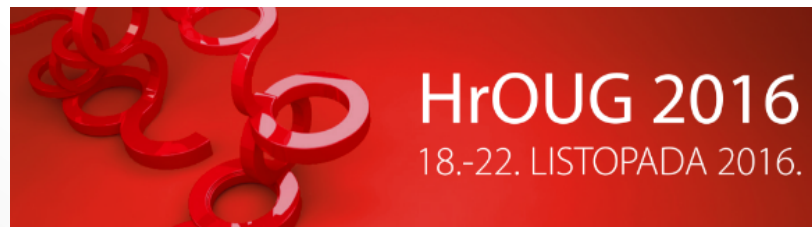


Secure your APEX application

Dimitri Gielis, APEX R&D
Aljaz Mali, Abakus Plus



Aljaž Mali

- IT Solutions Architect at Abakus plus, d.o.o
- SIOUG - Member of Executive Committee
- APEX Meetups organizer
- HTMLDB – just a toy?, SIOUG, Portorož 2004
- apex.world member of the month (march, 2016)
- APEX Text Messages
 - <http://www.oraopensource.com/blog/?category=APEX+Text+Messages>

Dimitri Gielis

- Founder & CEO of APEX R&D
- 17+ years of Oracle Experience (OCP & APEX Certified)
- Oracle ACE Director
- Presenter at Oracle Conferences (OOW, ODTUG, OGH, UKOUG, ...)
- <http://dgielis.blogspot.com>



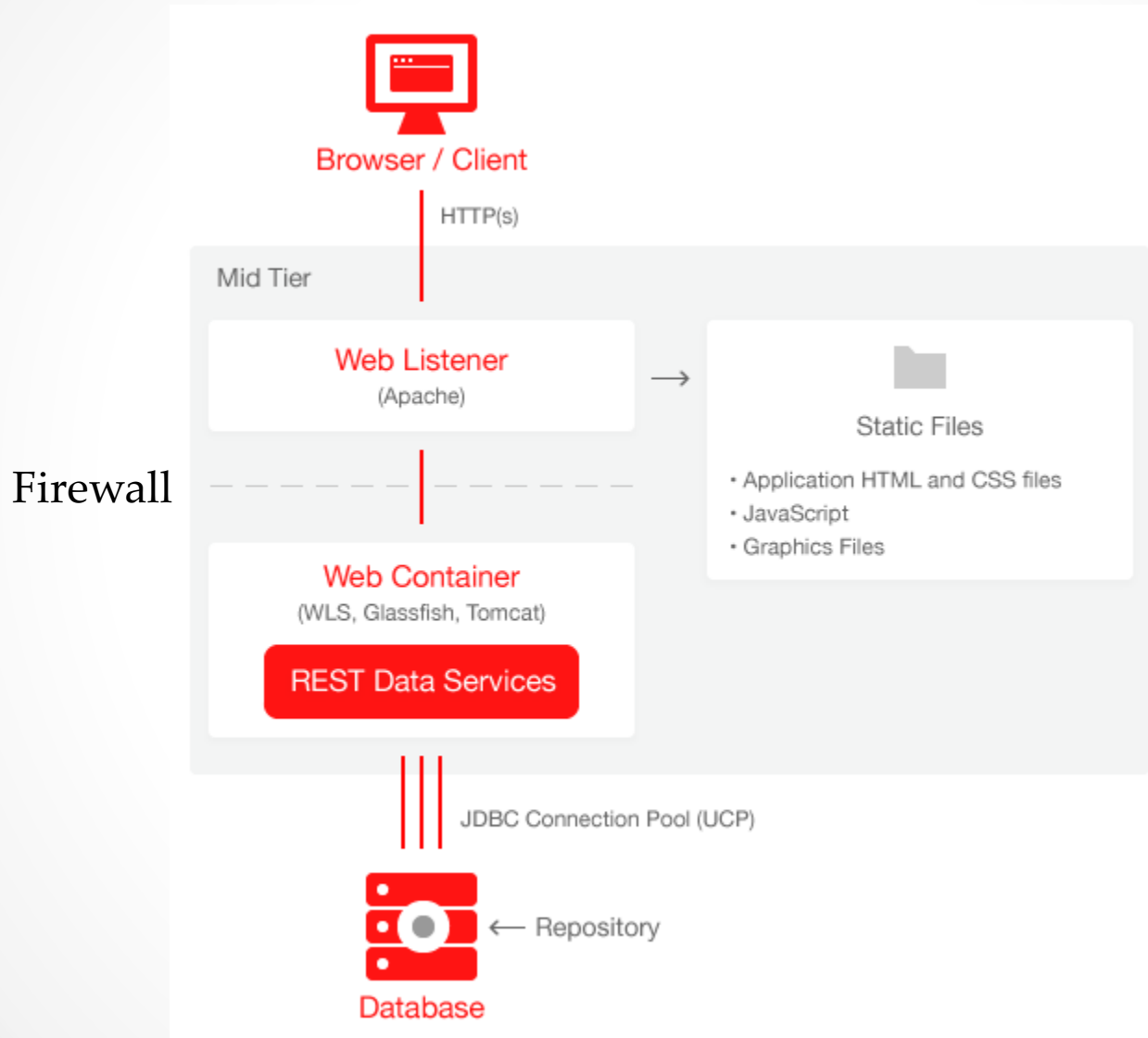
[@dgielis](#)

Agenda

- ORDS settings
 - Workspace and Application Settings
 - Authorization and Authentication
-
- SQL Injection
 - Cross Site Scripting
 - Session State Protection
 - Tools (Advisor, APEXSert, ApexSec)

ORDS

Architecture



Installation & Configuration

- Command line

```
java -jar ords.war install (simple)
```

```
java -jar ords.war install advanced
```

This Oracle REST Data Services instance has not yet been configured.

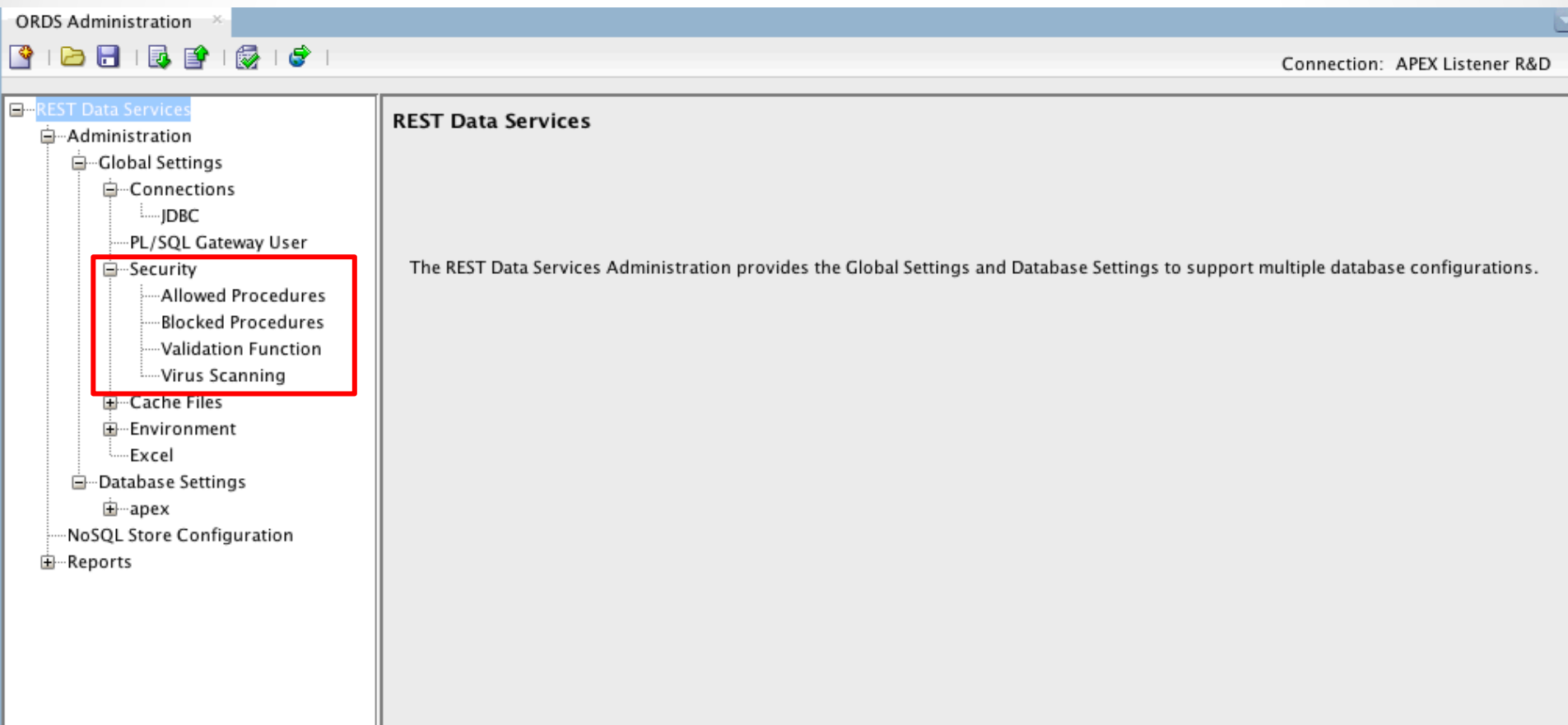
Please complete the following prompts

Enter the location to store configuration data: ...

