

Secure your APEX application

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



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](#)

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>

Agenda

- Security still an issue?
- ORDS settings
- Workspace and Application Settings
- Authentication and Authorization
- VPD, RAS, Shadow Schema
- SQL Injection
- Cross Site Scripting
- Session State Protection
- SSL and Reverse Proxy
- Tools (Advisor, APEXSert, ApexSec)

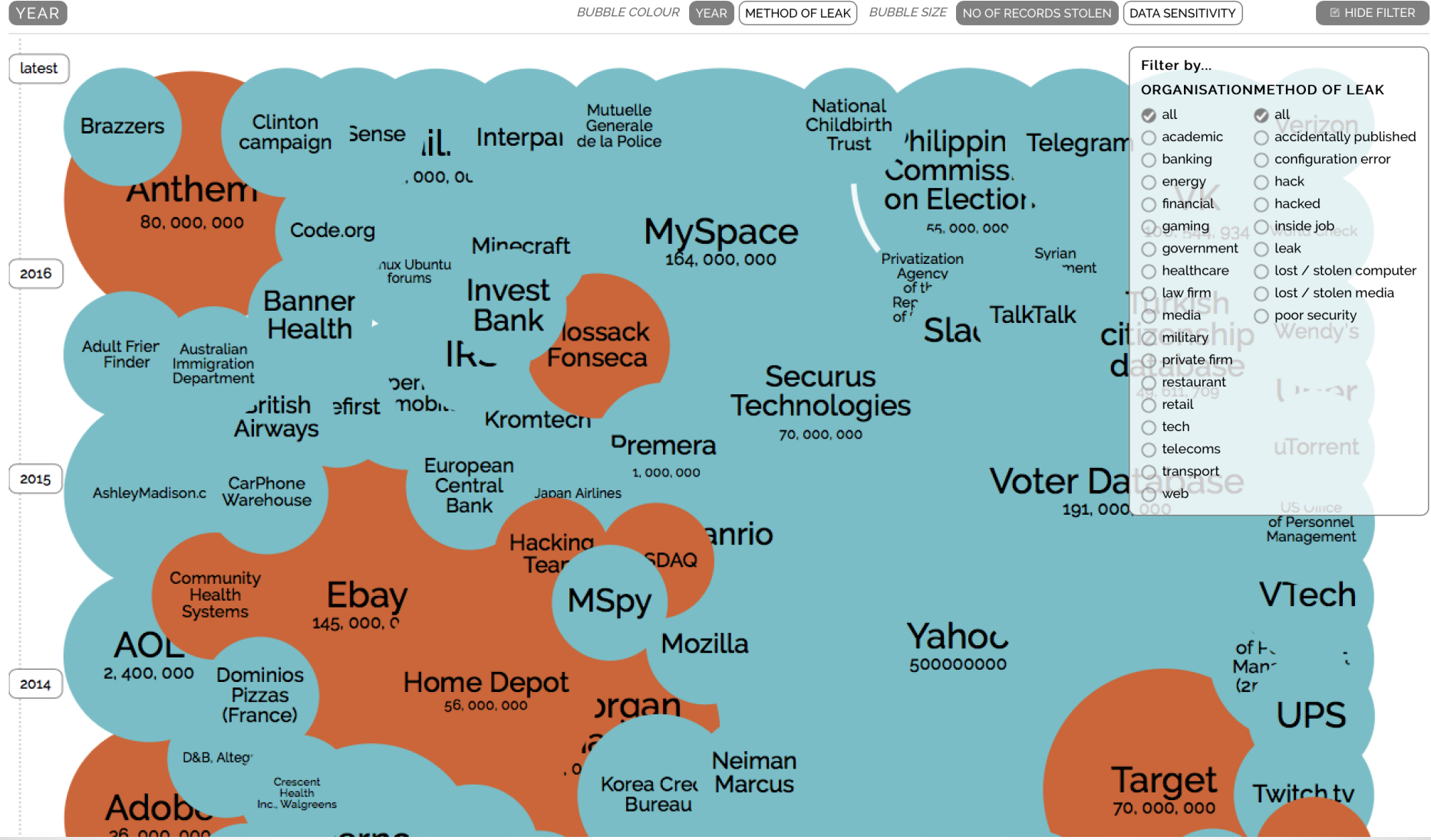
Security still an
issue?

World's Biggest Data Breaches

Selected losses greater than 30,000 records

(updated 24rd September 2016)

interesting story





Apache Tomcat

Hurry up, fix the CVE-2016-5425 privilege escalation flaw in Apache Tomcat

The security research Dawid Golunski reported a Root Privilege Escalation in the Apache Tomcat (RedHat-based distros) tracked as CVE-2016-5425. Apache Tomcat packages provided by default repositories of RedHat-based distributions (i.e. CentOS, RedHat,...

October 11, 2016 By Pierluigi Paganini Posted In [Breaking News](#) [Hacking](#)

- <http://securityaffairs.co/wordpress/>

Challenges on Security

- How secure is secure enough?
- Security taken into account from day 1; it's a process not a 1-time activity
- Before the facts vs After the facts



A black and white photograph of a person's back wearing a dark t-shirt. The t-shirt has the text "Everybody needs a hacker" printed on it in a white, sans-serif font. The background is blurred, showing what appears to be a bright light source.

Everybody needs a hacker



OWASP

The Open Web Application Security Project



Category:OWASP Top Ten Project

- Main
- OWASP Top 10 - 2016 Data Call Questions
- OWASP Top 10 for 2013
- OWASP Top 10 for 2010
- Translation Efforts
- Project Details

Some Commercial & OWASP Uses of the Top 10



OWASP Top 10 - 2016 Data Call Announcement

Public Notice: The OWASP Top 10 project is launching its effort to update the Top 10 again. The current version was released in 2013, so this update is expected to be the 2016 or more likely 2017 release. This time around, we are making an open data call so any organization with a broad set of application vulnerability statistics can contribute their data to the project. To make it easier for the project to consume this contributed data, we are requesting it be provided via a Google form.

DEADLINE: Data must be submitted by July 20, 2016 (Extended to July 31).

You are invited to fill out the form [OWASP Top 10 - 2016 Data Call](#) if you wish to submit your organization's data to the project. To help you prepare for your submission, all the questions are listed on the **OWASP Top 10 - 2016 Data Call Questions** tab, here in the wiki.

WARNING: All contributed data will be made public. DO NOT CONTRIBUTE anything you don't want to become publicly available.

