Google Apps integration in Oracle Apex

for Application Express 4.2 and 5
(using Oracle XE 11.2)

a presentation by

4apex
SMART

20. hroug
godišnja konferencija

Who is Smart4apex

Reliable Apex solutions by shared knowledge

Who is Smart4apex

- independent contractors in a cooperative, (zadruga in Croatian)
- present (kscope US, oracle open world US, apex-world NL, Usergroups (throughout Europe)
- share knowledge
- do projects
- use each others network
    - not only for new assignments
    - also to help customers in time of a full schedule

Who is Smart4apex

Arthur Kolkman

Sergei Martens (driving force/founder)

Learco Brizzi

Toine Tuerlings

Roel Hartman (chairman/founder)

Dick Dral

Christian Rokitta (secretary/founder)

Richard Martens (treasury/founder)

Who am I?

Richard Martens

- Independant Consultant since 2012
- smart4apex founding member (2010)
- Oracle since 2002 (Oracle 8i)
- PL/SQL, Apex, HTML(5), CSS(3), Javascript, XML, XSLT
- Special interest in UI
- RIMA on Oracle Forums
- Trainer at skillbuilders.com

Agenda

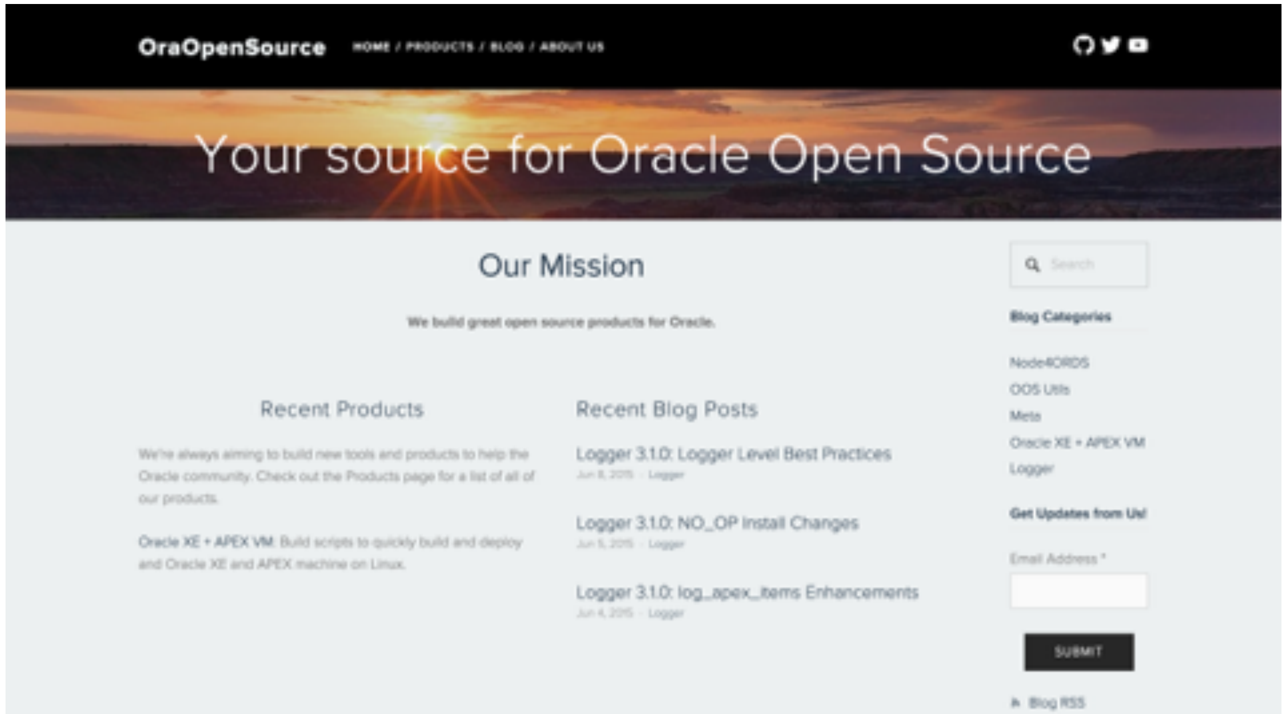
1. Mission
2. Building blocks
3. Authentication
    a. login flow
    b. prerequisites
    c. authentication plugin
    d. scopes and code
4. Calendar
    a. prerequisites
    b. scopes and code
5. Contacts
    a. prerequisites
    b. scopes and code
6. Drive
    a. prerequisites
    b. scopes and code
7. Email
    a. prerequisites
    b. scopes and code

Presentation available on Google Docs:

https://goo.gl/rVXo3e

Code will be made available on:

http://www.oraopensource.com



OraOpenSource
HOME / PRODUCTS / BLOG / ABOUT US
Your source for Oracle Open Source
Search
SUBMIT
Blog RSS

Our Mission

We build great open source products for Oracle.

Recent Products

We're always aiming to build new tools and products to help the Oracle community. Check out the Products page for a list of all of our products.

Oracle XE + APEX VM: Build scripts to quickly build and deploy and Oracle XE and APEX machine on Linux.

Recent Blog Posts

Logger 3.1.0: Logger Level Best Practices
Jun 8, 2015 · Logger

Logger 3.1.0: NO_OP Install Changes
Jun 5, 2015 · Logger

Logger 3.1.0: log_apex_items Enhancements
Jun 4, 2015 · Logger

Blog Categories

Node4ORDS
OOS Utils
Meta
Oracle XE + APEX VM
Logger

Get Updates from Us!

Email Address *

Why Google Apps ?

- Access to services from anywhere, at anytime
  A key benefit of the Google-hosted solution is that we can access email, contacts, and calendar from any computer or mobile device with an Internet connection, from anywhere in the world

- Innovative solutions
  We can leverage the ongoing creative and technical solutions of the Google Apps platform to provide employees with powerful, easy-to-use tools for getting their work done

- Highly scalable environment
  With Google Apps, our email capacity will grow automatically as our organization grows, and we'll avoid the complexity of internal systems

- More collaboration features
  With Google's next-generation applications, we can collaborate with colleagues, customers, and partners more easily and efficiently than ever before

- Instant messaging
  Because Google Apps includes Google Talk, we can now implement an instant messaging system for our organization

Mission

Create building blocks for an Apex application that can read, insert and update in Google Apps:

- Email
- Calendars
- Events
- Contacts
- Documents
- . . .