Installation & Configuration

- SQL Developer

```
java -jar ords.war user adminlistener "Listener Administrator"
```



Installation & Configuration

- PL/SQL Validation Function

IMPORTANT

The screenshot shows the ORDS Administration console interface. The title bar reads "ORDS Administration" and "Connection: APEX Listener R&D". The left sidebar contains a tree view with the following items: REST Data Services, Administration, Global Settings, Connections, PL/SQL Gateway User, Security, Cache Files, Environment, Excel, Database Settings, apex, NoSQL Store Configuration, and Reports. The "Validation Function" item under Security is highlighted. The main content area is titled "Validation Function - Global Settings". It contains a "Validation Function Type:" label with a dropdown menu set to "PL/SQL". Below this is a text box with the instruction: "Specify a validation function to determine if the requested procedure in the URL should be allowed for processing. Note: The function should return true if the procedure is allowed; otherwise, return false." At the bottom, there is a label "Validation Function:" followed by a text input field containing the value "www_flow_epg_include_modules.authorize".

ORDS Administration

Connection: APEX Listener R&D

REST Data Services

- Administration
 - Global Settings
 - Connections
 - JDBC
 - PL/SQL Gateway User
 - Security
 - Allowed Procedures
 - Blocked Procedures
 - Validation Function
 - Virus Scanning
 - Cache Files
 - Environment
 - Excel
 - Database Settings
 - apex
 - NoSQL Store Configuration
 - Reports

Validation Function - Global Settings

Validation Function Type:

Specify a validation function to determine if the requested procedure in the URL should be allowed for processing. Note: The function should return true if the procedure is allowed; otherwise, return false.

Validation Function:

`www_flow_epg_include_modules.authorize`

Validation Function

- PL/SQL vs JavaScript Validation Function

The screenshot shows the ORDS Administration console interface. The left sidebar contains a tree view of the administration structure, with 'Validation Function' selected under 'Global Settings' > 'Security'. The main panel is titled 'Validation Function - Global Settings'. It features a 'Validation Function Type' dropdown menu set to 'JavaScript'. Below this, a text block explains the purpose of the validation function. At the bottom, a text area contains a JavaScript function named 'isValid()' that checks for 'MSIE 6' in the user agent and a specific script name 'hr.get_picture'.

ORDS Administration

Connection: APEX Listener R&D

REST Data Services

- Administration
 - Global Settings
 - Connections
 - JDBC
 - PL/SQL Gateway User
 - Security
 - Allowed Procedures
 - Blocked Procedures
 - Validation Function
 - Virus Scanning
 - Cache Files
 - Environment
 - Excel
 - Database Settings
 - apex
 - NoSQL Store Configuration
 - Reports

Validation Function - Global Settings

Validation Function Type: JavaScript ▼

Specify a validation function to determine if the requested procedure in the URL should be allowed for processing. Note: The function should return true if the procedure is allowed; otherwise, return false.

Validation Function:

```
function isValid(){
  // no more ie6
  if ( USER_AGENT.indexOf("MSIE 6") > 0 ) {
    return "false";
  }
  // a procedure in the klrice schema named HI has been performing poorly so shut it off
  if ( SCRIPT_NAME == 'hr.get_picture' ) {
    return "FALSE";
  }
}
```

APEX Recommendation

[Home](#) / [Database](#) / [Oracle Application Express Documentation](#), Release 5.0

Application Express Application Builder User's Guide

Page 163 of 224

Table of Contents

Oracle Application Express Application Builder User's Guide

20.1 Understanding Administrator Security Best Practices

20.1.2 About Configuring Oracle REST Data Services with Oracle Application Express

Oracle REST Data Services (formerly known as Oracle Oracle Application Express Listener) is a J2EE application which communicates with the Oracle Database by mapping browser requests to the Application Express engine database over a SQL*Net connection. Oracle REST Data Services is the strategic direction for Oracle Application Express and Oracle recommends using it in practically all circumstances. In a production environment, you deploy Oracle REST Data Services web archive files to a supported Java EE application server, like Oracle Web Logic Server. Each deployment can be configured individually and serves the same purpose as a `mod_plsql` Database Access Descriptor, which is to communicate with an Oracle database.

An Oracle REST Data Services deployment configuration contains several security related parameters. In a configuration for Oracle Application Express, Oracle recommends to set the parameter `security.requestValidationFunction` to `wwv_flow_epg_include_modules.authorize`. This activates the white list of callable procedures which ships with Oracle Application Express and prohibits calls to other procedures. This can be extended using the validation functions shipped with Oracle Application Express. To learn more, see "Restricting Access to Oracle Application Express by Database Access Descriptor (DAD)" in [Oracle Application Express Administration Guide](#).

Importance of updating ORDS

1.2.8 Release 3.0.7

Oracle REST Data Services release 3.0.7 includes the following new features and other changes:

- [APEX_PUBLIC_USER](#) and [wwv_flow_epg_include_modules.authorize](#)

1.2.8.1 APEX_PUBLIC_USER and wwv_flow_epg_include_modules.authorize

In line with security best practices and as recommended by [Oracle Application Express Application Builder User's Guide](#) when a database pool is configured to use the `APEX_PUBLIC_USER`, Oracle REST Data Services automatically sets the value of the `security.requestValidationFunction` setting to be:

```
WWV_FLOW_EPG_INCLUDE_MODULES.AUTHORIZE
```

This setting activates the white list of callable procedures that ships with Oracle Application Express and prohibits calls to other procedures. See the [Oracle Application Express Application Builder User's Guide](#) for more information about this procedure and how to customize its behavior.

See the `readme.html` file for additional information.

Validation Function to limit application access

- PL/SQL Validation Function

```
<entry key="apex.security.requestValidationFunction">  
is_allowed(p_procedure => :PROCNAME, p_app_id => :P_FLOW_ID, p_page_id => :P_FLOW_STEP_ID)  
</entry>
```

```
create or replace function is_allowed(  
    p_procedure in varchar2,  
    p_app_id in varchar2,  
    p_page_id in varchar2)  
    return boolean as  
begin  
    if (p_app_id = 123) then  
        return wwv_flow_epg_include_modules.authorize(p_procedure);  
    end if;  
    return false;  
end;
```

Important settings: JDBC

- Configure the database connection pool
- Set the max size
- Set the initial size
- Set the timeouts

```
<entry key="jdbc.DriverType">thin</entry>
<entry key="jdbc.InactivityTimeout">1800</entry>
<entry key="jdbc.InitialLimit">10</entry>
<entry key="jdbc.MaxConnectionReuseCount">1000</entry>
<entry key="jdbc.MaxLimit">60</entry>
<entry key="jdbc.MaxStatementsLimit">40</entry>
<entry key="jdbc.MinLimit">1</entry>
<entry key="jdbc.maxRows">50000</entry>
<entry key="jdbc.statementTimeout">900</entry>
```

Virus scanner Integration

- Scans all files uploads for viruses before it reaches the database
- Supported by most commercial Virus scan servers (Symantec, McAfee, ...)
- Open source option: ClamAV
- ICAP protocol (RFC 3507)
- <http://www.e-cap.org/Downloads> •

