eleven-labs.com

Authentification d'un utilisateur par certificat X509

Thierry Thuon

25/06/2015

Eleven labs

Problématique

Comment authentifier un utilisateur sans utiliser un identifiant et un mot de passe ?

Eleven labs

Sommaire

- Chiffrement asymétrique
- Certificat X.509
- Configuration des clefs et du serveur
- Authentification de l'utilisateur par Symfony

Eleven labs

Sommaire

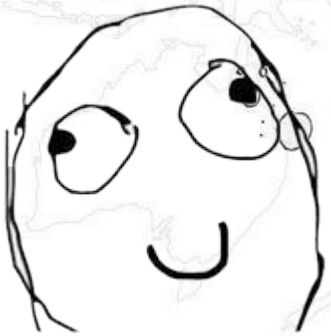
- Chiffrement asymétrique
- Certificat X.509
- Configuration des clefs et du serveur
- Authentification de l'utilisateur par Symfony

Eleven labs

Derpina & Derp

- Derpina et Derp veulent communiquer

Derpina

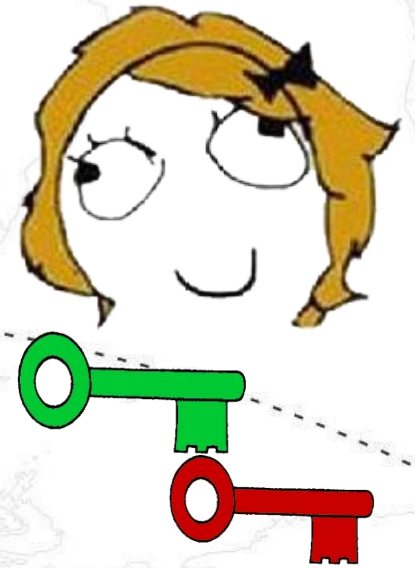


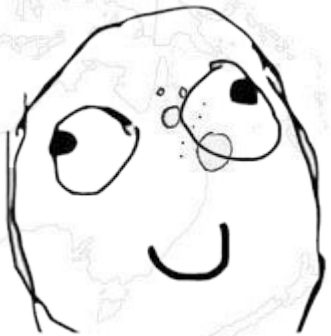
Derp

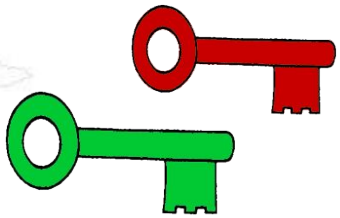
Eleven labs

Génération des clefs

- Chacun génère une paire de clef : privée et public



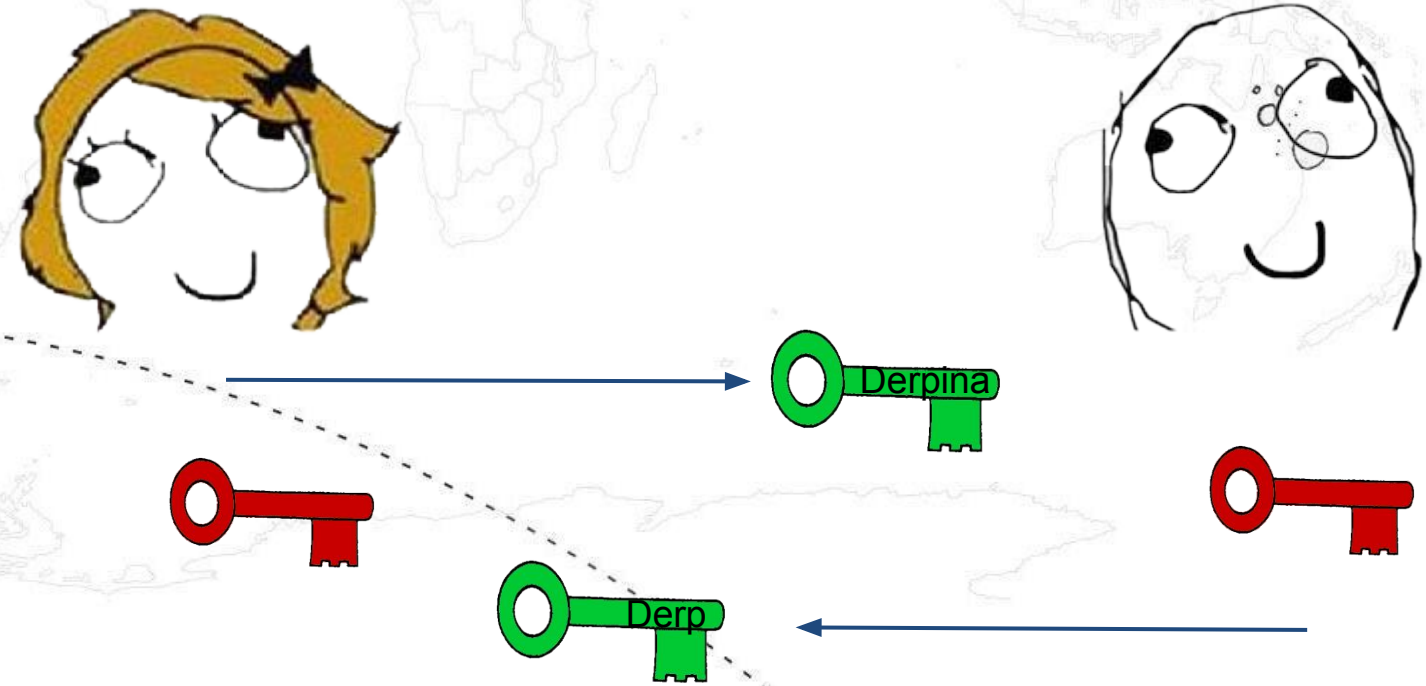




Eleven labs

Échange des clefs

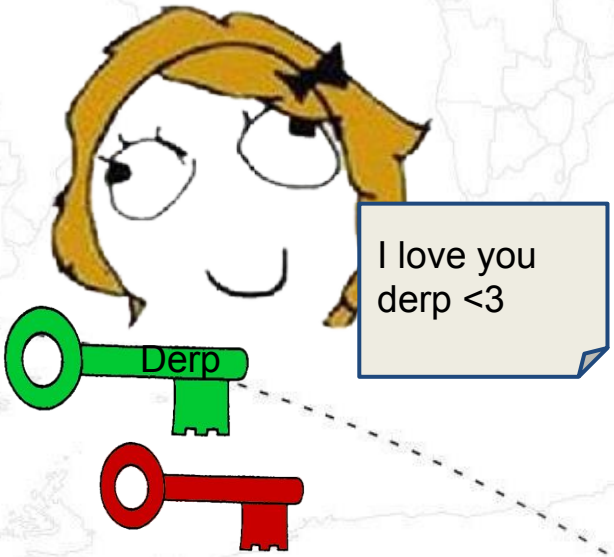
- Ils s'échangent leurs clef public.