Google Apps for Work

Google for Education

Google for Nonprofits

Building blocks

Apex Authentication plugin

- based upon PL/SQL package
- does not use any tables

PL/SQL packages for

- Authentication
- Calendar and Events
- Contacts
- Documents
- Email

Authentication-Code is available in GitHub

https://github.com/smart4solutions/apex_oauth

Building blocks should

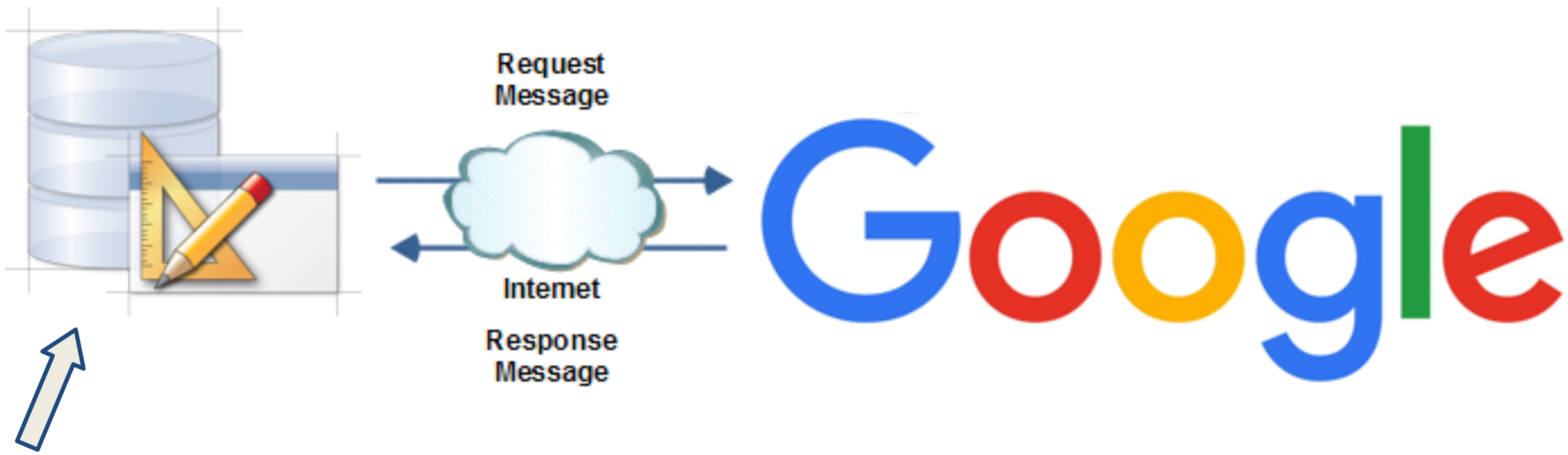
- not rely on any tables
- easy to use for programmers in their applications
- self-documenting

Challenges

- Google provides Java client but Oracle XE does not support Java
- Google uses json extensively as response format
  - Oracle 11 and Apex 4.x have no tools to read json
  - Use of json packages by Jonas Krogsboell (https://github.com/pljson/pljson)
- All Google communication use HTTPS, creating a wallet is out of scope for this presentation, but there is a trick!

A bit on web services

- A web service is just like a regular web-page
- The result of the webpage is often <XML> or {JSON} but can have any content
- Restful and SOAP-XML
  - http://blog.smartbear.com/apis/understanding-soap-and-rest-basics