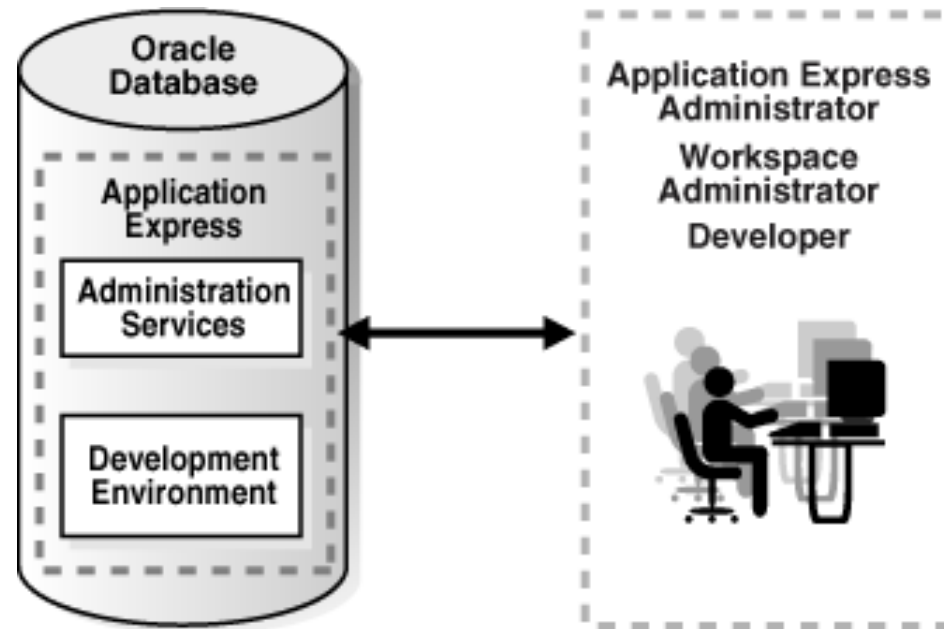
Secure REST web services

- REST web services with OAuth2

Build-in Webserver

- ORDS >= 3.0.0; improved build-in web server (Jetty)

Workspaces



Internal workspace



Oracle Application Express



internal



ADMIN



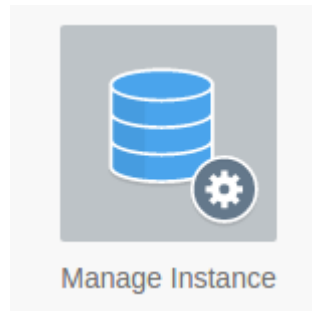
.....



Sign In

[Reset Password](#)

Instance administration



Instance Settings	
	Feature Configuration
	Security
	Instance Settings
	Workspace Purge Settings

Security

Security

Configure service level security settings typically used to lock down a production service.

Set Workspace Cookie No  

Disable Administrator Login Yes  

Disable Workspace Login Yes  

Allow Public File Upload No  

Restrict Access by IP Address 

Instance Proxy 

Checksum Hash Function Most Secure  

Rejoin Sessions Enabled for All Sessions  

Unhandled Errors Return HTTP 400  

Set Workspace Cookie



Oracle Application Express

 demosec



 aljaz



 password

Sign In

[Reset Password](#)



Oracle Application Express

 workspace

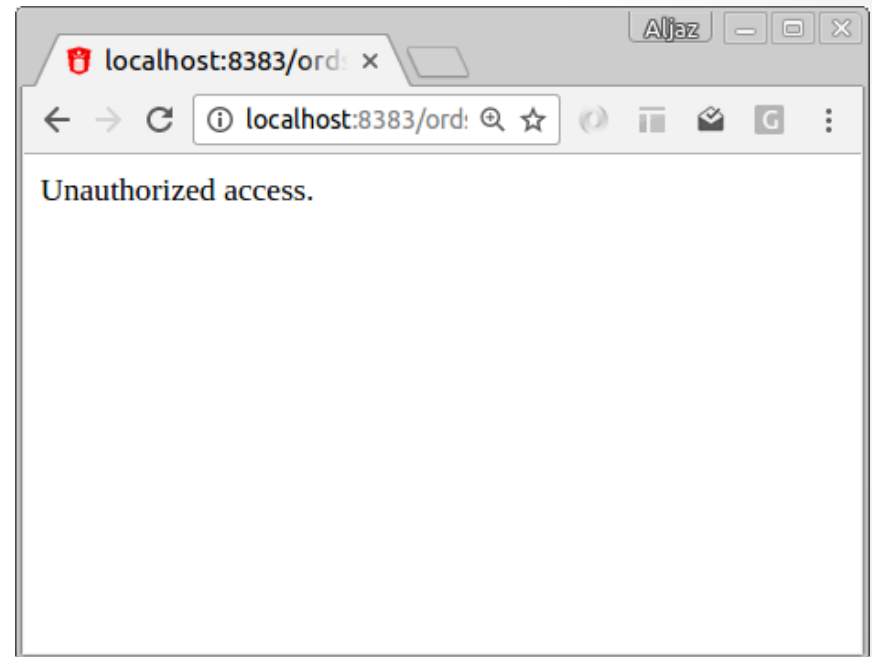
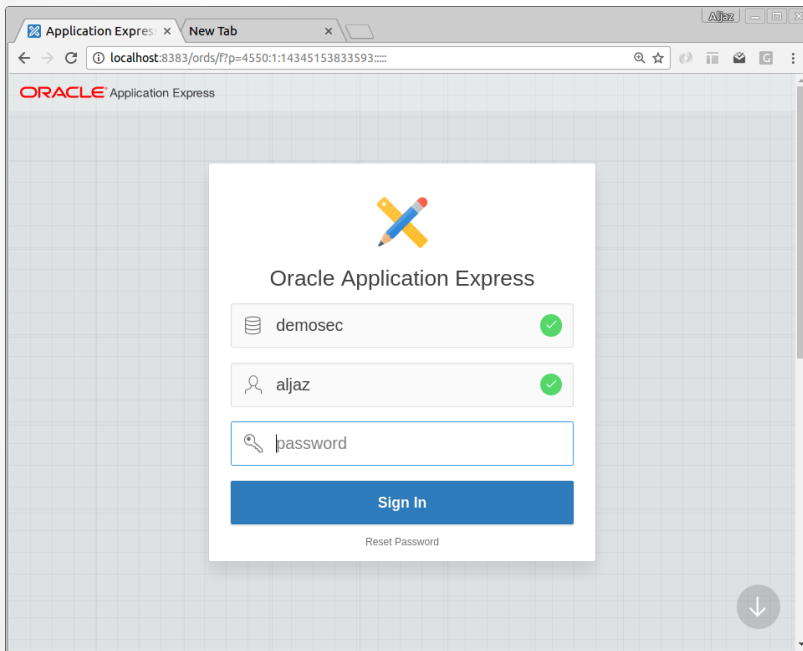
 username

 password

Sign In

[Reset Password](#)

Disable login



Rejoin sessions

When rejoin sessions is enabled, Application Express attempts to use the session cookie to join an existing session, when a URL does not contain a session ID.

- Disabled
- Enabled for Public Sessions
- Enabled for All Sessions

Can be also set on pages

A more restrictive instance level setting overrides this page level value

Rejoin sessions

Enabling rejoin sessions exposes your application to possible security breaches, as it can enable attackers to take over existing end user sessions.

- Triggering malicious session state changes or other modifications
- Triggering unintended changes between applications

For security reasons, Oracles recommends that administrators disable Rejoin Sessions unless they implement workspace isolation by configuring the Allow Hostname attribute at the workspace or instance-level.

http://docs.oracle.com/cd/E59726_01/doc.50/e39147/sec_admin_ssl.htm#CIHHIFBG

Workspace isolation

- Specify which DNS aliases of the web server can be used with applications
- Incoming HTTP request URL's hostname part must match one of the listed hostnames
- You can configure more specific values that override this one at workspace level

Workspace isolation

- instance value: www.example.com
- WS HR: hr.example.com
- Same Origin security policy provides a client-side barrier between HR applications and other applications

Security

Security

Configure service level security settings typically used to lock down a production service.

