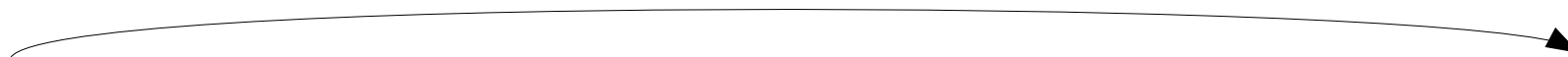


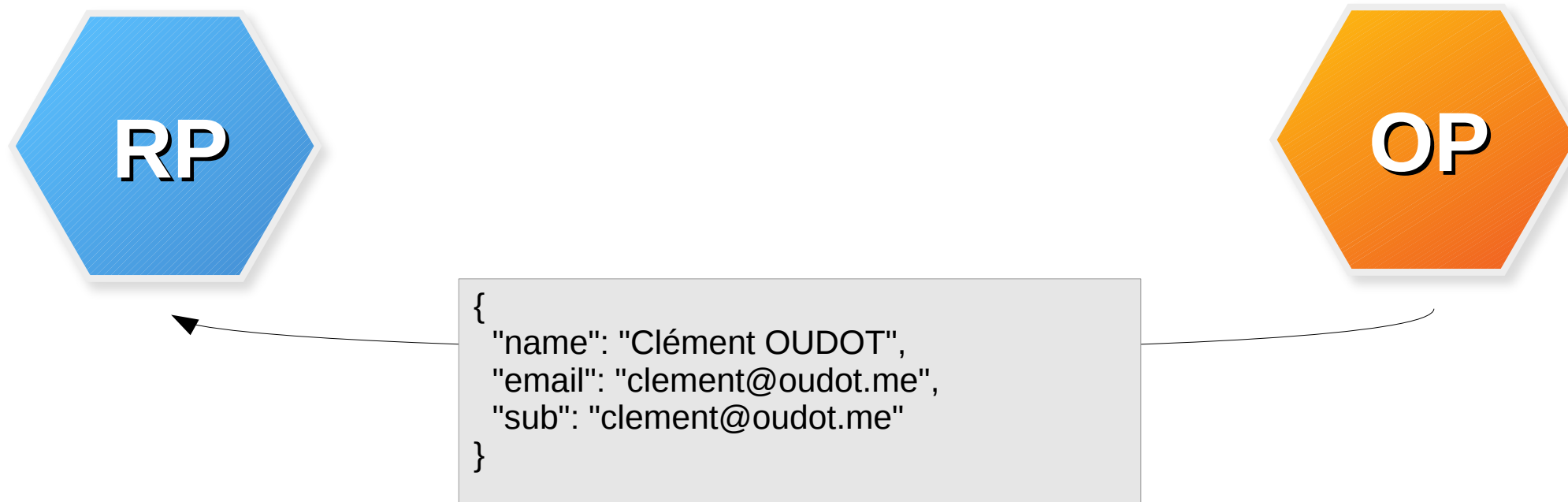
```
{
  "token_type" : "Bearer",
  "id_token" :
  "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJhY3liOiJsb2
  EtMlslmF1dGhfdGltZSI6MTQzMjExMzU5MywiaWF0IjoxN
  DMyMTEzOTY2LCJhdF9oYXNoIjoiaWF4IjoiOjEwIiwiaWF0Ijoi
  XpXZWZLc002QSI6Imh0dHA6Ly9hdXRoLmV4YW
  [ ... ] ofn0mg",
  "expires_in" : "3600",
  "access_token" :
  "512cdb7b97e073d0656ac9684cc715fe"
}
```



```
{
  "acr": "loa-2",
  "auth_time": 1432113593,
  "iat": 1432113966,
  "at_hash": "9axzsNi9pNDk5zWefKsM6A",
  "iss": "http://auth.example.com/",
  "exp": "3600",
  "azp": "lemonldap",
  "nonce": "1234567890",
  "sub": "clement@oudot.me",
  "aud": [
    "lemonldap"
  ]
}
```

```
POST /oauth2/userinfo HTTP/1.1  
Host: auth.example.com  
Authorization: Bearer 512cdb7b97e073d0656ac9684cc715fe  
Content-Type: application/x-www-form-urlencoded
```