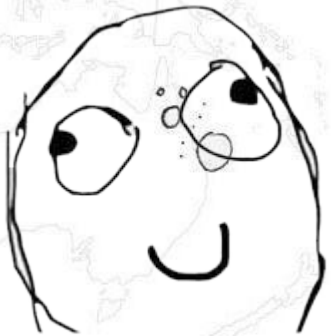


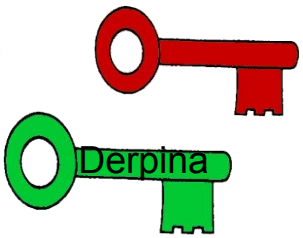
Eleven labs

Le message

- Derpina écrit un message à Derp



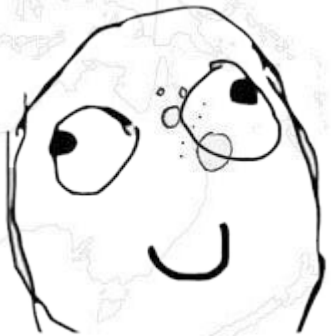


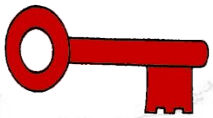


Eleven labs

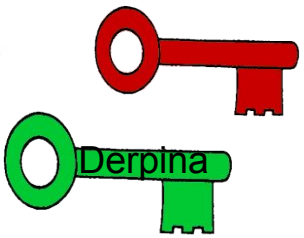
Chiffrement du message

- Le message est chiffré avec le clef public de Derp





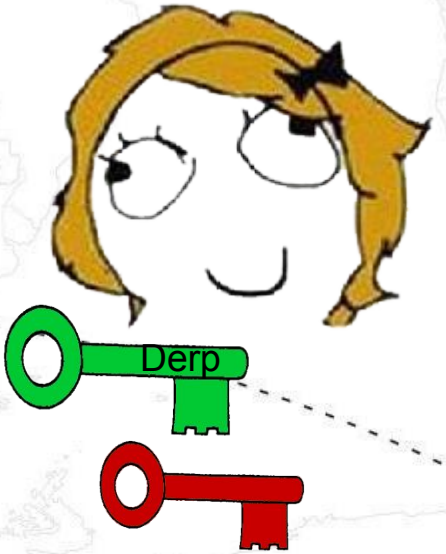
sqa31fag”f:
/fae%Age%
1Fea$p^èg

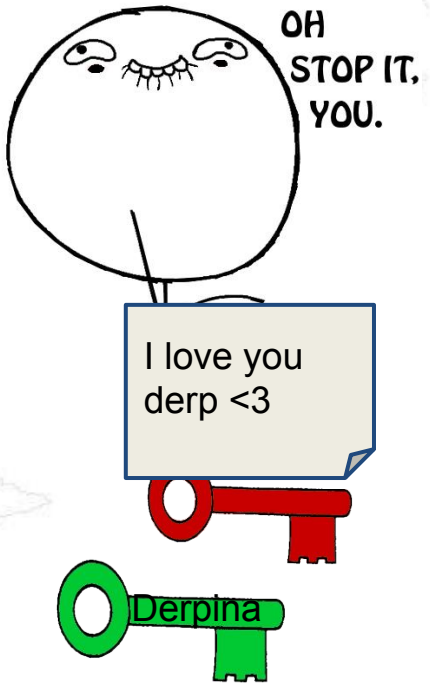


Eleven labs

Déchiffrement du message

- Le message est déchiffré avec le clef de privée de Derp





Eleven labs

Sommaire

- Chiffrement asymétrique
- Certificat X.509
- Configuration des clefs et du serveur
- Authentification de l'utilisateur par Symfony

Eleven labs

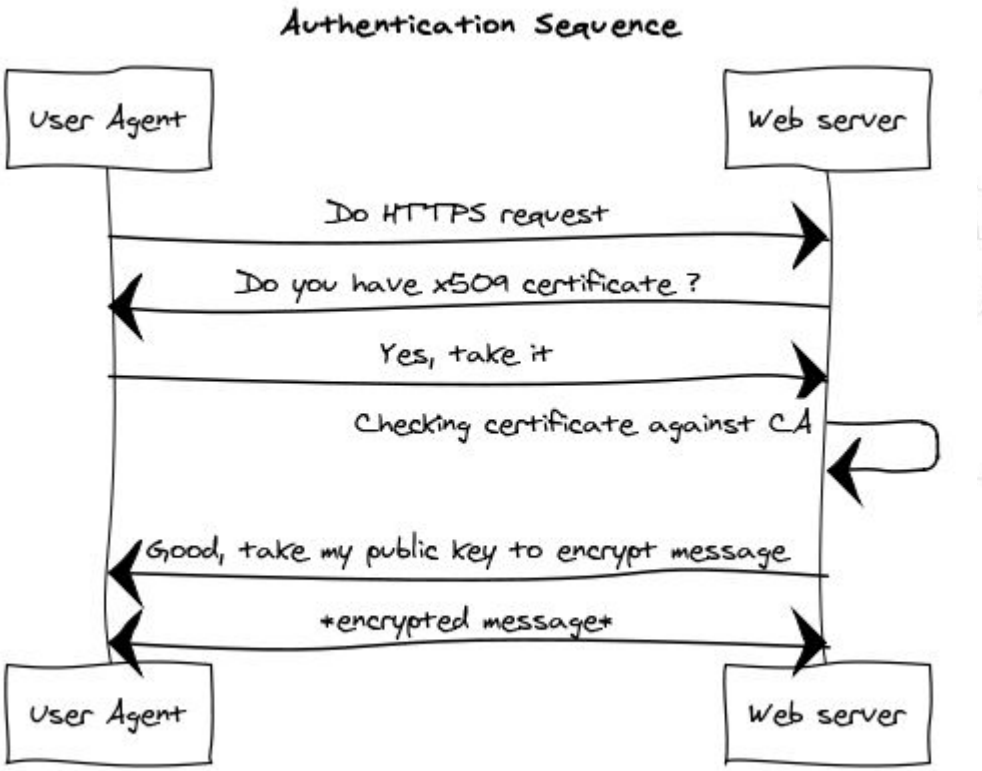
x509, késako?