Set Workspace Cookie No  

Disable Administrator Login Yes  

Disable Workspace Login Yes  

Allow Public File Upload No  

Restrict Access by IP Address 

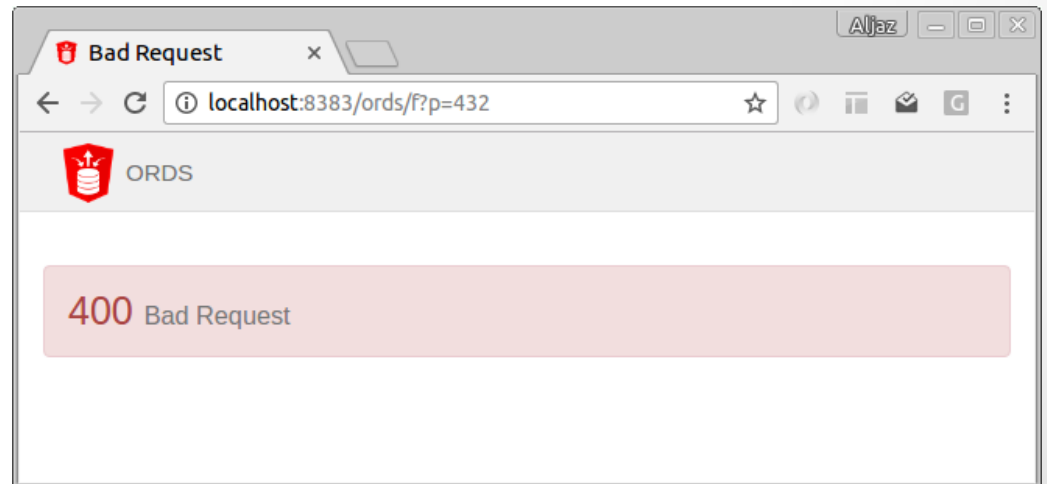
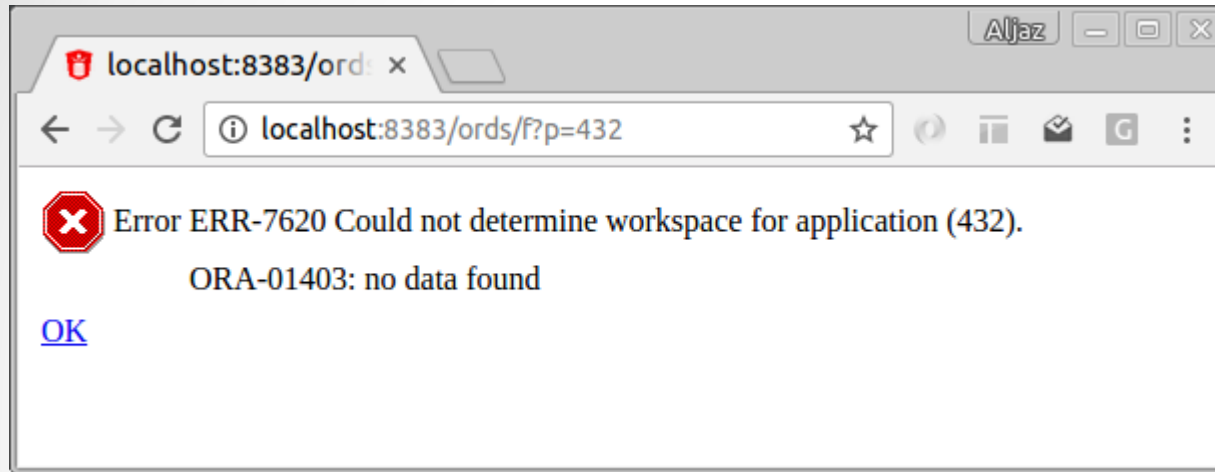
Instance Proxy 

Checksum Hash Function Most Secure  

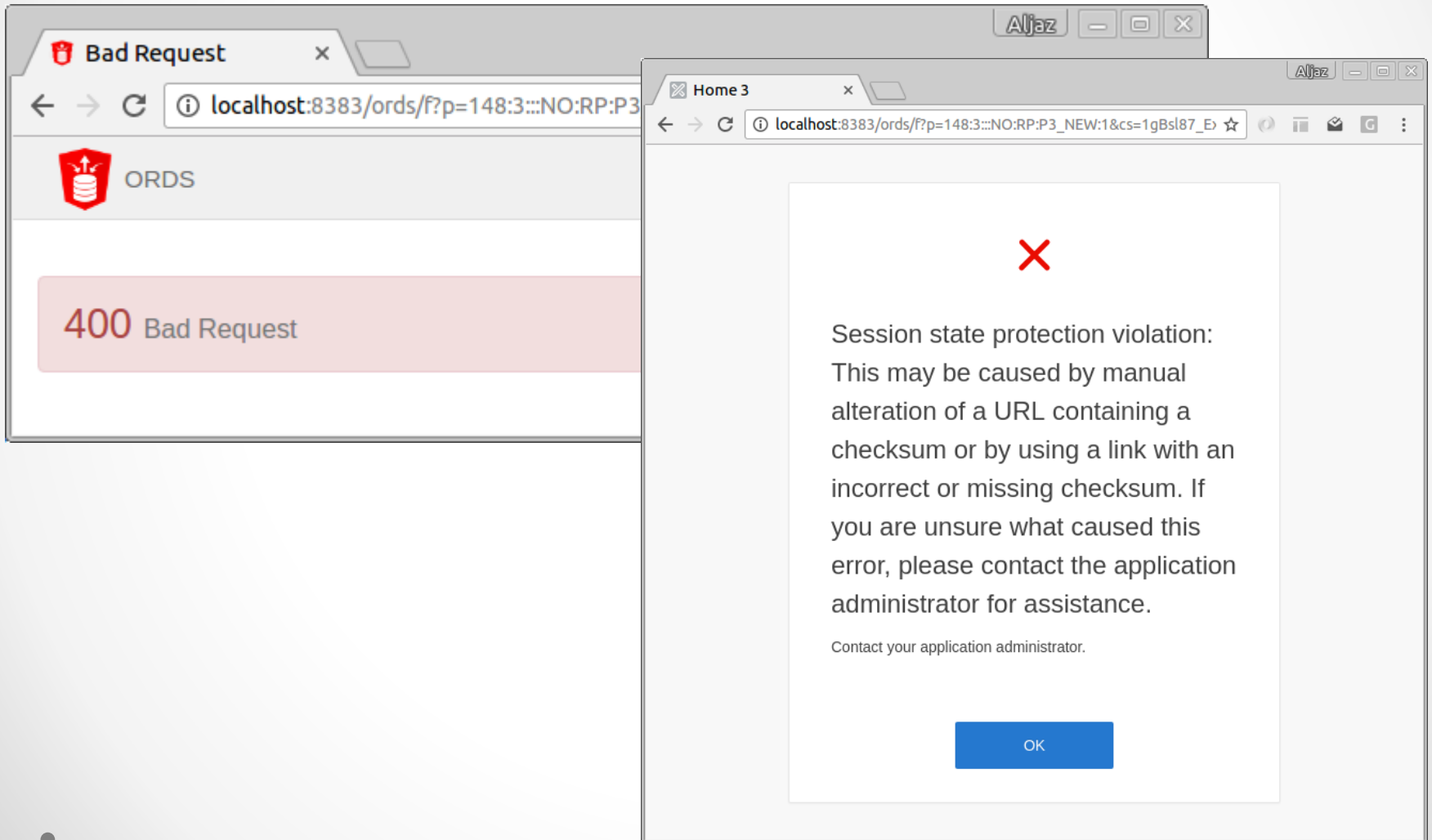
Rejoin Sessions Enabled for All Sessions  

Unhandled Errors Return HTTP 400  

Unhandled Errors



Unhandled Errors



HTTP protocol

HTTP Protocol

Warning: *Require HTTPS* will make Application Express unreachable by the HTTP protocol. Before enabling this setting, ensure that the HTTPS protocol is enabled and configured correctly on your server.

Require HTTPS  

Require Outbound HTTPS  

HTTP Response Headers



HTTP Response Headers

- <https://scotthelme.co.uk/hardening-your-http-response-headers/>
- <https://securityheaders.io/>

Security Report Summary



Site:	https://apex.oracle.com/pls/apex/f?p=102895:1:
IP Address:	141.146.114.8
Report Time:	14 Oct 2016 08:04:54 UTC
Report Short URL:	https://schr.io/dDN
Headers:	✓ X-Frame-Options ✗ Strict-Transport-Security ✗ Content-Security-Policy ✗ Public-Key-Pins ✗ X-XSS-Protection ✗ X-Content-Type-Options

HTTP Response Headers

Content-Security-Policy: default-src 'self' 'unsafe-inline' 'unsafe-eval' #CDN#;
object-src 'self'; img-src 'self' data;;
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff

Security Report Summary



Site:	https://apex.oracle.com/pls/apex/f?p=102878:1:
IP Address:	141.146.114.8
Report Time:	14 Oct 2016 08:10:21 UTC
Report Short URL:	https://schr.io/dDQ
Headers:	✓ X-XSS-Protection ✓ Content-Security-Policy ✓ X-Content-Type-Options ✓ X-Frame-Options ✗ Strict-Transport-Security ✗ Public-Key-Pins
Warning:	Grade capped at A, please see warnings below.

Other settings

- RESTful Access, expose report regions as RESTful services
- Session Timeout (session length, idle time)
- Delay after failed login attempts in Seconds
- Password Policy

Password policy

Password Policy

Manage password policy for Application Express users (workspace administrators, developers, and end users) in all workspaces.