OWASP Top 10

The OWASP Top Ten is a powerful awareness document for web application security. The OWASP

What is the OWASP Top 10?

The OWASP Top 10 provides:

- A list of the 10 Most Critical Web Application Security Risks

And for each Risk it provides:

- A description
- Example vulnerabilities
- Example attacks
- Guidance on how to avoid
- References to OWASP and other related resources

Project Leader

- Dave Wichers

Quick Download

- [OWASP Top 10 2013 - PDF](#)
- [OWASP Top 10 2013 - wiki](#)
- [OWASP Top 10 2013 Presentation - Covering Each Item in the Top 10 \(PPTX\)](#)

Email List

[Project Email List](#)

News and Events

- [20 May 2016] OWASP Top 10 - 2016 Data Call Announced

- Home
- About OWASP
- Acknowledgements
- Advertising
- AppSec Events
- Books
- Brand Resources
- Chapters
- Donate to OWASP
- Downloads
- Funding
- Governance
- Initiatives
- Mailing Lists
- Membership
- Merchandise
- News
- Community portal
- Presentations
- Press
- Projects
- Video
- Volunteer

- Reference
 - Activities
 - Attacks
 - Code Snippets
 - Controls
 - Glossary

OWASP: Top 10 Security Risks

A1-Injection

Injection flaws, such as SQL, OS, and LDAP injection occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

A2-Broken Authentication and Session Management

Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities.

A3-Cross-Site Scripting (XSS)

XSS flaws occur whenever an application takes untrusted data and sends it to a web browser without proper validation or escaping. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.

A4-Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.

A5-Security Misconfiguration

Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server, and platform. Secure settings should be defined, implemented, and maintained, as defaults are often insecure. Additionally, software should be kept up to date.

OWASP: Top 10 Security Risks

A6-Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data deserves extra protection such as encryption at rest or in transit, as well as special precautions when exchanged with the browser.

A7-Missing Function Level Access Control

Most web applications verify function level access rights before making that functionality visible in the UI. However, applications need to perform the same access control checks on the server when each function is accessed. If requests are not verified, attackers will be able to forge requests in order to access functionality without proper authorization.

A8-Cross-Site Request Forgery (CSRF)

A CSRF attack forces a logged-on victim's browser to send a forged HTTP request, including the victim's session cookie and any other automatically included authentication information, to a vulnerable web application. This allows the attacker to force the victim's browser to generate requests the vulnerable application thinks are legitimate requests from the victim.

A9-Using Components with Known Vulnerabilities

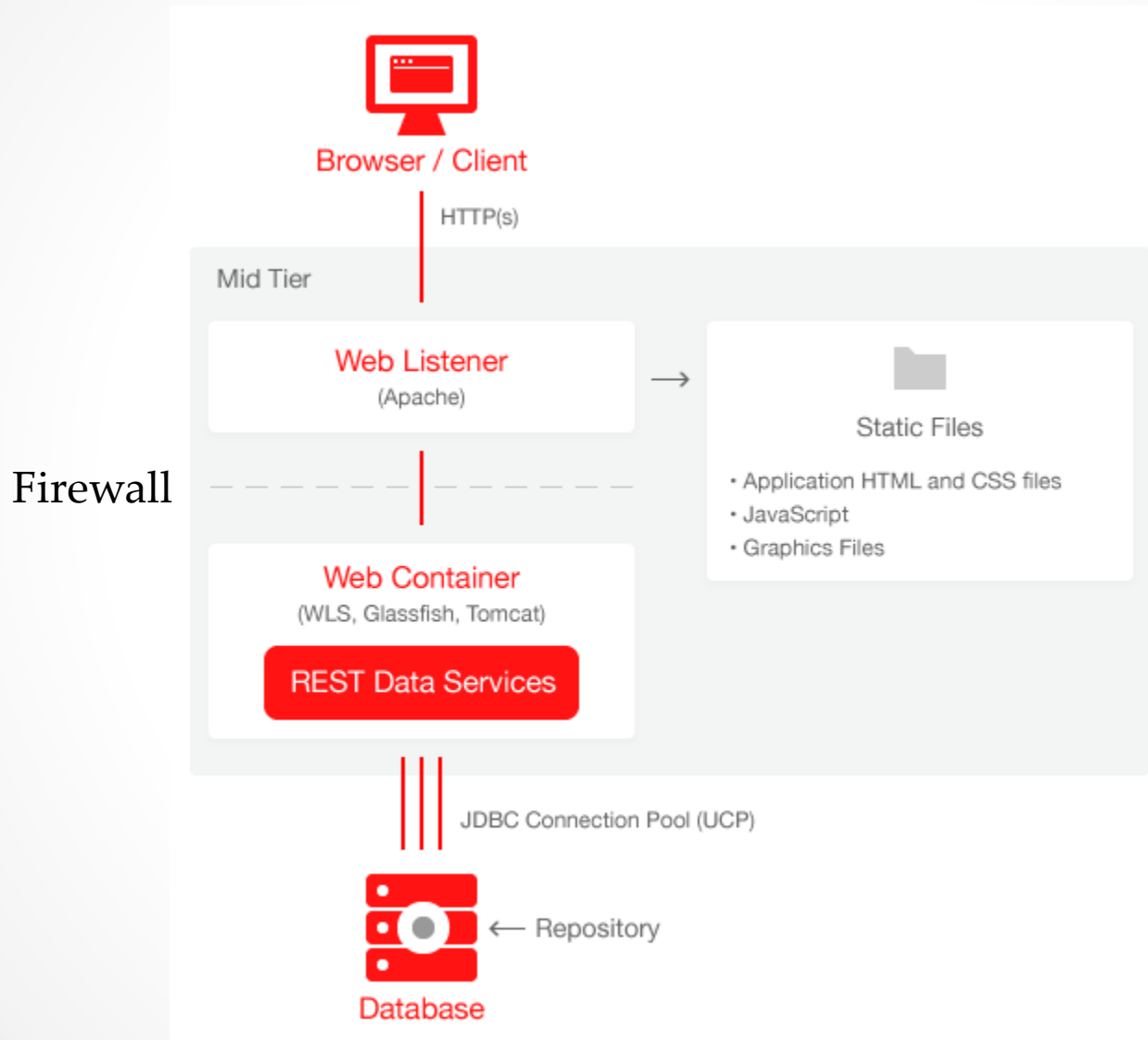
Components, such as libraries, frameworks, and other software modules, almost always run with full privileges. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications using components with known vulnerabilities may undermine application defenses and enable a range of possible attacks and impacts.