Norme de cryptographie pour les infrastructures de clefs publique : https://tools.ietf.org/html/rfc5280

- Autorité de certification
- Clefs publiques / privées

Eleven labs

Vue d'ensemble



Eleven labs

Sommaire

- Chiffrement asymétrique
- Certificat X.509
- Configuration des clefs et du serveur
- Authentification de l'utilisateur par Symfony

Eleven labs

Création de l'autorité de certification

```
openssl genrsa -out ca.key 1024

openssl req -new -key ca.key -out ca.csr

openssl x509 -req -days 365 -in ca.csr -
 signkey ca.key -out ca.crt
```

Eleven labs

Création du certificat web

```
openssl genrsa -out server.key 1024
openssl req -new -key server.key -out server.
 csr
openssl x509 -req -days 365 -in server.csr -
 signkey server.key -out server.crt
```

Eleven labs

Création du certificat client

```
openssl genrsa -out client.key 1024

openssl req -new -key client.key -out client.
 csr
```

Eleven labs

Signature par l'autorité de certification

```
openssl x509 -req -days 365 -CA ca.crt -CAkey
 ca.key -CAcreateserial -in client.csr -out
 client.crt
```

Eleven labs

Export du certificat client

```
openssl pkcs12 -export -clcerts -in client.crt
 -inkey client.key -out client.p12
```

Eleven labs

Configuration apache2

```
<VirtualHost *:443>
  ServerName local.piaf.eu
  DocumentRoot /home/lepiaf/training/web

  SSLEngine On
  SSLCertificateFile /home/lepiaf/ssl/server.crt
  SSLCertificateKeyFile /home/lepiaf/ssl/server.key
  SSLCACertificateFile /home/lepiaf/ssl/ca.crt
</VirtualHost>
```

Eleven labs

Configuration apache2

```
<Directory /home/lepiaf/training/web>
  SSLOptions +StdEnvVars
  AllowOverride All
  Require all granted
  Allow from All
</Directory>

<Location /admin>
  SSLVerifyClient require
</Location>
```

https://httpd.apache.org/docs/2.2/mod/mod_ssl.html#envvars
https://httpd.apache.org/docs/2.2/mod/mod_ssl.html#sslverifyclient