Password Hash Function

Most Secure

Minimum Password Length

0

Minimum Password Differences

0

Must Contain At Least One Alphabetic Character

No

Must Contain At Least One Numeric Character

No

Must Contain At Least One Punctuation Character

No

Must Contain At Least One Upper Case Character

No

Must Contain At Least One Lower Case Character

No

Must Not Contain Username

No

Must Not Contain Workspace Name

No

Must Not Contain

oracle:hello:welcome:guest:user:database

Alphabetic Characters

abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ

Punctuation Characters

!"#\$%&'()*+,-./:;<=>?_

Service Administrator Password Policy

Use policy specified in Workspace Password Policy

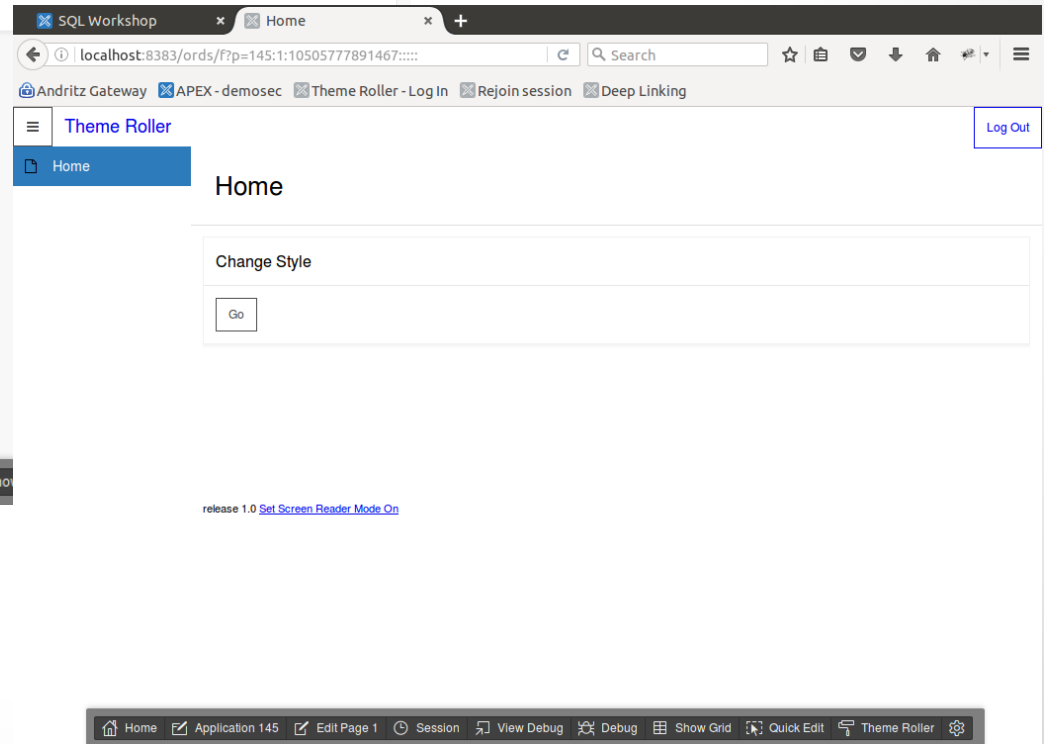
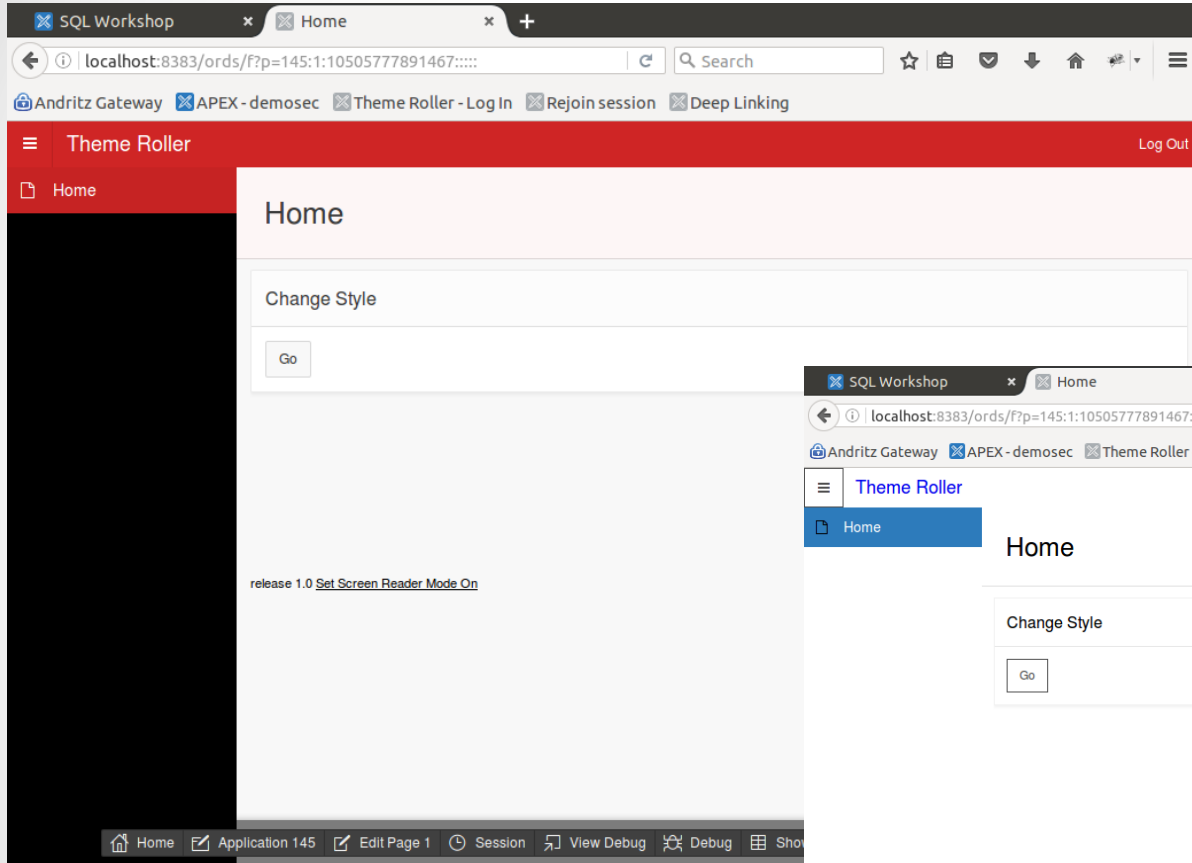
☒ Use default strong password policy

Invalid Login Credentials
Please wait 7 seconds to login again.

Feature Configuration

- SQL Workshop
 - Enable RESTful Services
 - Controls the ability to create and access RESTful Services mapped to SQL and PL/SQL. RESTful Services can also be enabled or disabled for individual workspaces.
 - Theme Roller
 - When setting to “No”, restart ORDS

Theme Roller



Create workspace

- Connecting database schema and APEX
- Check privileges for existing schema
- Review privileges if APEX creates new schema

Select whether or not the schema already exists. If the schema exists, select the schema from the list. If the schema does not exist, enter a name and password and choose the size of the associated tablespace to be created.

Re-use existing schema?	No		
* Schema Name	DEMOSEC		
* Schema Password	*****		
* Space Quota (MB)	500		

New schema privileges

Worksheet

Query Builder

1

```
select * from SYS.DBA_SYS_PRIVS where grantee = 'DEMOSEC';
```

Query Result x

    SQL | All Rows Fetched: 14 in 0.007 seconds

	GRANTEE	PRIVILEGE	ADMIN_OPTION
1	DEMOSEC	CREATE INDEXTYPE	NO
2	DEMOSEC	CREATE SEQUENCE	NO
3	DEMOSEC	CREATE TABLE	NO
4	DEMOSEC	CREATE DIMENSION	NO
5	DEMOSEC	CREATE SYNONYM	NO
6	DEMOSEC	CREATE OPERATOR	NO
7	DEMOSEC	CREATE TRIGGER	NO
8	DEMOSEC	CREATE VIEW	NO
9	DEMOSEC	CREATE SESSION	NO
10	DEMOSEC	CREATE JOB	NO
11	DEMOSEC	CREATE CLUSTER	NO
12	DEMOSEC	CREATE TYPE	NO
13	DEMOSEC	CREATE MATERIALIZED VIEW	NO
14	DEMOSEC	CREATE PROCEDURE	NO

Worksheet

Query Builder

1

```
select * from SYS.DBA_ROLE_PRIVS where grantee = 'DEMOSEC';
```

Query Result x

SQL | All Rows Fetched: 0 in 0.008 seconds

	GRANTEE	GRANTE...	ADMIN_...	DEFAULT...
--	---------	-----------	-----------	------------

Worksheet

Query Builder

1

```
select * from SYS.DBA_TAB_PRIVS where grantee = 'DEMOSEC';
```

Query Result

SQL | All Rows Fetched: 0 in 0.03 seconds

GRANTEE

OWNER

TABLE_N...