Our Oracle / Apex server

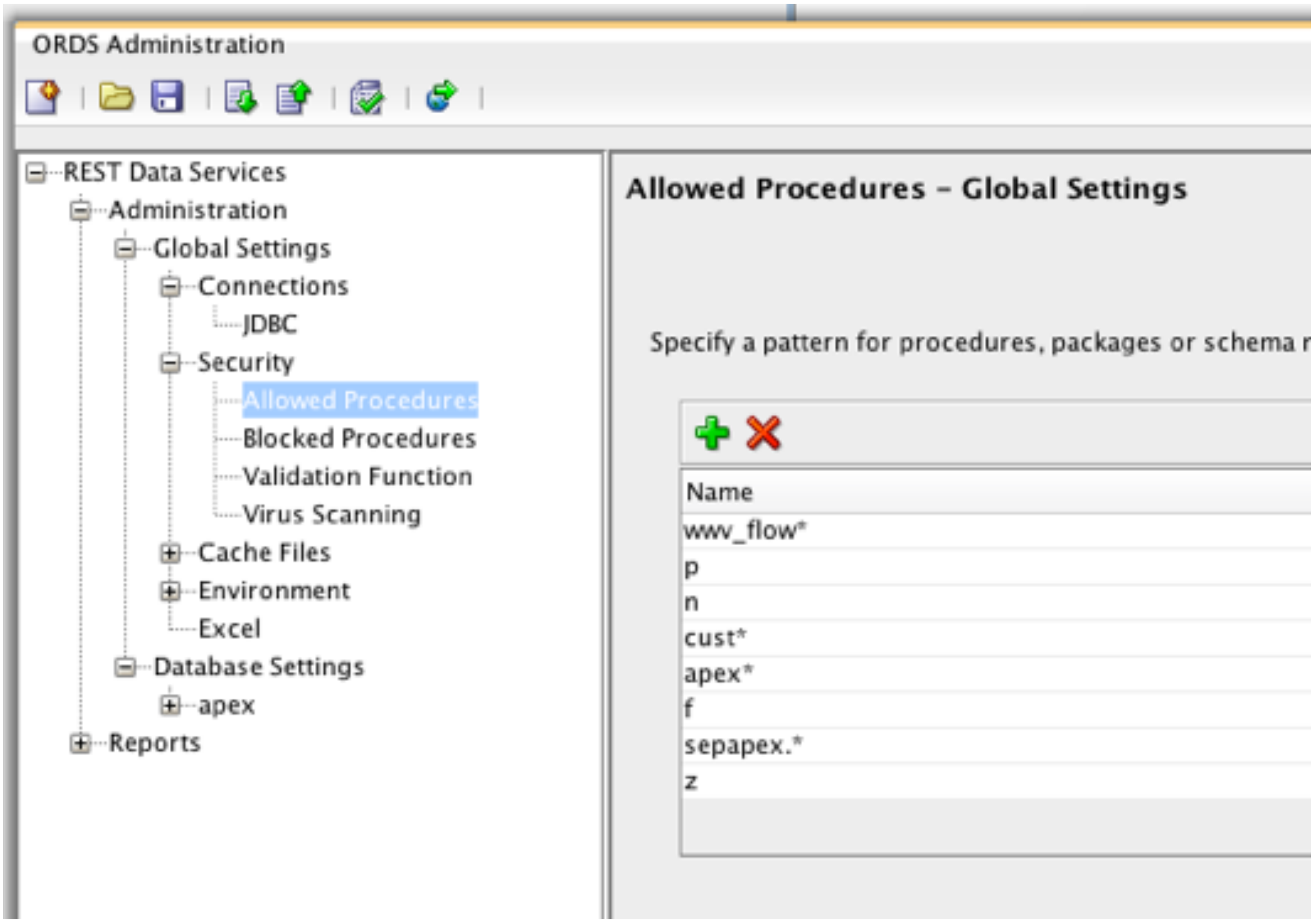
Authentication (oAUTH2) prerequisites

You must

1. apply for a google developer account
2. use an apex-listener

   either APEX or ORDS
3. enable "Allowed procedures"

   in the listener administration
4. Also allow the apex procedures
   a. *f, n, p, z*
   b. *cust**
   c. *wwv_flow**
   d. *apex**
   e. *<schema>.**



ORDS Administration

- REST Data Services
  - Administration
    - Global Settings
      - Connections
        - JDBC
      - Security
        - Allowed Procedures
        - Blocked Procedures
        - Validation Function
        - Virus Scanning
      - Cache Files
      - Environment
      - Excel
    - Database Settings
      - apex
  - Reports

Allowed Procedures – Global Settings

Specify a pattern for procedures, packages or schema r

Name
wwv_flow*
p
n
cust*
apex*
f
sepapex.*
z

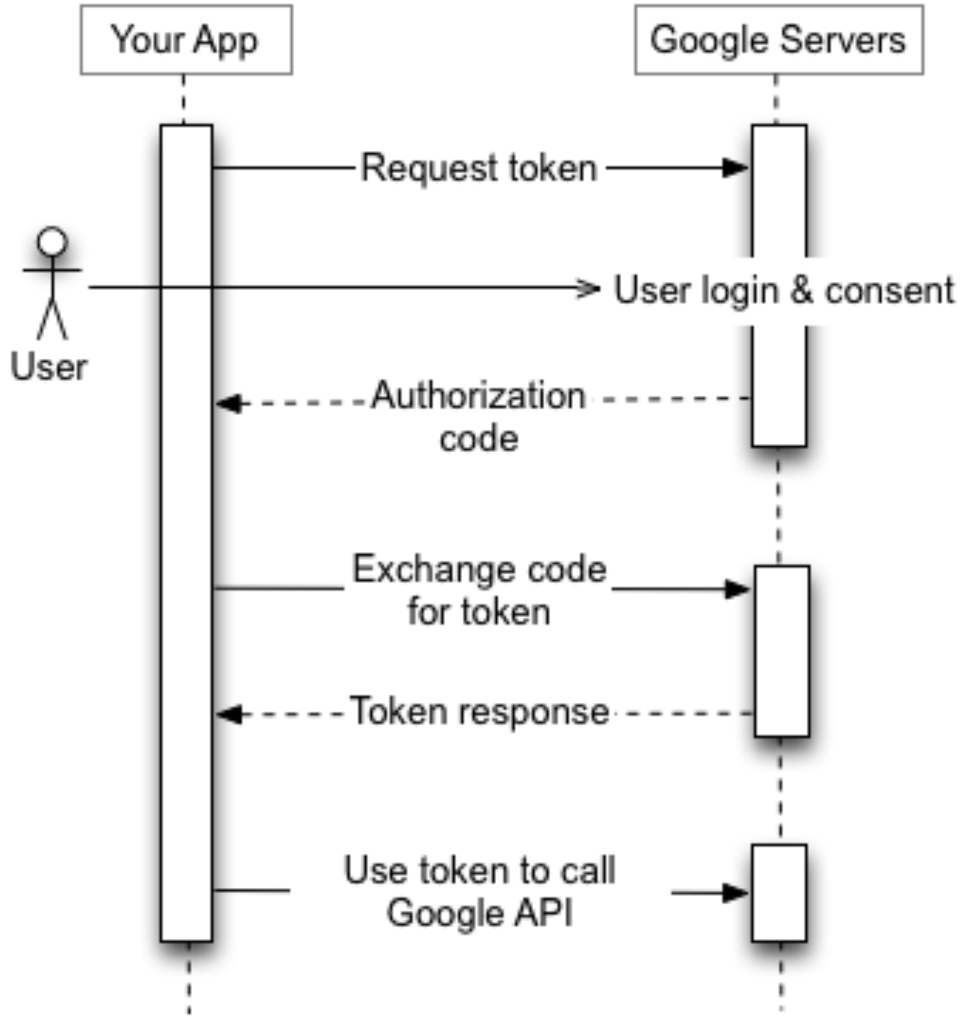
5. Don't forget to grant execute on the packages
   a. *grant execute on*

      <schema>.s4sa_oauth_pck to apex_public_user

https://console.developers.google.com

Authentication (oAUTH2)

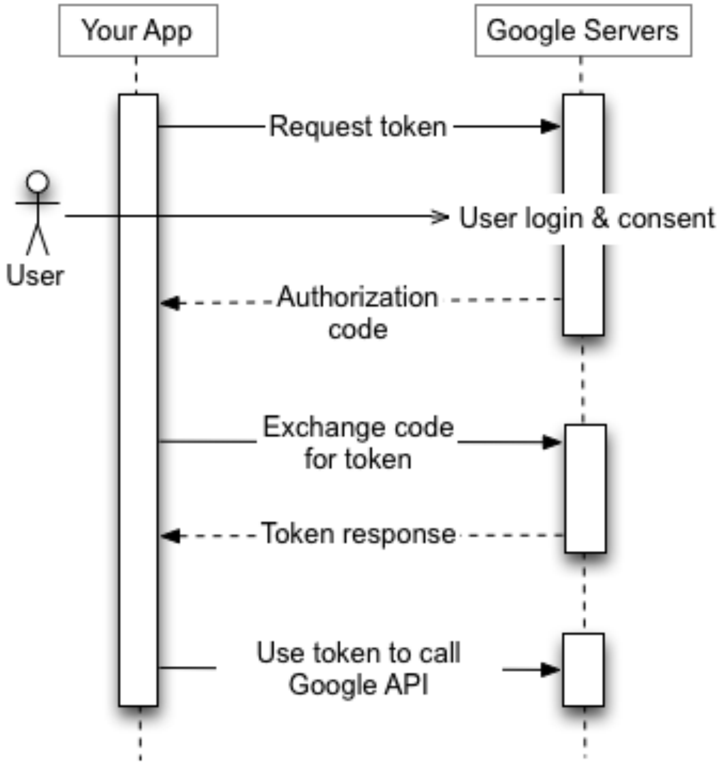
1. apex redirects end-user to google login-page
2. after successful login into google, google redirects the end-user back to a redirect URL on your server
   (this is a pl/sql stored procedure)
3. when the pl/sql procedure runs it:
   a. requests google for an exchange token (using RESTFUL web services)
   b. reads a “token” from the google response
   c. requests further info (email-address, name etc.)
   d. creates a session for the end-user
   e. stores the token in an application-item and in an apex-collection
   f. redirects the user to the home-page
4. apex is now equipped with a token to do further requests to the google API’s