A10-Invalidated Redirects and Forwards

Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.

ORDS

Architecture



Installation & Configuration

- Command line

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

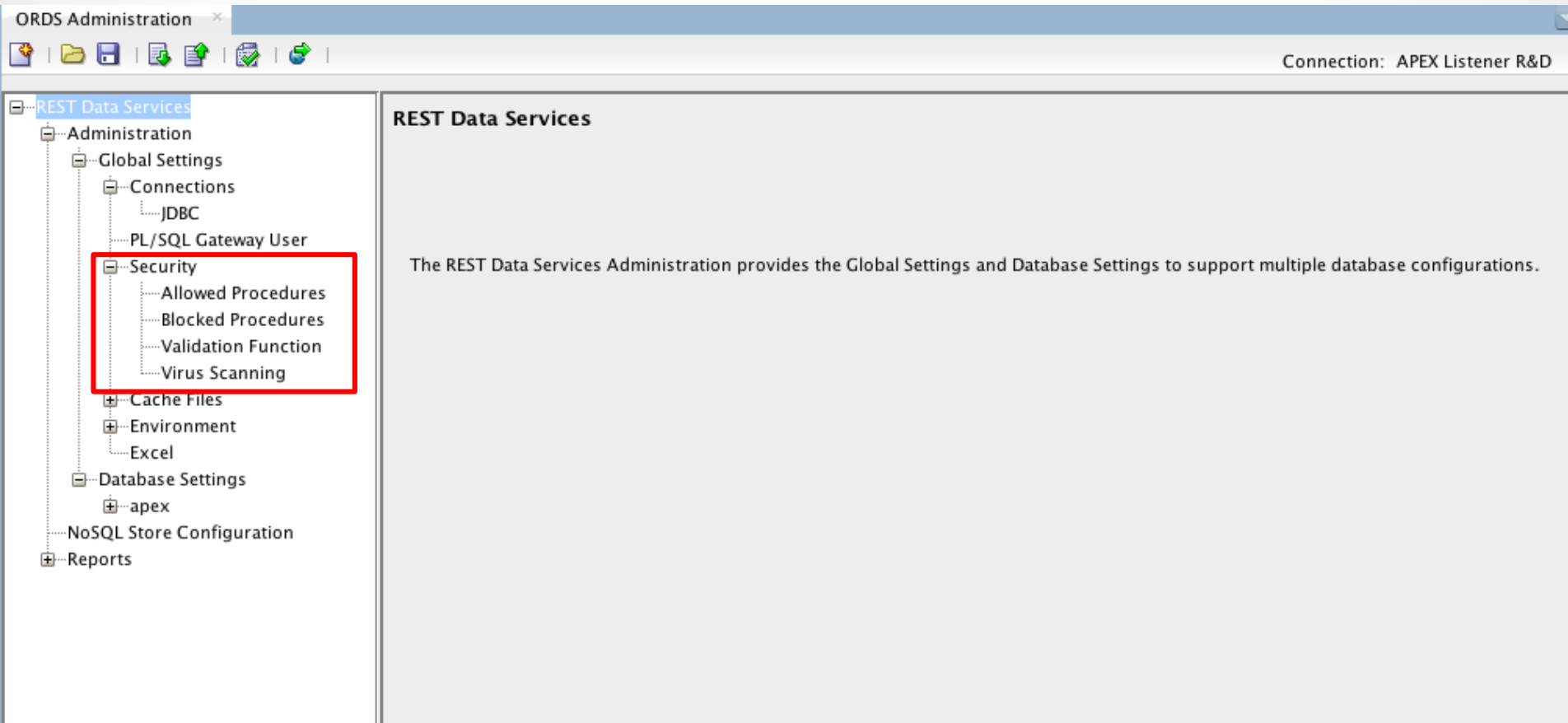
```
java -jar ords.war install advanced
```