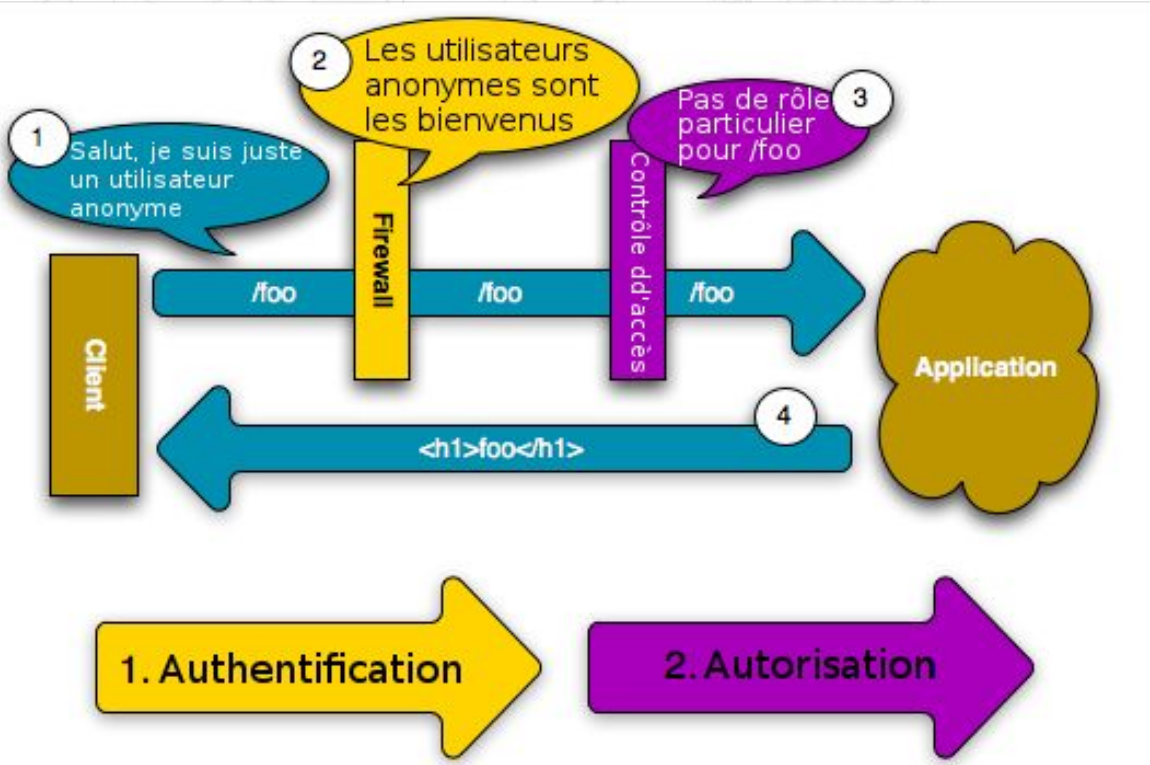
Eleven labs

Sommaire

- Chiffrement asymétrique
- Certificat X.509
- Configuration des clefs et du serveur
- Authentification de l'utilisateur par Symfony

Eleven labs

Rappel sur la sécurité dans Symfony



Eleven labs

Configuration du pare-feu

```
security:
    providers:
          user_provider:
              id: lepiaf_app.security.user_provider
    firewalls:
        secured_area:
            pattern: ^/admin
            x509:
                provider: user_provider
                user: SSL_CLIENT_S_DN_Email
                credential: SSL_CLIENT_S_DN
```