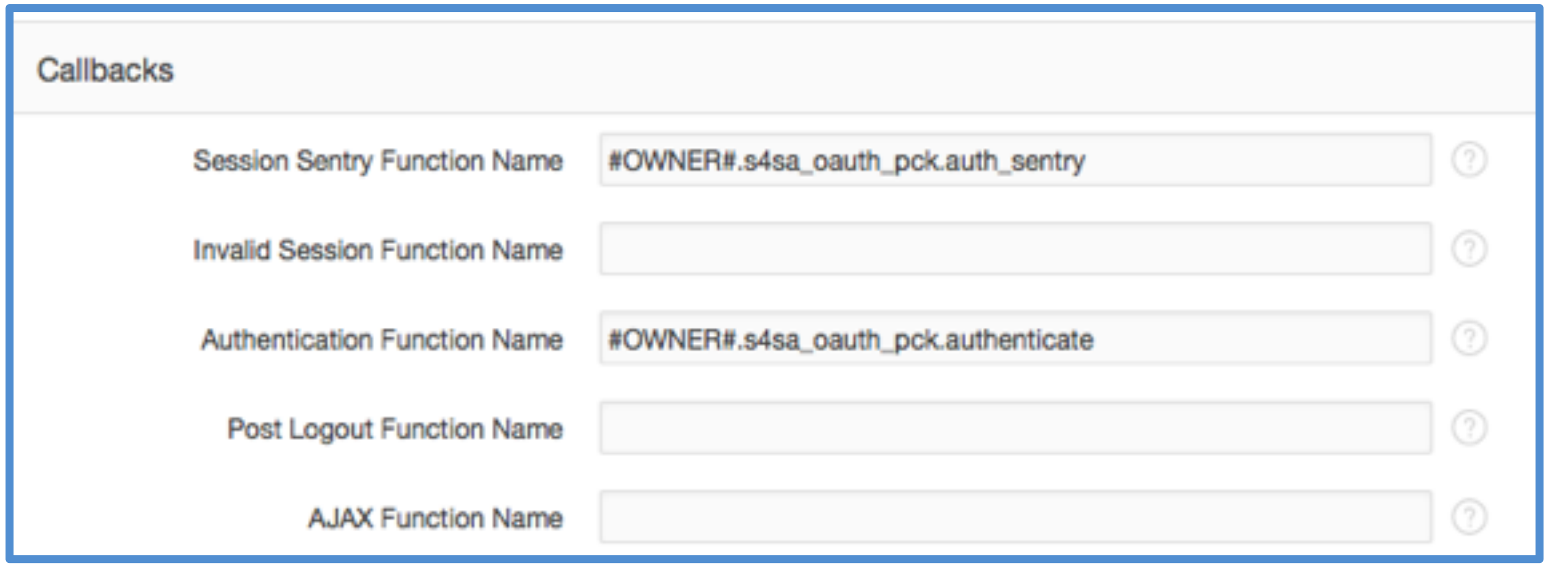


Authentication (oAUTH2)

Using the Apex Authentication plugin

1. import plugin before doing anything else
2. create a new authentication
   a. “Based on a pre-configured scheme from the gallery”
   b. Give name and choose “S4S oAuth2” plugin
3. Result:





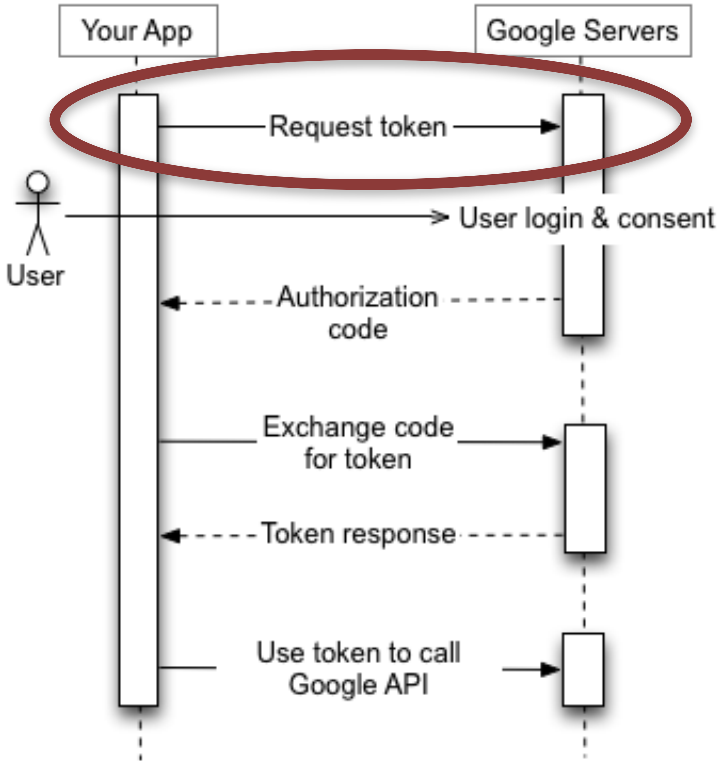
Callbacks

Session Sentry Function Name	#OWNER#.s4sa_oauth_pck.auth_sentry
Invalid Session Function Name	
Authentication Function Name	#OWNER#.s4sa_oauth_pck.authenticate
Post Logout Function Name	
AJAX Function Name	