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"
```



The screenshot shows the ORDS Administration console interface. The title bar indicates the connection is to 'APEX Listener R&D'. The left-hand navigation pane displays a tree structure under 'REST Data Services'. The 'Security' folder is highlighted with a red rectangle and contains the following sub-items: 'Allowed Procedures', 'Blocked Procedures', 'Validation Function', and 'Virus Scanning'. The main content area on the right is titled 'REST Data Services' and contains the text: 'The REST Data Services Administration provides the Global Settings and Database Settings to support multiple database configurations.'

Installation & Configuration

- PL/SQL Validation Function

The screenshot shows the ORDS Administration console interface. The title bar reads "ORDS Administration" and the connection is identified as "APEX Listener R&D". The left-hand navigation pane is expanded to show the "Administration" section, with "Global Settings" selected. Under "Global Settings", the "Validation Function" option is highlighted in blue. The main content area is titled "Validation Function - Global Settings". It features a "Validation Function Type:" label followed by a dropdown menu currently set to "PL/SQL". Below this, a descriptive text states: "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 "Validation Function:" label and 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

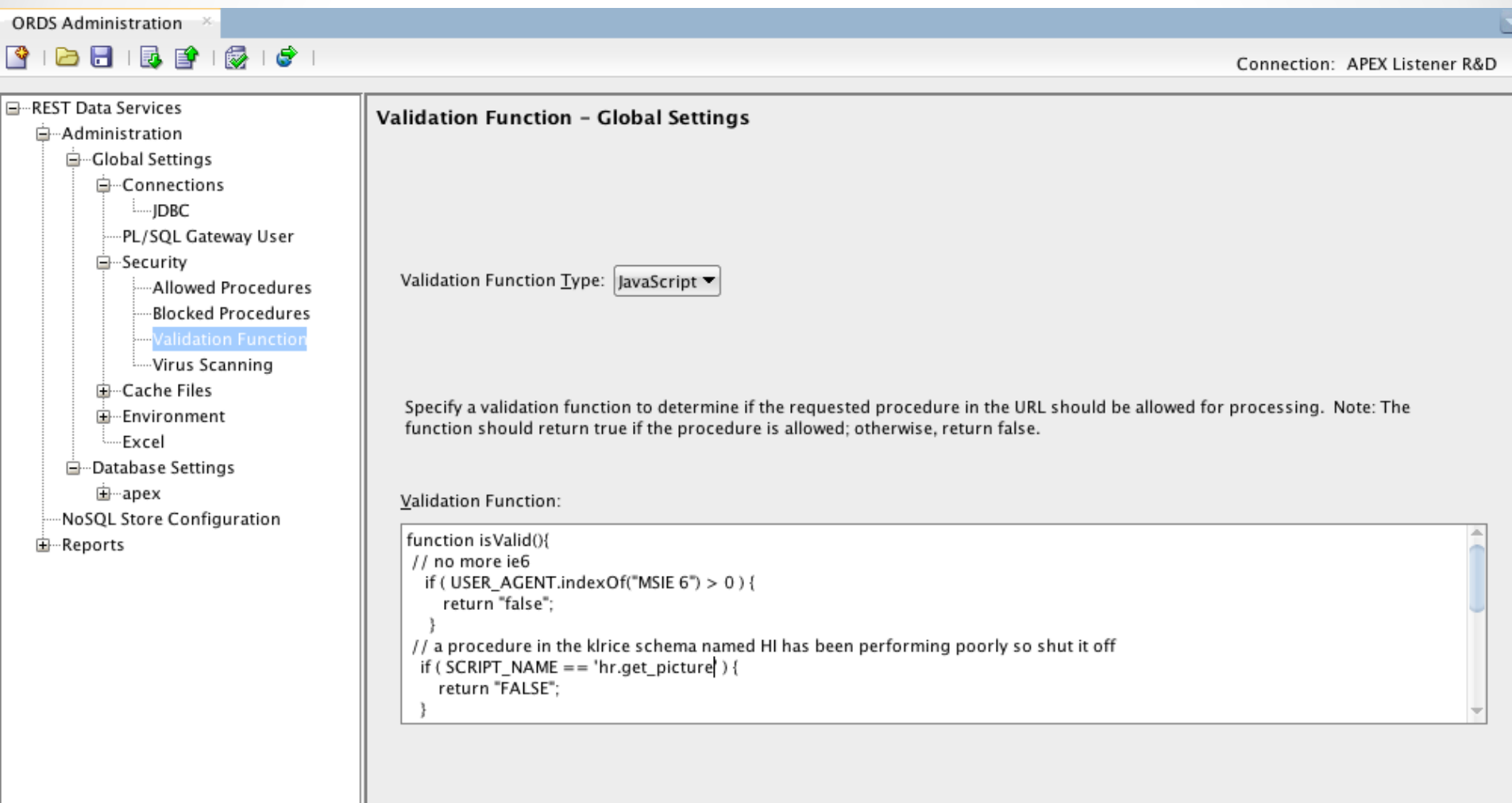
Validation Function Type:

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:

Validation Function

- PL/SQL vs JavaScript Validation Function



The screenshot shows the ORDS Administration console interface. The left-hand navigation pane is expanded to show the 'Validation Function' option under the 'Global Settings' section. The main content area is titled 'Validation Function - Global Settings'. It features a dropdown menu for 'Validation Function Type' set to 'JavaScript'. Below this, there is a descriptive text: '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 text area for the 'Validation Function' code, which 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 x

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

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  
        www\_flow\_epg\_include\_modules.authorize(p_procedure);  
        return true;  
    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)

Secure REST web services

- REST web services with OAuth2

Build-in Webserver

- ORDS \geq 3.0.0; improved build-in web server (Jetty)

Workspace and Application Settings

Workspace and Application Administration

In an Oracle Application Express development environment, users log in to a shared work area called a **workspace**. A workspace is a virtual private database that enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data and applications private. This flexible architecture enables a single database instance to manage thousands of applications.



Workspace and Application Administration

Developers can create and edit applications and view developer activity, session state, workspace activity, application, and schema reports. Workspace administrators additionally can create and edit user accounts, manage groups, and manage development services.

Workspace and Application Administration

- **Developers** create and edit applications.
- **Workspace administrators** are developers who also perform administrator tasks specific to their workspace such as managing user accounts, monitoring workspace activity, and viewing log files. See "Workspace and Application Administration".
- **Oracle Application Express administrators (instance administrators)** are super users that are responsible for managing an entire Oracle Application Express instance. Instance administrators manage workspace provisioning, configure features and instance settings, and **manage security**.

Internal workspace



Oracle Application Express



internal



ADMIN



.....



Sign In

[Reset Password](#)

Instance administration



Manage Requests



Manage Instance

Create Workspace >



Manage Workspaces



Monitor Activity

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? | <input type="text" value="No"/> | ? |
| * Schema Name | <input type="text" value="DEMOSEC"/> | ? |
| * Schema Password | <input type="password" value="*****"/> | ? |
| * Space Quota (MB) | <input type="text" value="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 x

SQL | All Rows Fetched: 0 in 0.03 seconds

| GRANTEE | OWNER | TABLE_N... | GRANTOR | PRIVILEGE | GRANTA... | HIERARC... |
|---------|-------|------------|---------|-----------|-----------|------------|
|---------|-------|------------|---------|-----------|-----------|------------|

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

Instance Settings - Wallet

- A wallet is a password-protected container that is used to store authentication and signing credentials. The Oracle wallet is used for all HTTP requests over Secured Socket Layer (SSL), namely HTTPS.

Wallet Path ?

Wallet Password ?

Check to confirm that you wish to change the wallet password

Security

Security

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

| | | | | |
|-------------------------------|---|----------------------------------|----------------------------------|----------------------------------|
| Set Workspace Cookie | <input type="text" value="No"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |
| Disable Administrator Login | <input type="text" value="Yes"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |
| Disable Workspace Login | <input type="text" value="Yes"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |
| Allow Public File Upload | <input type="text" value="No"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |
| Restrict Access by IP Address | <input type="text"/> | | | <input type="button" value="?"/> |
| Instance Proxy | <input type="text"/> | | | <input type="button" value="?"/> |
| Checksum Hash Function | <input type="text" value="Most Secure"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |
| Rejoin Sessions | <input type="text" value="Enabled for All Sessions"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |
| Unhandled Errors | <input type="text" value="Return HTTP 400"/> | <input type="button" value="⌵"/> | <input type="button" value="⌶"/> | <input type="button" value="?"/> |

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



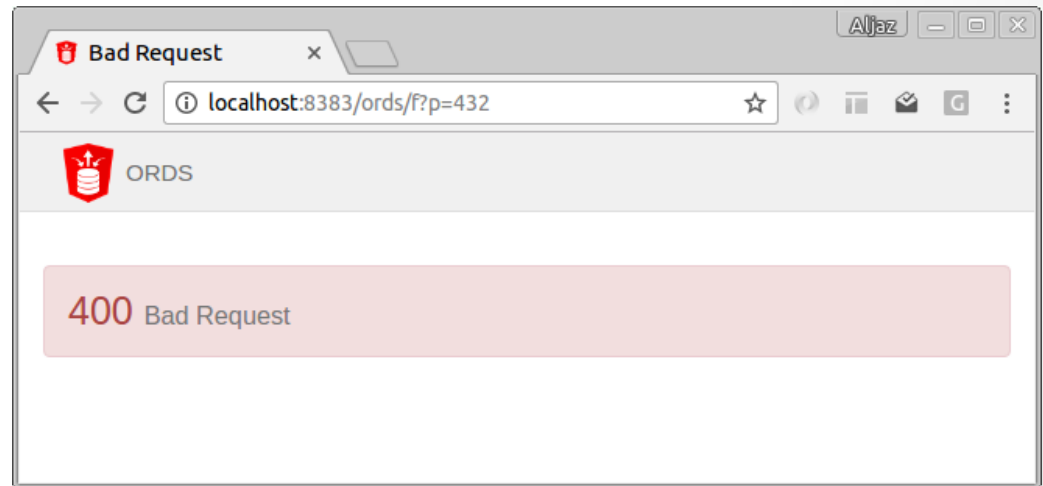
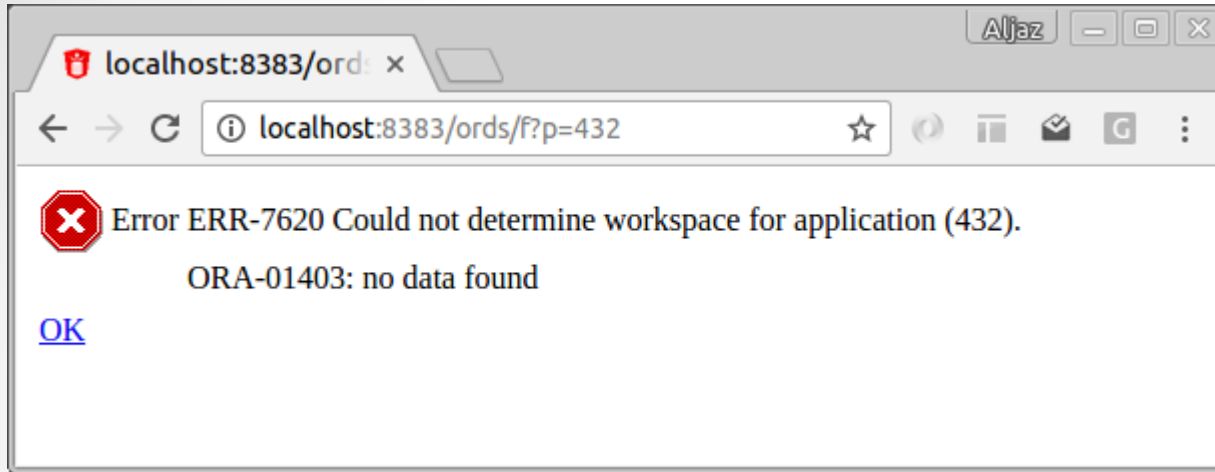
Workspace isolation

- 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

Unhandled Errors



Unhandled Errors

The image shows two overlapping browser windows. The background window, titled "Bad Request", displays a 400 Bad Request error from ORDS. The foreground window, titled "Home 3", shows a modal dialog box with a red X icon and the following text:

Session state protection violation:
This may be caused by manual alteration of a URL containing a checksum or by using a link with an incorrect or missing checksum. If you are unsure what caused this error, please contact the application administrator for assistance.

Contact your application administrator.

OK

HTTP protocol

- Require HTTPS
- Require Outbound HTTPS
- HTTP Response Headers
 - Content-Security-Policy
 - X-XSS-Protection: '1; mode=block'
 - X-Content-Type-Options: 'nosniff'
 - <https://scotthelme.co.uk/hardening-your-http-response-headers/>
 - <https://securityheaders.io/>
 - Error parsing header X-XSS-Protection: '1; mode=block': expected 0 or 1 at character position 0. The default protections will be applied.

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

Application Security Attributes

- Authentication
 - Public User (USER_IS_PUBLIC_USER, USER_IS_NOT_PUBLIC_USER, APEX_APPLICATION.G_PUBLIC_USER)
- Authorization
 - Authentication Scheme
- 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
- Deep Linking
- Maximum Session Length in Seconds (0, >0, empty)



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

Authentication & Authorization

Authentication Control

- Manage security settings for developer and end user login.
- After failed logins, Application Express will display a countdown of this number times the number of failed login attempts, before it accepts new login attempts with the same username.
- Enter 0 to disable the countdown and allow immediate access.
- Applies to all Authentication Schemes
-

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
- Depending on the selected Authentication Scheme Type, you can use various settings and program hooks to fine-tune your application's authentication



Authentication

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

Custom authentication

| | | |
|---|---|---|
| Sentry Function Name | <input type="text"/> | ? |
| Invalid Session Procedure Name | <input type="text"/> | ? |
| Authentication Function Name | <input type="text" value="my_custom_function"/> | ? |
| Post Logout Procedure Name | <input type="text"/> | ? |
| Enable Legacy Authentication Attributes | <input type="text" value="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;
```