Symfony\Bundle\SecurityBundle\DependencyInjection\Security\Factory\X509Factory

Eleven labs

Fournisseur d'utilisateur

Un service avec une classe qui étend

- Symfony\Component\Security\Core\User\UserProviderInterface

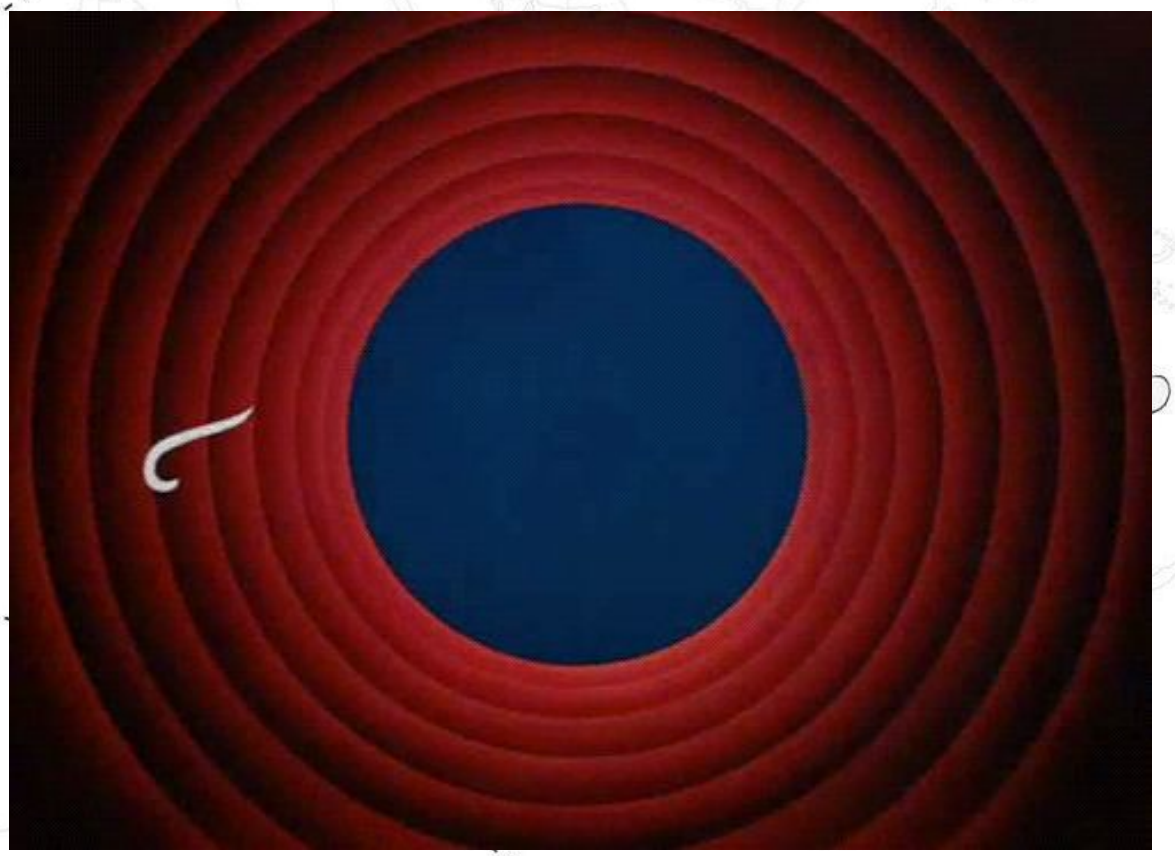
Doit implémenter

- loadUserByUsername
- refreshUser
- supportsClass

In-Memory peut être utilisé

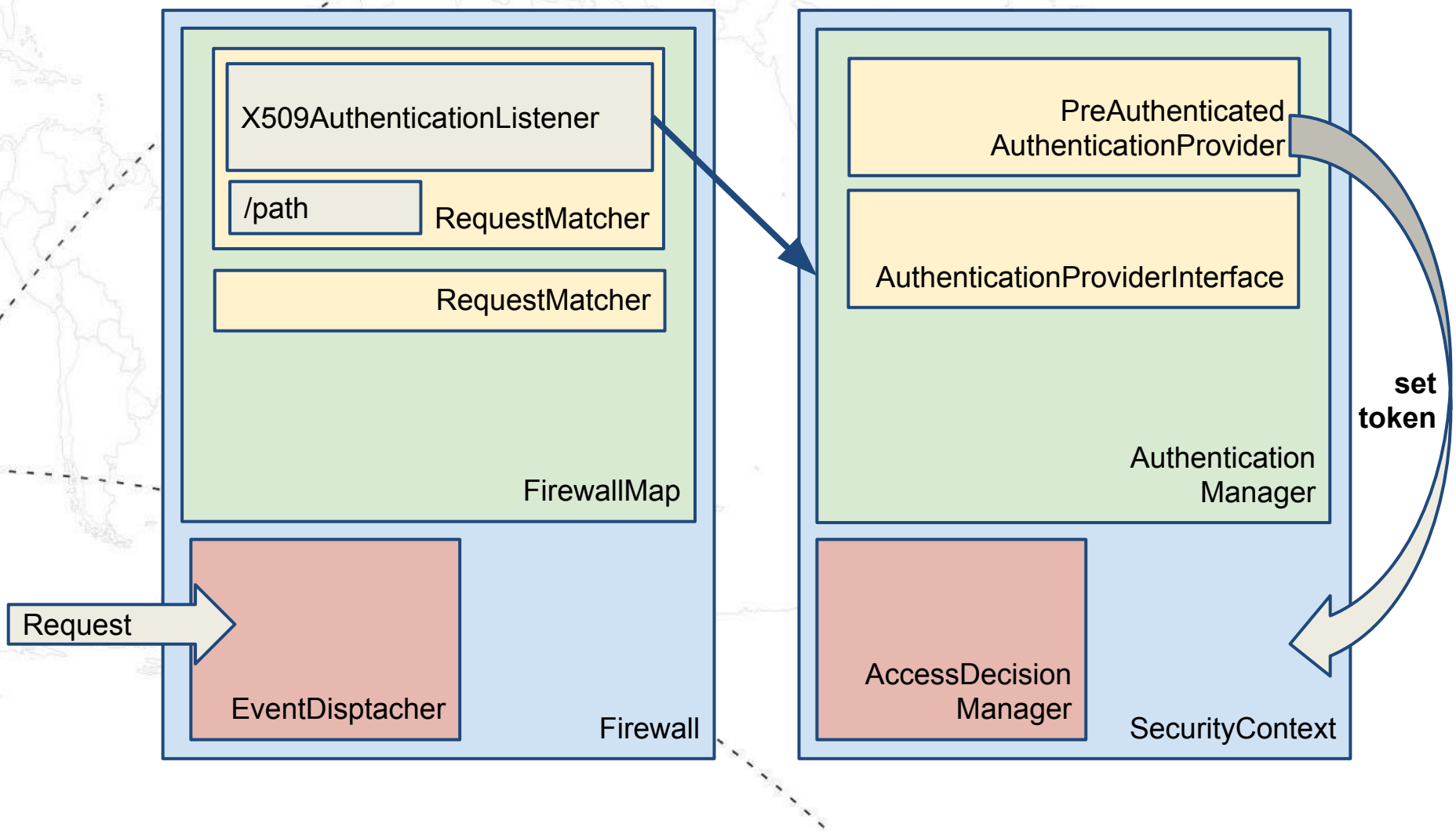
Eleven labs

That’s it !



Eleven labs

Vue globale



Eleven labs

Le pare-feu

Listener sur kernel.request :

- Symfony\Component\Security\Http\Firewall

Mapping URL

- Symfony\Component\Security\Http\Firewall\X509AuthenticationListener

http://symfony.com/doc/2.3/components/security/firewall.html#a-firewall-for-http-requests

Eleven labs

Contexte de sécurité

Le contexte de sécurité a des gestionnaire d'authentification

- Symfony\Component\Security\Core\Authentication\Provider\PreAuthenticatedAuthenticationProvider

Création du token

- Symfony\Component\Security\Core\Authentication\Token\PreAuthenticatedToken

http://symfony.com/doc/2.3/components/security/authentication.html#the-authentication-manager
http://symfony.com/doc/2.3/components/security/authentication.html#authentication-providers

Démo

Démonstration

https://local.piaf.eu/app_dev.php

https://local.piaf.eu/app_dev.php/admin

Eleven labs

Ressources

https://github.com/lepiaf/symfony-ssl-auth

http://symfony.com/doc/current/cookbook/security/pre_authenticated.html

http://symfony.com/doc/current/cookbook/security/entity_provider.html

http://symfony.com/doc/current/cookbook/security/custom_provider.html

http://www.impetus.us/~rjmooney/projects/misc/clientcertauth.html

https://stormpath.com/blog/what-x509-certificate/