demo: authentication plugin
demo: application authentication settings

Authentication (oAUTH2)

```
function google_authentication
  ( p_authentication in apex_plugin.t_authentication
  , p_plugin         in apex_plugin.t_plugin
  , p_password       in varchar2
  ) return apex_plugin.t_authentication_auth_result
  is
    t_retval        apex_plugin.t_authentication_auth_result;
    t_seconds_left  number;
    cursor c_oauth_user
    is     select c.n001 - ((sysdate - c.d001) * 24 * 60 * 60) as seconds_left
           from   apex_collections c
           where  c.collection_name = s4sg_util_pck.g_collname;
  begin

    open c_oauth_user;
    fetch c_oauth_user into t_seconds_left;
    close c_oauth_user;

    t_retval.is_authenticated := nvl(t_seconds_left, 0) > 0;

    if not t_retval.is_authenticated then
      s4sg_auth_pck.redirect_oauth2
        ( p_scope          => s4sg_util_pck.g_scope
        , p_gotopage       => v('APP_PAGE_ID')
        , p_force_approval => s4sg_util_pck.g_force_approval
        , p_ggl_extras     => s4sg_util_pck.g_ggl_extras
        );
    end if;

    return t_retval;

  end google_authentication;
```



scope	description
profile	basic login scope
plus-login	access to social features
email	email-address etc
see: https://developers.google.com/+/api/oauth#login-scopes	

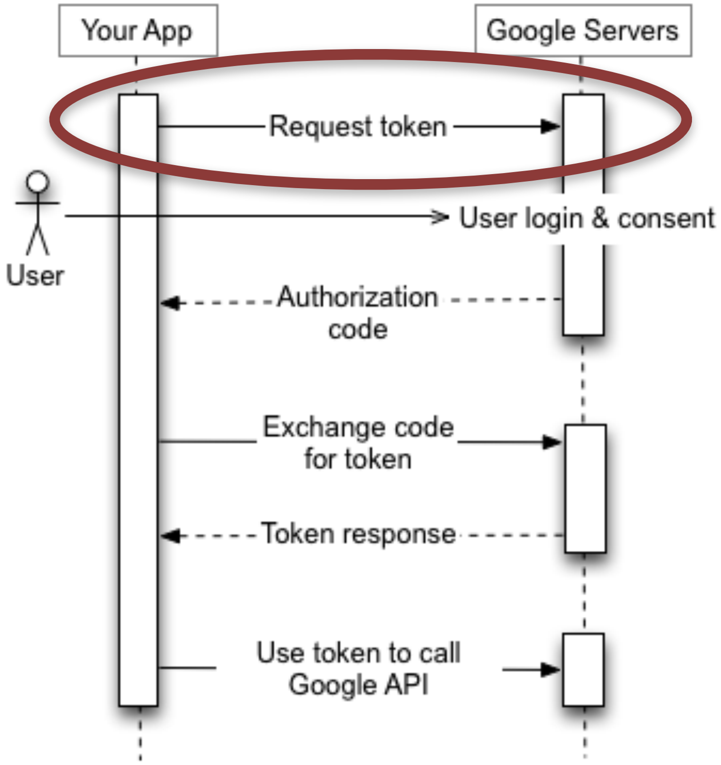
Authentication - scope (oAUTH2)

Scope

URI

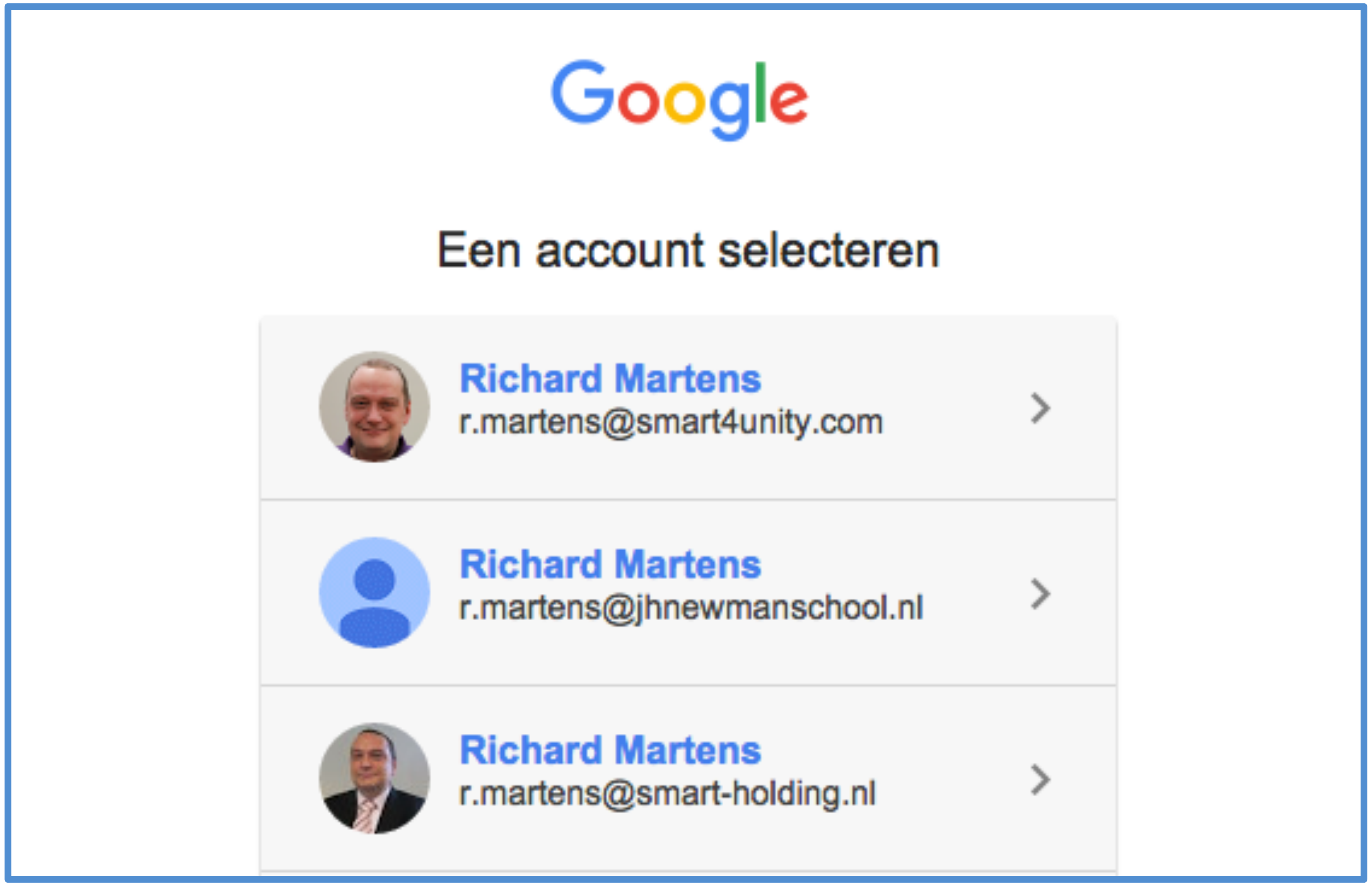
State

```
procedure redirec
  ( p_gotopage                  har2 default null
  )
  is
    t_url     varchar2(32767);
  begin

    t_url := 'https://accounts.google.com/o/oauth2/auth?client_id=' || g_provider.client_id
                        || '&redirect_uri='   || g_provider.redirect_uri
                        || '&scope='          || apex_util.url_encode(g_provider.scope)
                        || '&state='          || v('APP_SESSION') || ':' || v('WORKSPACE_ID')
                                              || ':' || v('APP_ID') || ':' || p_gotopage
                        || '&response_type=' || 'code'  -- mandatory for google
                        || g_provider.extras
                        || case g_provider.force_approval
                             when 'Y' then '&approval_prompt=force'
                             else ''
                           end
                        ;

    owa_util.redirect_url ( t_url );

    apex_application.stop_apex_engine;

  end redirect_oauth2;
```