GRANTOR

PRIVILEGE

GRANTA...

HIERARC...

Application Settings



Oracle Application Express



demosec



aljaz



.....



Sign In

[Reset Password](#)

Application Security Settings

Authentication

Authentication is the process of establishing each user's identity before they can access your application. You may define multiple authentication schemes for your application, however c

Application **145** ?

Public User ?

Authentication Scheme ?

Authorization

Application authorization schemes control access to all pages within an application. Unauthorized access to the application, regardless of which page is requested, will cause an error pa

Authorization Scheme ?

Run on Public Pages ?

Session Management

Rejoin Sessions ?

Deep Linking ?

Maximum Session Length in Seconds ?

When Maximum Session Length is not set, the workspace level or instance level value will be used (currently, 28800 seconds).

Session Timeout URL ?

Maximum Session Idle Time in Seconds ?

When Maximum Session Idle Time is not set, the workspace level or instance level value will be used (currently, 3600 seconds).

Session Idle Timeout URL ?

Authentication

Application **145** ?

Public User ?

Authentication Scheme ?

- Public User

- Identifies the Oracle schema (or user) used to connect to the database through the Database Access Descriptor (DAD)
- (USER_IS_PUBLIC_USER, USER_IS_NOT_PUBLIC_USER, APEX_APPLICATION.G_PUBLIC_USER)

Authorization

Authorization Scheme ?

Run on Public Pages ?

- Authorization
 - Authorization Scheme (authorization scheme for your application – every page)
- Run on Public Pages
 - Controls whether the application-level authorization scheme is checked on public pages

Session management

- Rejoin Sessions
- Maximum Session Length in Seconds (0, >0, empty)
- Deep Linking

Deep Linking

Browsers often save the URLs of opened tabs and try to restore the sessions after a restart, causing a deep link. This behavior may be undesirable (for example if a URL points to a page in the middle of a multi-step wizard).

- **Enabled** - The URL to a specific page ultimately redirects there, possibly after the user has to logged in.
- **Disabled** - If the URL does not contain a valid session ID, Application Express redirects to the application's home page.

Browser Security

- Cache
 - Oracle recommends that this attribute be disabled
 - HTTP header cache-control: no-store (modern browsers)
- Embed in Frames
- HTML Escaping Mode
 - Basic - Escape &, ", < and >
 - Extended - Escape &, ", <, >, ', / and non-ASCII characters if the database character set is not AL32UTF8
- HTTP Response Headers

Database Session

- Parsing Schema
 - #OWNER
- Initialization PL/SQL Code
 - "show page" or "accept page" request
 - after the APP_USER value is established
 - dbms_session.set_context,...
- Cleanup PL/SQL Code
 - at the end of page processing
 - dbms_session.close_database_link,...

Runtime API Usage

- Modify This Application
 - APEX_UTIL.SET_CURRENT_THEME_STYLE,...
- Modify Other Applications
- Modify Workspace Repository
 - APEX_UTIL.CREATE_USER,...

Form Autocomplete

- By default this is enabled on all pages
- Data will be stored on the client's local browser
- `autocomplete="off"`
- friendly vs. secured

Application Definition

Logging Yes  

Debugging No  

Status Available  

Build Status Run Application Only  

Message for unavailable application This application is currently unavailable at this time. 

Restrict to comma separated user list (status must equal Restricted Access) aljaz, dimitri 

Error handling function



1 error has occurred

ORA-00001: unique constraint (DEMOSEC.DEMO_PK) violated



1 error has occurred

This error is handled by error handling function

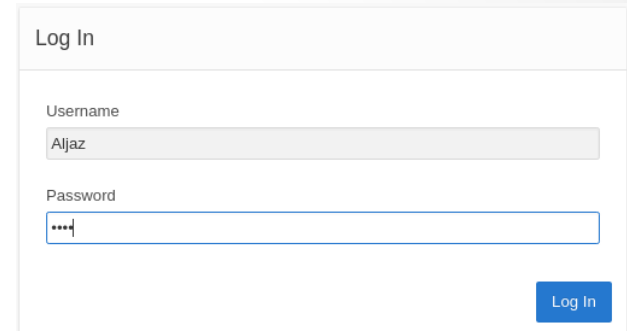
Authentication

- Authentication is the process of establishing each user's identity **before they can access your application**
- The current Authentication Scheme determines **how** your application **identifies** and **verifies** the end user



Authentication Schemes

- Database Accounts
- Application Express Accounts
- LDAP Directory
- Oracle Application Server Single Sign-On
- HTTP Header Variable
- No Authentication (Using DAD)
- Open Door Credentials
- Custom



The image shows a web-based login form. At the top, there is a header section with the text 'Log In'. Below this, there are two input fields. The first is labeled 'Username' and contains the text 'Aljaz'. The second is labeled 'Password' and contains four asterisks '****'. To the right of these fields is a blue button with the text 'Log In' in white.

Authentication settings

Sentry Function Name		?
Invalid Session Procedure Name		?
Authentication Function Name	my_custom_function	?
Post Logout Procedure Name		?
Enable Legacy Authentication Attributes	No 	?

```
1 function my_custom_function(p_username varchar2, p_password varchar2)
2   return boolean
3 is
4 begin
5
6   if upper(p_username) = 'A' and upper(p_password) = 'B' then
7     return true;
8   else
9     return false;
10  end if;
11
12 end;
```

Sentry function

- Function that is executed by the Application Express engine **at the start of any request** made to the engine, such as **before each page is shown or processed**, or an AJAX request is issued
- If this function returns false, this marks the session as not valid and the Invalid Session Procedure will be invoked. After that, Application Express redirects to the URL defined in 'Session Not Valid > Go To'

Invalid Session Procedure

- PL/SQL procedure that gets called if an invalid session has been detected

```
1 procedure invalid_session_basic_auth
2 is
3 begin
4
5     owa_util.status_line (
6         nstatus      => 401,
7         creason       => 'Basic Authentication required',
8         bclose_header => false);
9
10    http.p('WWW-Authenticate: Basic realm="protected realm"');
11
12    apex_application.stop_apex_engine;
13
14 end invalid_session_basic_auth;
```

Verify Function Name

- Called after the session sentry returned successfully
- This function can for example be used to restrict the use of an application to specific business hours

```
1 function check_business_hours return boolean
2 is
3 begin
4     return to_char(to_char(sysdate, 'hh24:mi')) between '08:00' and '17:00';
5 end check_business_hours;
```

Login Processing

- Pre-Authentication Procedure
 - procedure to be executed after the login page is submitted and just before credentials verification is performed
- Post-Authentication Procedure
 - procedure to be executed by the Application Express login procedure (login API) after the authentication step
 - The login procedure will execute this code after it verified the user's credentials, but before registering the user in the session and redirecting to the desired application page

```
1 apex_custom_auth.set_user (  
2   p_user => regexp_replace(:APP_USER, '@.*', null) );  
3 |
```

Post Logout Procedure

- Procedure that gets called after the end user clicked on the logout URL.
- It can be used for logging.

```
1 procedure my_post_logout
2 is
3 begin
4     insert into logout_log ( application_id, session id, logout_time, user_name )
5     values ( :APP_ID, :APP_SESSION, sysdate, :APP_USER );
6 end;
```

Session Cookie

- Setting session cookie, which is required to identify an Application Express session, together with the session id in the URL. If no value for the session cookie name is specified, Application Express picks a default value.
- Sharing authentication across multiple APEX applications
- Secure: YES
 - Allow the session management cookie to be sent from the browser only when the protocol is HTTPS

Authorization Schemes

Authorization schemes enable you to protect applications, pages, and application components (region, button, item, processes, ...)



Authorization Schemes

Authorization Scheme

* Scheme Type ?

* PL/SQL Function Body ?

* Identify error message displayed when scheme violated ?