Stratégies Cloud / On Premise



Bien cadrer son projet de SSO

- Identifier les utilisateurs devant accéder aux applications : comptes internes, partenaires, clients ?
- Définir son modèle de droits d'accès
- Lister les applications et leurs compatibilités avec les protocoles SSO
- Distinguer les applications Cloud et On Premise

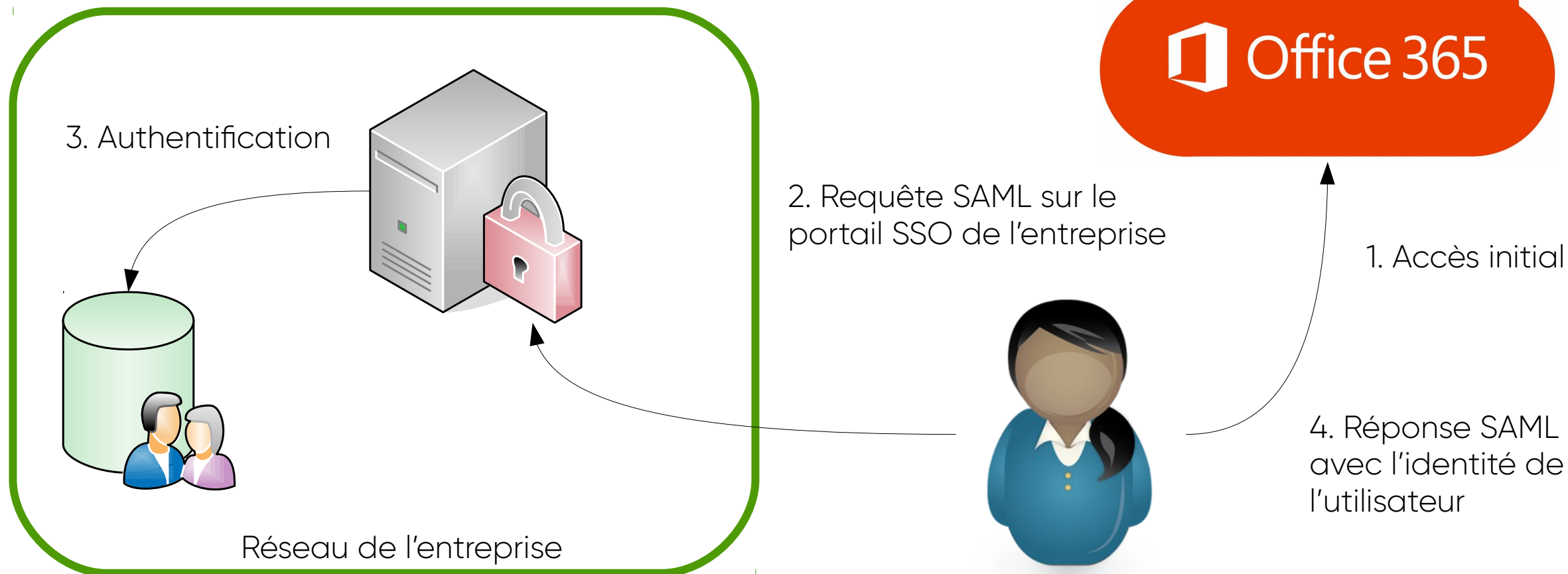
On Premise

- Applications modernes et « legacy »
- Réseau de confiance
- Tous les types de connexion SSO peuvent être mis en place :
 - Protection par agent (en-têtes HTTP)
 - CAS
 - SAML
 - OpenID Connect

Cloud (SaaS)

- Applications modernes
- Pas de réseau de confiance
- Privilégier les protocoles récents et sécurisés :
 - SAML
 - OpenID Connect

Exemple : Office 365





Merci pour votre attention

IDENTITY DAYS

24 octobre 2019 -
PARIS

 @IdentityDays
#identitydays2019

**Merci à tous nos
partenaires !**