scope	description
profile	basic login scope
plus-login	access to social features
email	email-address etc
see: https://developers.google.com/+/api/oauth#login-scopes	

Authentication (oAUTH2)



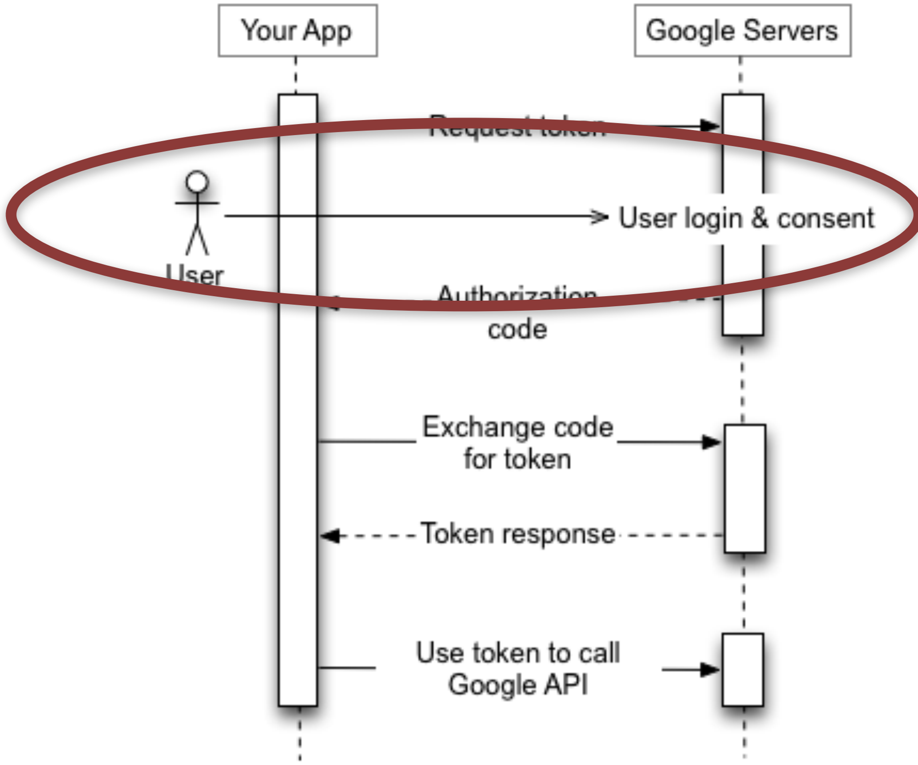
Google

Een account selecteren

Richard Martens
r.martens@smart4unity.com

Richard Martens
r.martens@jhnewmanschool.nl

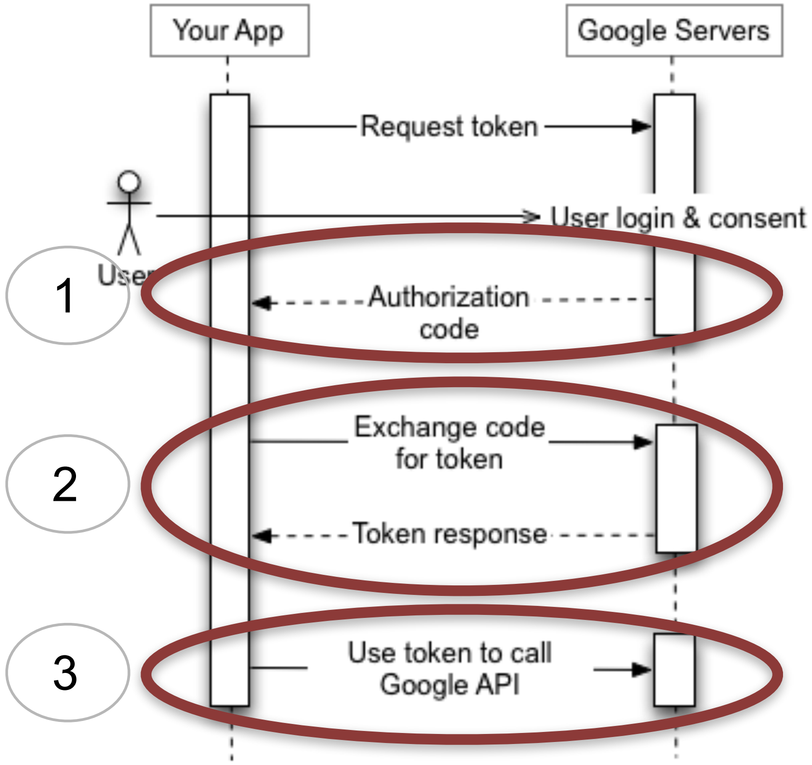
Richard Martens
r.martens@smart-holding.nl



demo

Authentication (oAUTH2)

```
procedure oauth2callback
  ( state             in varchar2 default null
  , code              in varchar2 default null
  , error             in varchar2 default null
  , error_description in varchar2 default null
  , token             in varchar2 default null
  )
  is -- pseudo code
  begin

    -- 1 check for error provided by google in querystring
    -- 1 put querystring in variables (using string_to_table)

    -- 2 get token using authorization code

    -- 3 get user-info using google provided token

    -- create session without login
    -- create or truncate collection
    -- populate collection using user-info
    -- perform login (APEX_CUSTOM_AUTH.LOGIN)

  end oauth2callback;
```



demo

Google Apps for Work

Contacts

Contacts consist of:

- contactlist
- contact

Retrieving contacts

- maximum 25 contacts per request

TBD

- pagination
- saving contacts (delete, create, update)

More info

- https://developers.google.com/google-apps/contacts/v3/

scope	***description***
https://www.google.com/m8/feeds	***read/write***
https://www.googleapis.com/auth/contacts.readonly	***read-only***

Drive

Drive consist of:

- Retrieving folder contents
- maximum 99 contacts per request

TBD

- pagination
- saving documents

More info

- https://developers.google.com/drive/web/about-sdk

scope	description
https://www.googleapis.com/auth/drive	***read/write***
https://www.googleapis.com/auth/drive.readonly	***read-only***

Calendar

Calendar consists of

- calendarlist
- calendar
- eventlist
- event

TBD

- pagination

More info:

- https://developers.google.com/google-apps/calendar/

1. Get list of calendars
2. Let the user choose a calendar
3. Get list of events
4. Let the user insert an event

scope	*description*
https://www.googleapis.com/auth/calendar	***read/write***
https://www.googleapis.com/auth/calendar.readonly	***read-only***

Google Apps for Work

Email

Email consist of:

- Sending email
  - replacement of apex_mail.send
  - attachments not yet included

Email will not include a full client!

- we have gmail or “inbox” for that

TBD:

- getting mails using a query
- sending attachments

More info:

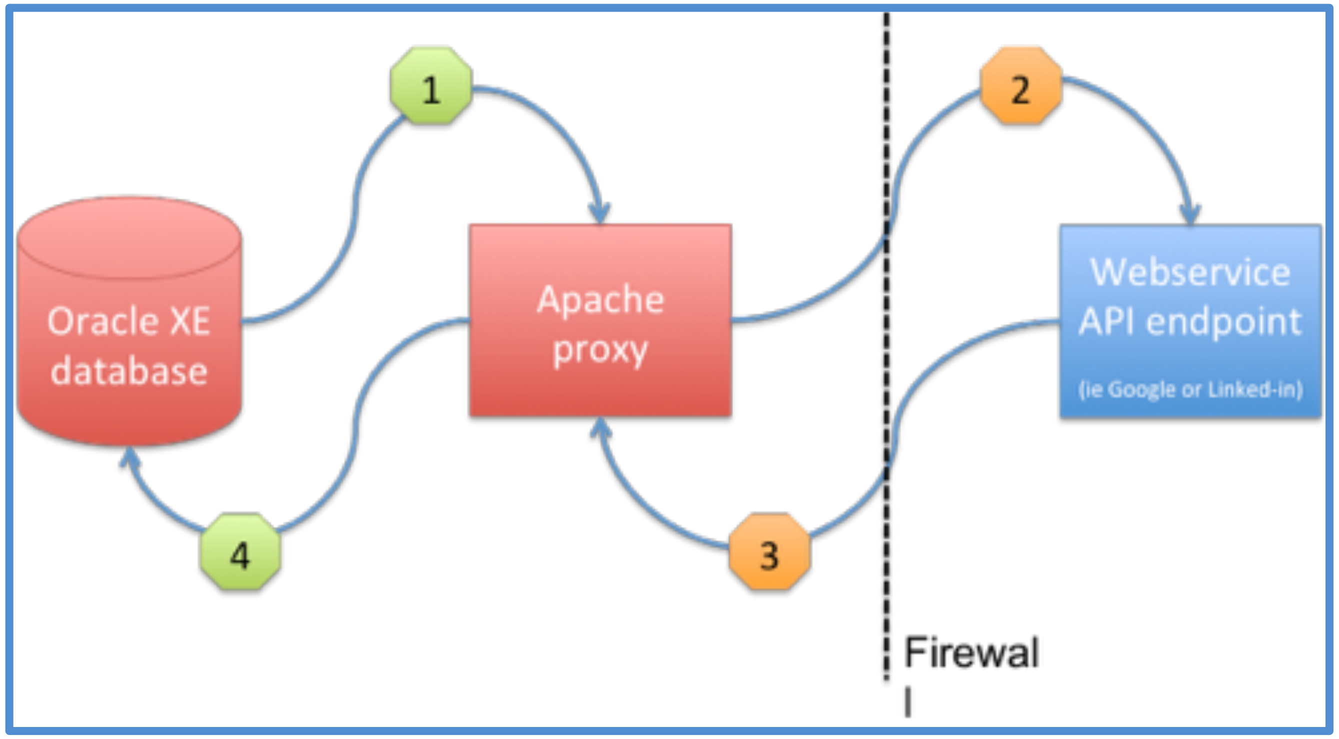
- https://developers.google.com/gmail/api/
- https://developers.google.com/gmail/api/guides/sending

scope	*description*
https://www.googleapis.com/auth/drive	***read/write***
https://www.googleapis.com/auth/drive.readonly	***read-only***

using a Reverse Proxy

Why do we need a reverse proxy?