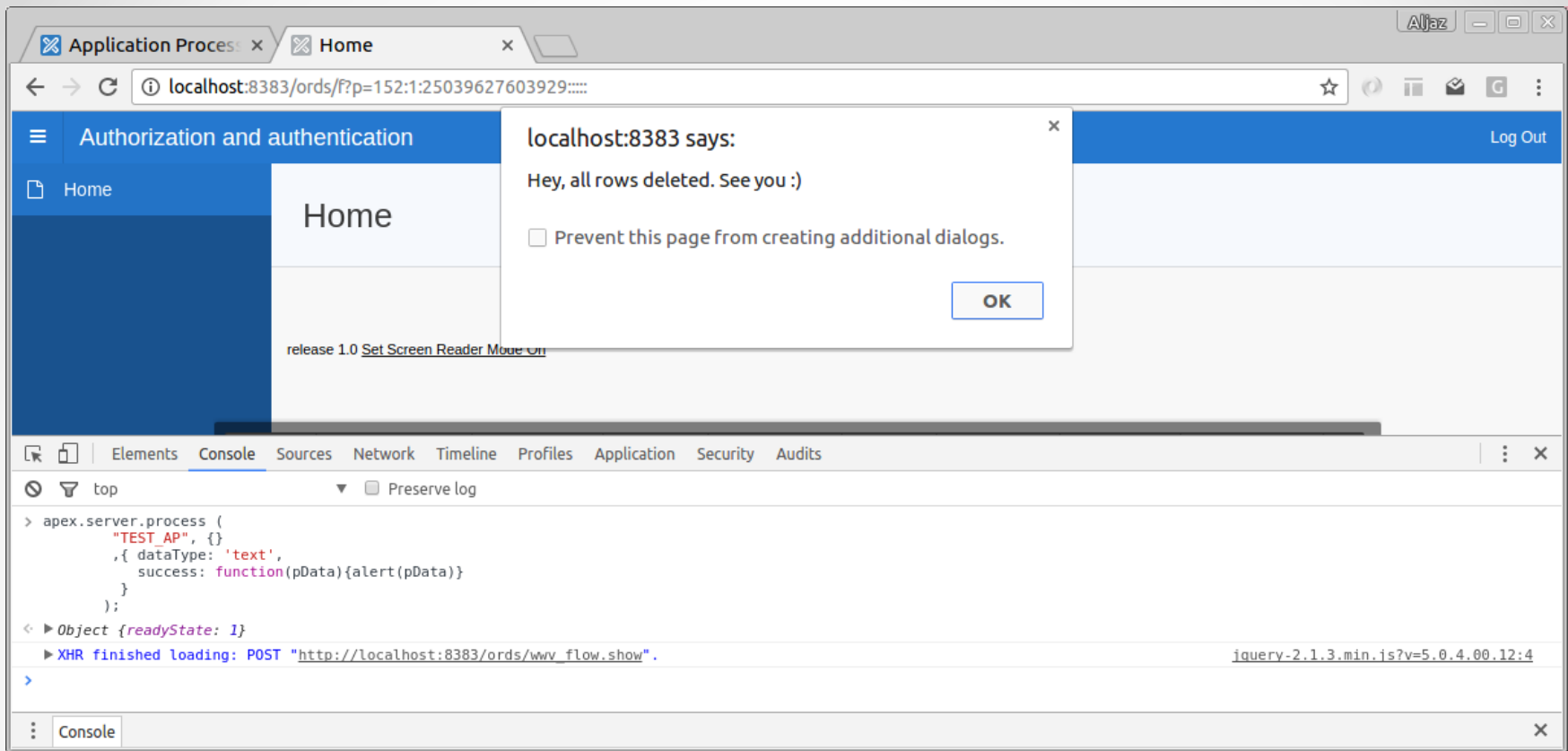
Evaluation Point

- Once per session
- Once per page view
- Once per component
- Always (No Caching)

The default value Once per session is the most efficient

Don't overlook

- If you protect buttons, don't forget to also protect processes (process can be invoked over URL or javascript)
- Protect application processes (they can be called over URL or javascript)
- Use AJAX Callback on pages



```
apex.server.process (
  "TEST_AP", {}
  ,{ dataType: 'text',
    success: function(pData){alert(pData)}
  }
);
```

SQL Injection

What is SQL Injection?

- SQL Injection vulnerabilities arise when the end-users (attackers) can modify the syntax of a database query.



Impact of SQL Injection

- See any data (also in other tables)
- Do DML (insert, update, delete) operations
- Run PL/SQL

Most common threats

- Use of substitution strings (&ITEM.)
- Use of Dynamic SQL: wrong concatenations
- Report with source SQL Query (PL/SQL function body returning SQL Query)
- Use of Execute Immediate in PL/SQL
- Use of SQL in parameter e.g. APEX_COLLECTION

Where?

- Any component where SQL or PL/SQL is used!
 - Computations
 - Processes
 - Reports
 - Charts
 - Item Source
 - Display Conditions
 - List of Values
 - Lists
 - Authorization schemes
 - ...

Substitution Strings

```
select *  
  from emp  
 where ename = '&P1_SEARCH.'
```



Substitution Strings

Search1

KING KING' or 1=1--

Ename	Job	Mgr	Hiredate	Sal	Comm
KING	PRESIDENT	-	17-NOV-81	5000	-

Substitution Strings

```
select *  
  from emp  
 where ename = :P1_SEARCH
```



Substitution Strings

```
select
  null as link,
  year as label,
  sum(amount) as "Year &P1_YEAR."
from my_table
```



Substitution Strings

```
select
  null as link,
  year as label,
  sum(amount) as "Year &P1_YEAR."
from my_table
```

Protect the item P1_YEAR

Restricted
Characters

Whitelist for a-Z, 0-9 and space



Bind variables & Dynamic SQL

```
l_sql :=  
'SELECT *  
  FROM emp  
 WHERE empno = ' || :P1_EMPNO;  
  
RETURN l_sql;
```



Bind variables & Dynamic SQL

```
l_sql :=  
'SELECT *  
  FROM emp  
 WHERE empno = :P1_EMPNO';  
  
RETURN l_sql;
```



Bind variables & Dynamic SQL

```
l_sql :=  
'SELECT *  
  FROM emp  
 WHERE empno = to_number(:P1_EMPNO)';  
  
RETURN l_sql;
```



Bind variables & Dynamic SQL

```
l_id := to_number(:P1_EMPNO);  
l_sql :=  
  'SELECT *  
    FROM emp  
   WHERE empno = ' || l_id;  
  
RETURN l_sql;
```



Bind variables & Dynamic SQL

```
l_like := '%' || :P1_SEARCH || '%';  
l_sql :=  
  'SELECT *  
    FROM emp  
   WHERE ename like ' || l_like ;  
  
RETURN l_sql;
```



Bind variables & Dynamic SQL

```
l_like := '%' || :P1_SEARCH || '%';  
l_sql :=  
'SELECT *  
  FROM emp  
 WHERE ename like '  
        || DBMS_ASSERT.ENQUOTE_LITERAL(l_like);  
  
RETURN l_sql;
```



v() function & Dynamic SQL

```
l_sql :=  
'SELECT *  
  FROM emp  
 WHERE empno = ' || v( 'P1_EMPNO' );  
  
RETURN l_sql;
```



v() function & Dynamic SQL

```
l_sql :=  
'SELECT *  
  FROM emp  
 WHERE empno = v(''P1_EMPNO'')';  
  
RETURN l_sql;
```



Dynamic SQL

```
l_column := :P1_COLUMN;  
l_table  := :P1_TABLE;  
l_sql :=  
'SELECT ' || l_column ||  
' FROM ' || l_table;  
  
RETURN l_sql;
```



Dynamic SQL

```
l_column := DBMS_ASSERT.SIMPLE_SQL_NAME(:P1_COLUMN);  
l_table   := DBMS_ASSERT.SIMPLE_SQL_NAME(:P1_TABLE);  
  
l_sql :=  
'SELECT ' || l_column ||  
' FROM ' || l_table;  
  
RETURN l_sql;
```



Fixes

- Use bind variables *correctly*
- Careful with concatenations!
- DBMS_ASSERT.SIMPLE_SQL_NAME
- DBMS_ASSERT.ENQUOTE_LITERAL
- Item protection

Cross Site Scripting - XSS

Cross-site scripting (XSS) is a type of computer security vulnerability typically found in web applications. XSS enables attackers **to inject client-side scripts into web pages viewed by other users.**

Cross-site scripting carried out on websites accounted for roughly **84% of all security vulnerabilities** documented by Symantec as of 2007.[1]

Their effect may range from a petty nuisance to a significant security risk, depending on the sensitivity of the data handled by the vulnerable site and the nature of any security mitigation implemented by the site's owner.