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;
```

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 |
```

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.
- Secure: YES
 - Allow the session management cookie to be sent from the browser only when the protocol is HTTPS
- Sharing authentication across multiple APEX applications

Authorization Schemes

- Authorization schemes enable you to protect applications, pages, and application components (region, button, item, processes, ...)
- Common authorization scheme types include Exists, Not Exists SQL Queries, and PL/SQL Function Returning Boolean

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

The screenshot shows a web browser window with two tabs: "Application Process" and "Home". The address bar shows the URL `localhost:8383/ords/f?p=152:1:25039627603929:::`. The page content includes a blue header with "Authorization and authentication" and "Home" links, and a main area with the text "Home". An alert dialog box is open in the center, titled "localhost:8383 says:", with the message "Hey, all rows deleted. See you :)" and an "OK" button. Below the dialog, there is a checkbox labeled "Prevent this page from creating additional dialogs." and a "Log Out" link in the top right corner. The developer console is open at the bottom, showing the following JavaScript code:

```
> apex.server.process (
  "TEST_AP", {}
  ,{ dataType: 'text',
    success: function(pData){alert(pData)}
  }
);
< ▶ Object {readyState: 1}
▶ XHR finished loading: POST "http://localhost:8383/ords/wwv_flow.show". jquery-2.1.3.min.js?v=5.0.4.00.12:4
>
```