- Oracle natively does not support root certificates
- We must tell Oracle what certificates to trust using a wallet
- XE does not have tools to create a wallet
- XE 11.2 has a bug that prevents use of SHA-2 signed certificates (ie linkedin) this is solved in Oracle 11.2.0.3
- Certificates have a validation end-date therefore are re-issued now and then



using a Reverse Proxy

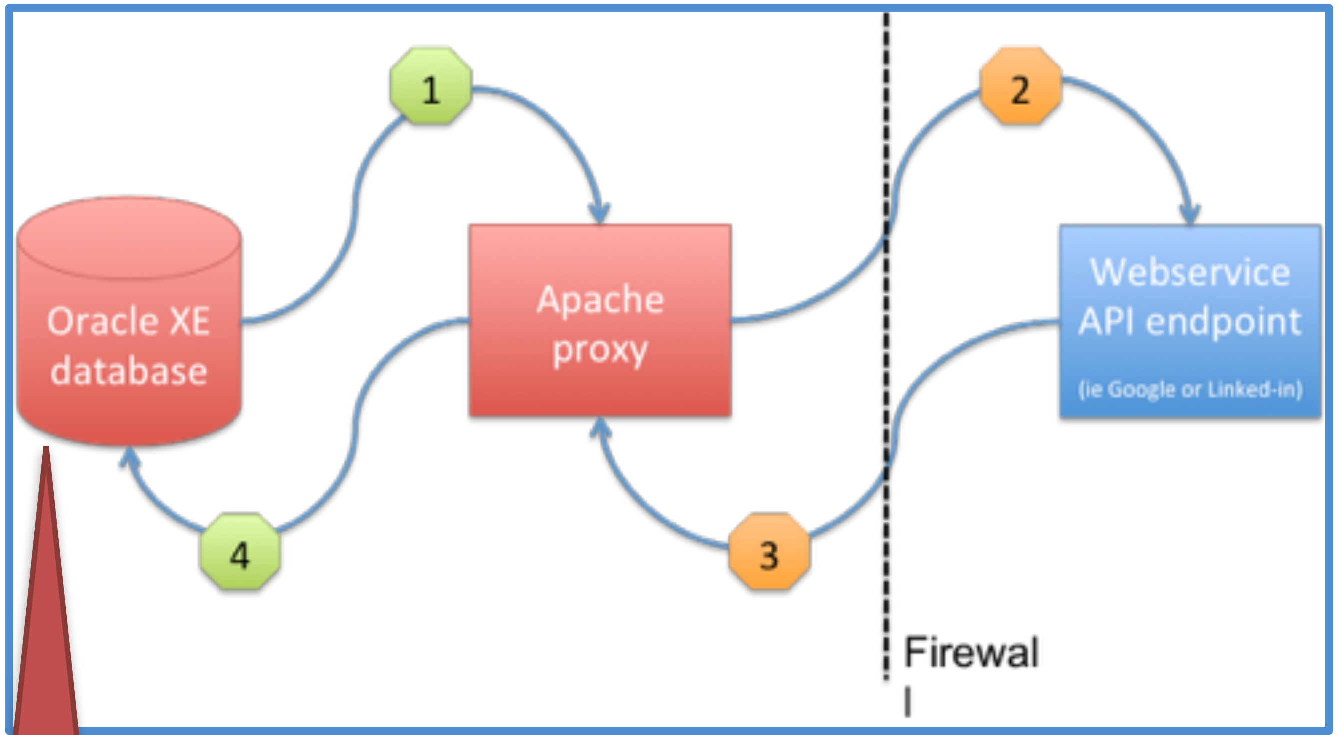
How does the reverse proxy work?

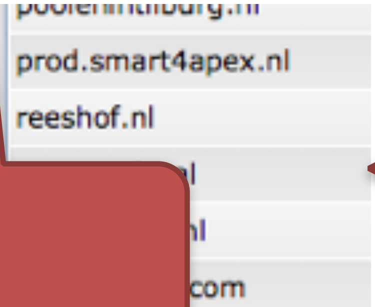
1. the request is made to a “local” apache proxy using pl/sql
2. apache forwards the request on our behalf
3. apache receives the result on our behalf
4. apache send the result to our pl/sql code

The only thing that changes in our code is the URI-endpoint (in that it now utilizes a prefix)

Advantages

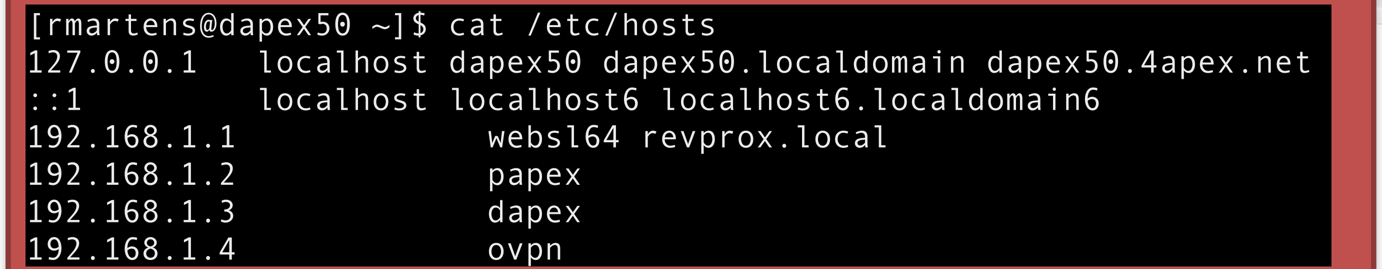
- no wallet to manage
- no effort in importing new certificates when re-issued
- minimal change in pl/sql code
- works for all https encrypted traffic





defined as a normal “domain” in our directadmin webconfigurator

our database server has an entry in its “/etc/hosts file



```
[rmartens@dapex50 ~]$ cat /etc/hosts
127.0.0.1   localhost dapex50 dapex50.localdomain dapex50.4apex.net
::1         localhost localhost6 localhost6.localdomain6
192.168.1.1             websl64 revprox.local
192.168.1.2             papex
192.168.1.3             dapex
192.168.1.4             ovpn
```

using a Reverse Proxy

```
RewriteEngine On
ProxyVia On

## proxy tbv linkedin
ProxyRequests Off
SSLProxyEngine On
<Proxy *>
  Order deny,allow
  Allow from all
</Proxy>
############
#  GOOGLE  #
############
ProxyPass /www.googleapis.com/ https://www.googleapis.com/
ProxyPassReverse /www.googleapis.com/ https://www.googleapis.com/
ProxyPass /apps-apis.google.com/ https://apps-apis.google.com/
ProxyPassReverse /apps-apis.google.com/ https://apps-apis.google.com/
ProxyPass /accounts.google.com/ https://accounts.google.com/
ProxyPassReverse /accounts.google.com/ https://accounts.google.com/
ProxyPass /www.google.com/ https://www.google.com/
ProxyPassReverse /www.google.com/ https://www.google.com/
```

Questions ?

Next year:

Office 365
Home Premium
A
O
N
P
W
P
X
SkyDrive
skype™
Complete Office.

20.
hroug
godišnja konferencija