Many Types of XSS

- Stored XSS
 - JavaScript in database
- Reflected XSS
 - Embedded JavaScript in URL request
- Stored XSS in uploaded files
 - HTML, Text file with .jpg extension, etc.

APEX and XSS

Form on DEMO_XSS

Xss

```
<script>alert('Hello world');</script>
```

Cancel Delete Apply Changes

localhost:8383/ords/f?p=151:1:3846451719914:.....

Cross Site Scripting

XSS Report

Q v

localhost:8383 says:
Hello world
☐ Prevent this page from creating additional dialogs.
OK

Log Out

Create

Xss

1 - 1

XSS prevention

- When saving data
 - Restricted Characters
 - All characters can be saved.
 - Whitelist for a-Z, 0-9 and space
 - Blacklist HTML command characters (<>")
 - Blacklist &<>"/;,* | =% and -- (includes pl/sql comment)
 - Blacklist &<>"/;,* | =% or -- and new line
- When displaying data
 - Escape special characters : YES (default)
 - Manually : apex_escape

Escaping

```
<script>alert("test");</script>
```



```
&lt;script&gt;alert(&quot;test&quot;);&lt;&#x2F;script&gt;
```

Original	Escaped
&	&
"	"
<	<
>	>
'	'
/	/

APEX and XSS

- APEX is doing job protecting against XSS attacks (but it depends what the developer is doing)
- (Display) items are protected by default
- Reports (columns) are protected by default
- &PAGE_ITEM. is always protected
- What about &APP_ITEM. ????
- URL is escaped
- Rich Text Editor

apex_escape

```
1 select |
2   "ID",
3   "RTE",
4   "RTE" RTE PROT,
5   APEX_ESCAPE.HTML(RTE) "RTE APEX_ESCAPE",
6   APEX_ESCAPE.HTML_WHITELIST(RTE) "SHOW ESCAPED",
7   APEX_ESCAPE.HTML_WHITELIST(RTE) "RTE_HTML_WHITELIST",
8   APEX_ESCAPE.HTML_WHITELIST(RTE,
9     rte.f html whitelist tags('<table>,</table>,<tbody>,</tbody>,<td>,</td>,<tr>,</tr>'))
10     "RTE_HTML_WHITELIST_I"
11
12 from
13   "#OWNER#". "DEMO_RTE1"
14
15
```

apex_escape.html

- This function escapes characters which can change the context in an html environment.
- By default, the escaping mode is "Extended"
- `APEX_ESCAPE.SET_HTML_ESCAPING_MODE`
- If the mode is "Basic", the function behaves like `sys.htf.escape_sc`

apex_escape.html_whitelist

- The HTML_WHITELIST function performs HTML escape on all characters in the input text except the specified whitelist tags.
- This function can be useful if the input text contains simple html markup but a developer wants to ensure that an attacker cannot use malicious tags for cross-site scripting.

apex_escape.html_whitelist

```
APEX_ESCAPE.HTML_WHITELIST (  
  p_html IN VARCHAR2,  
  p_whitelist_tags IN VARCHAR2 DEFAULT c_html_whitelist_tags )  
  return VARCHAR2;
```

c_html_whitelist_tags:

<h1>,</h1>,<h2>,</h2>,<h3>,</h3>,<h4>,</h4>,<p>,</p>,,,,,<i>,</i>,,,,,,,,,<dl>,</dl>,<dt>,</dt>,<dd>,</dd>,<pre>,</pre>,<code>,</code>,
,
,
,
,<hr />

Page items

▼ Identification

Title	&P9_MY_ITEM.		
Type	Static Content	↕	⋮

▼ Source

PL/SQL Code

```
:P9_MY_ITEM := '<script>alert("test");</script>';
```

```
<script>alert("test");</script>
```

Application items

▼ Identification

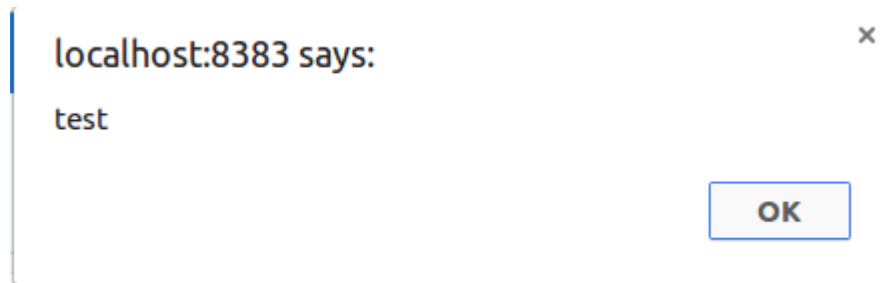
Title &MY_APP_ITEM.

Type Static Content

▼ Source

PL/SQL Code

:MY_APP_ITEM := '<script>alert("test");</script>';



▼ Source

PL/SQL Code

:MY_APP_ITEM := apex_escape.html('<script>alert("test");</script>');

PL/SQL Dynamic Region

declare

l_my_content varchar2(32000);

begin

select my_content

into l_my_content

from my_table

where id = :P1_ID;

http.p(apex_escape.html(l_my_content));

end;

Rich Text Editor

- <http://recx ltd.blogspot.si/2012/03/securing-oracle-apex-allow-rich-text.html>
- `loadjava -resolve -genmissing -user u/p Antisamy.jar`
 - `Antisamy/policies/antisamy-anythinggoes-1.4.4.xml`
 - `Antisamy/policies/antisamy-ebay-1.4.4.xml`
 - `Antisamy/policies/antisamy-myspace-1.4.4.xml`
 - `Antisamy/policies/antisamy-slashdot-1.4.4.xml`
 - `Antisamy/policies/antisamy-tinymce-1.4.4.xml`
 - `Antisamy/policies/default.xml`

Rich Text Editor

```
create or replace function recx_antisamy_scan (dirtyinput varchar2)
  return varchar2
as language java
name 'recx_antisamy.scan (java.lang.String) return java.lang.String';

create or replace procedure recx_antisamy_set_policy (dirtyinput varchar2)
as language java
name 'recx_antisamy.policy (java.lang.String) ';

--Before Page Header
recx_antisamy_set_policy('antisamy-myspace-1.4.4.xml');

select
  rte.recx_antisamy_scan(rte) rte_clean
from
  DEMO_RTE;
```

Session state protection

- Enabling Session State Protection can prevent hackers from tampering with URLs within your application
- Enabling Session State Protection is a two-step process.
 - First, you enable the feature.
 - Second, you set page and item security attributes
- Any items passed through protected page via the URL must have a checksum

Page Access Protection

- Unrestricted
- Arguments Must Have Checksum
- No Arguments Supported
 - URL can not contain Request, Clear Cache, or Name/Value Pair arguments
- No URL Access
 - page may be the target of a Branch to Page branch type, as this does not perform a URL redirect

Application and Page items

- Unrestricted
- Restricted - May not be set from browser -
 - Display Only (Save State=No)
 - Text Field (Disabled, does not save state)
 - Stop and Start Grid Layout (Displays label only)
- Checksum Required: Application Level (Deep Linking)
- Checksum Required: User Level (Deep Linking)
- Checksum Required: Session Level

PREPARE_URL Function

```
APEX_UTIL.PREPARE_URL (  
  p_url          IN VARCHAR2,  
  p_url_charset  IN VARCHAR2 default null,  
  p_checksum_type IN VARCHAR2 default null)  
RETURN VARCHAR2;
```

```
DECLARE  
  l_url varchar2(2000);  
  l_app number := v('APP_ID');  
  l_session number := v('APP_SESSION');  
BEGIN  
  l_url := APEX_UTIL.PREPARE_URL(  
    p_url => 'f?p=' || l_app || ':1:' || l_session || '::NO::P1_ITEM:xyz',  
    p_checksum_type => 'SESSION');  
END;
```

p_checksum_type=>SESSION or 3, PRIVATE_BOOKMARK or 2, or PUBLIC_BOOKMARK or 1

Store value encrypted in session state

- If the contents of an item contain sensitive data, then you should encrypt the value when it is stored in the Application Express session state management tables. Otherwise, anyone with rights to read the Application Express meta data tables could potentially write a query to extract this sensitive data.
- Only values up to 4000 bytes in length can be encrypted. Attempts to encrypt values longer than 4000 bytes produce an error message.

Tools

Tools

- Built-in Advisor (Application -> Utilities -> Advisor)
- ApexSec
 - <https://apexsec.recx.co.uk/>
- APEX-SERT
 - <http://www.oraopensource.com/blog/?category=APEX-SERT>

Secure your APEX application

Dimitri Gielis, APEX R&D
Aljaz Mali, Abakus Plus