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

VPD, RAS, Shadow Schema

Challenges on Data Access Control

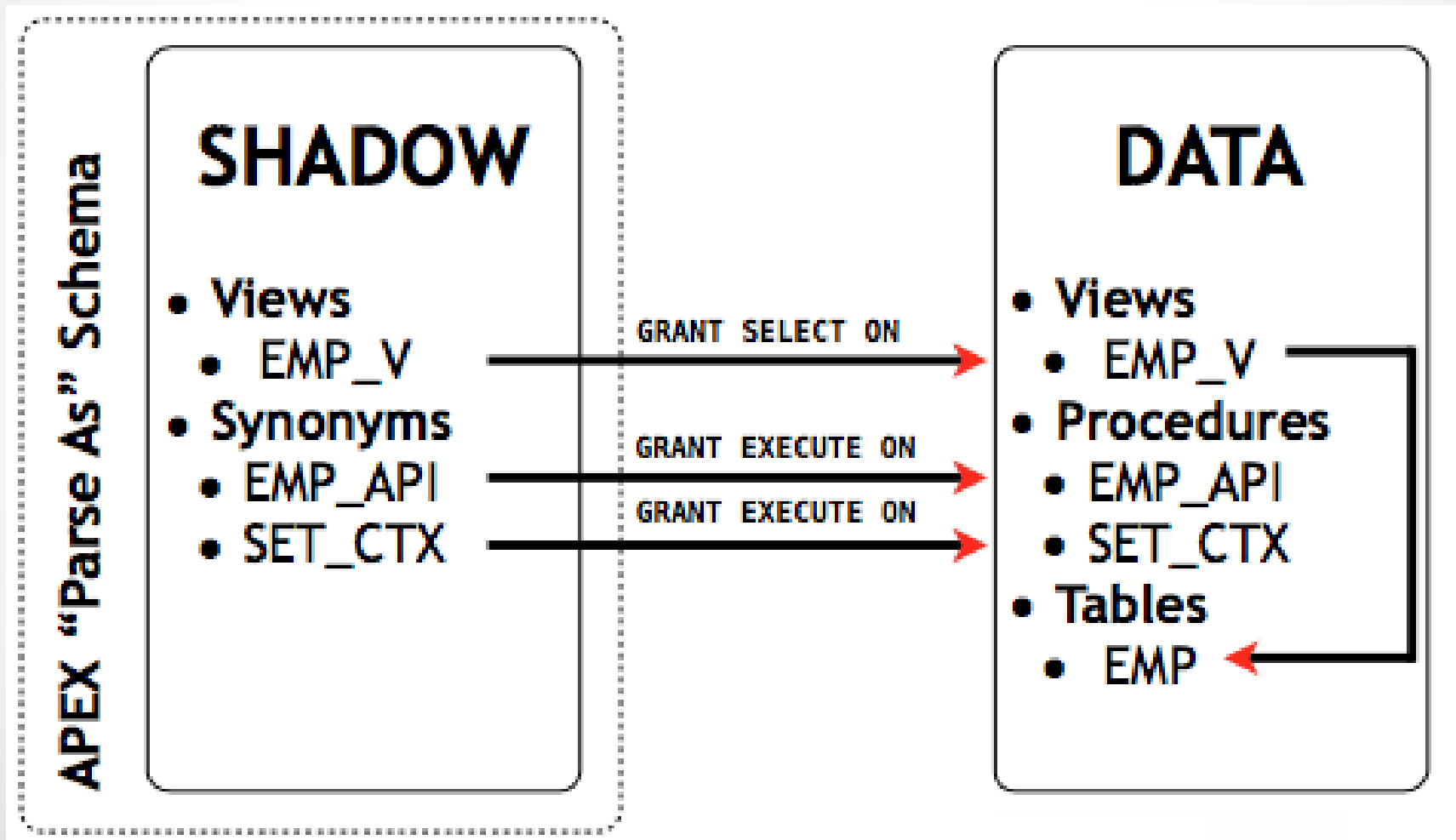
- Code executed under privileged user
- Database unaware of end users
- Data access policy (data security) is hard coded in
 - Where-clause - application level
 - Views - database level
 - Shadow Schema
 - Virtual Private Database (VPD)
 - database level
 - Real Application Security (RAS)
 - database level



Shadow Schema

- Schema with as little privileges as possible
- No objects with data
- Workspace and applications that have to be secure linked to this schema
- “White Listing”: don’t allow anything, except for ...

Shadow Schema



APEX implications

- See data: Secure views
 - PL/SQL Initialisation code (!)
- DML data: Automatic Row Processing only possible with instead of triggers or use of custom PL/SQL APIs
 - API Generation in SQL Workshop (!)

VPD

- No-cost feature of Oracle Enterprise Edition Database
- VPD dynamically adds a WHERE clause
- Defined by Policies
- DBMS_RLS package

Benefits VPD

- Secures data at the database layer
 - Works regardless of the technology used to access the table
 - You can put the VPD policies in a separate schema so that it is isolated from your developers
- Simplified development
 - No need to add a where clause everywhere
- Can be applied to columns (show null for some columns, or only hide rows when a column is in, ...)
- Use of application context
(in APEX use PL/SQL Initialisation code)

VPD Policy Function

```
FUNCTION emp_vpd
  (p_schema IN VARCHAR2 DEFAULT NULL,
   p_object IN VARCHAR2 DEFAULT NULL)
RETURN VARCHAR2
AS
  l_where      VARCHAR2(1000);
  l_app_user   VARCHAR2(255) := v('APP_USER');
  l_g_deptno   NUMBER       := nv('G_DEPTNO');
BEGIN
  IF l_app_user IS NOT NULL AND l_g_deptno IS
NOT NULL THEN
    l_where := 'deptno = ' || l_g_deptno;
  END IF;
  RETURN l_where;
END;
/
```

DBMS_RLS

DBMS_RLS.add_policy

```
(object_schema => :OWNER,  
 object_name   => 'EMP' ,  
 policy_name   => 'EMP_VPD' ,  
 function_schema => :OWNER,  
 policy_function => 'EMP_PKG.EMP_VPD' ,  
 statement_types => 'SELECT' );
```

DBMS_RLS.drop_policy

```
(object_schema => :OWNER,  
 object_name   => 'EMP' ,  
 policy_name   => 'EMP_VPD' );
```

Real Application Security (RAS)

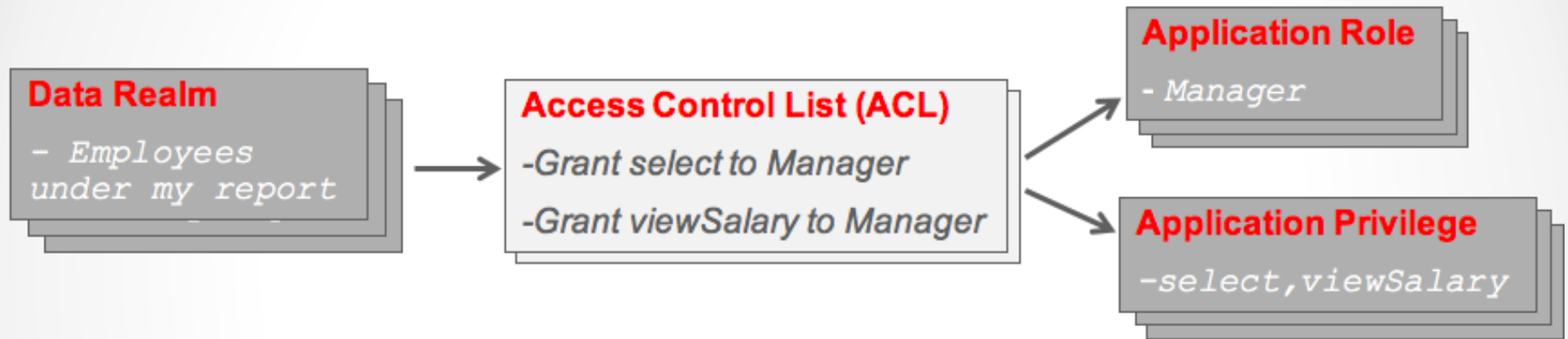
A database authorisation solution for end-to-end application security

RAS Key features

- Support Application Users and Sessions
 - Schema-less user, security and application context in DB
- Support Application Privileges and Roles
- Support fine-grained data access control on rows and columns
 - Based on user operation execution context
 - Enforce security close to data



RAS Concepts: Policy components



- Data Security policy is a collection of Data Realms and ACLs
- Each Data Realm has an associated ACL with grants

RAS: setup with PL/SQL API

```
xs_principal.create_role(name => 'emp_role',  
enabled => true);
```

```
xs_security_class.create_security_class(  
    name          => 'hr.hrprivs',  
    parent_list  => xs$name_list('sys.dml'),  
    priv_list    =>  
xs$privilege_list(xs$privilege('view_salary'))  
;
```


RAS Administration Tool

The screenshot displays the RAS Administration Tool interface for configuring a policy. The breadcrumb trail is Home > Policies > Policy Definition. The main section is titled 'Policy' and contains the following fields:

- Policy Name:** HRM.EMPLOYEE_POLICY
- Description:** Policy for Employee Records
- Protected Objects:** HRM.EMPLOYEES

Buttons for 'Cancel', 'Delete', and 'Apply Changes' are located at the top right of the policy configuration area.

The 'Data Realm Authorization' section contains a table with the following data:

| Realm Description | SQL Predicate | ACL | Reorder |
|--|--|-----------------------------------|---------|
| <input type="checkbox"/> ALL RECORDS | 1=1 | HRM.ALL_EMP_ACL | ▲ ▼ |
| <input type="checkbox"/> MY_RECORD | EMPLOYEE_ID IN (SELECT EMPLOYEE_ID FROM HRM.USER_PROFILE WHERE LOGON_NAME = XS_SYS_CONTEXT('XSSSESSION','USERNAME')) | HRM.MY_EMP_ACL | ▲ ▼ |
| <input type="checkbox"/> MY_REPORTS | EMPLOYEE_ID IN (SELECT EMPLOYEE_ID FROM (SELECT EMPLOYEE_ID, level M FROM HRM.MANAGERS M START WITH M.EMPLOYEE_ID IN (SELECT ... | HRM.MY_REPORT_ACL | ▲ ▼ |

Buttons for 'Delete' and 'Add' are located at the top right of the Data Realm Authorization section.

The 'Column Authorization' section contains a table with the following data:

| Column | Privilege | Description |
|---------------------------------|-------------|-----------------------|
| <input type="checkbox"/> SALARY | VIEW_SALARY | To view Salary column |
| <input type="checkbox"/> SSN | VIEW_SSN | To view SSN column |

Buttons for 'Delete' and 'Add' are located at the top right of the Column Authorization section.

Employees Table

- 1. All records
- 2. My record
- 3. My reports

Restricted Salary & SSN Columns

Privilege Grants

Real Application Security Features

Controlled Delegation

- VP delegating calendar management function to an Assistant

Effective-date support

- Contractor getting access for a specific duration

Negative grants

- Access to certain reports allowed only on intranet

Code-based security

- Batch programs with elevated privileges to summarize data

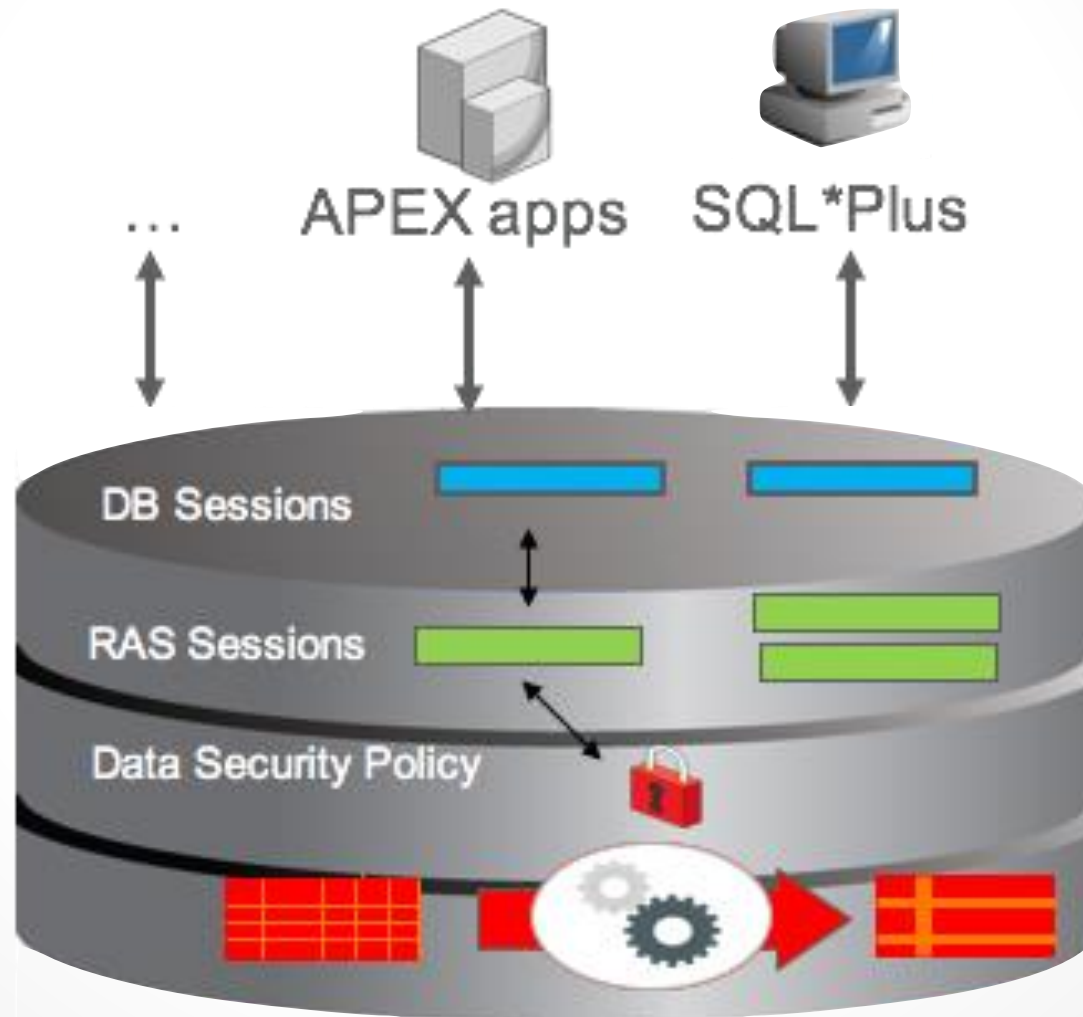
Function Security

- Conditional rendering of User Interface

Auditing

- Application users, privileges, roles are known to database

RAS Architecture

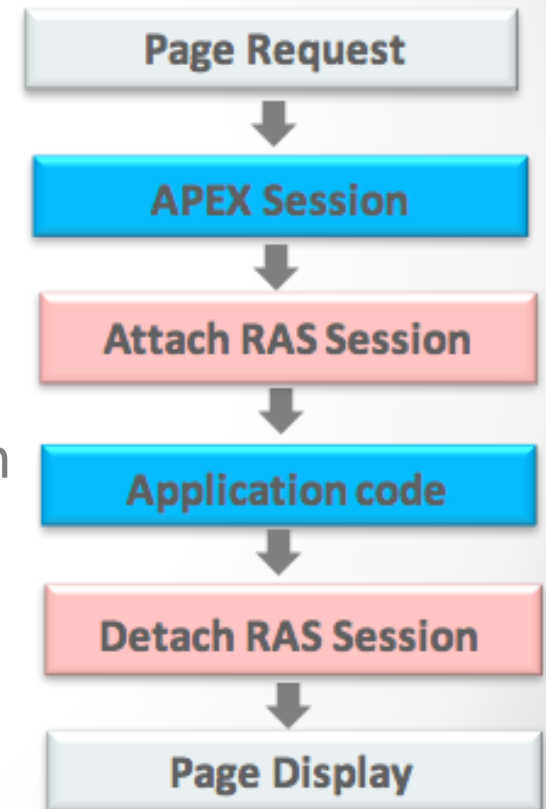


RAS in APEX



RAS Integration with APEX

- Application users continue to be provisioned in the database or identity stores
 - User authentication remains in APEX
- RAS session contains application user, its roles, and session context
 - Based on APEX user's security context
- Application code executes within RAS session
 - Attached and detached to a db session



RAS Integration with APEX 5

- APEX can use RAS users, roles, and data security policy
 - Instead of custom authorization using VPD
- RAS Session is transparently created based on APEX session
- For APEX authorization schemes, use RAS ACL check operators

Authentication Scheme

| Show All | Name | Subscription | Source | St |
|----------|------|--------------|--------|----|
|----------|------|--------------|--------|----|

Real Application Security

RAS Mode:

Authorization Scheme

* Scheme Type: PL/SQL Function Returning Boolean

* PL/SQL Function Body

```
declare
result pls_integer;
begin
select ORA_CHECK_ACL(:UPD_CHECK_ACLIDS, 'UPDATE') into result;
if result = 1 then
return true;
end if;
end;
```

RAS Benefits

- Stronger security
 - Enforced regardless of entry points: direct, APEX, or middleware
 - Audit end-user activity in database audit trail
- Simpler development
 - Declarative policy, relieves writing authorization code
 - Native support for application roles, application privileges, application users
- High Performance Access Control
 - Optimized for typical data access patterns within core database
- Simpler administration
 - Centralized management, end-to-end uniform security across mid-tier and database

RAS - to know...

- One RAS repository for the whole database
- Takes a bit of time to get used to the implementation and naming
- RASADM can help, but ...
 - RASADM doesn't expose all features
 - RASADM app didn't always behave as expected (had to patch it to get some things working)
- Once you enable RAS make sure to test your app (!)
APEX Advisor can't check for the correct grants (yet).

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



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

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.

Escaping

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



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

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

Log Out

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

Create

Xss

1 - 1

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

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

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>,<b>,</b>,<strong>,</strong>,<i>,</i>,<em>,</em>,<ul>,</ul>,<ol>,</ol>,<li>,</li>,<dl>,</dl>,<dt>,</dt>,<dd>,</dd>,<pre>,</pre>,<code>,</code>,<br />,<br />,<br>,<BR>,<hr />
```

XSS protection

- 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

APEX and XSS

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





Application items

- Referencing items in HTML regions
- Page items always escaped
- You have escape application items manually


Page items

▼ Identification

Title

Type  

▼ Source

PL/SQL Code 

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

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

Application items

▼ Identification

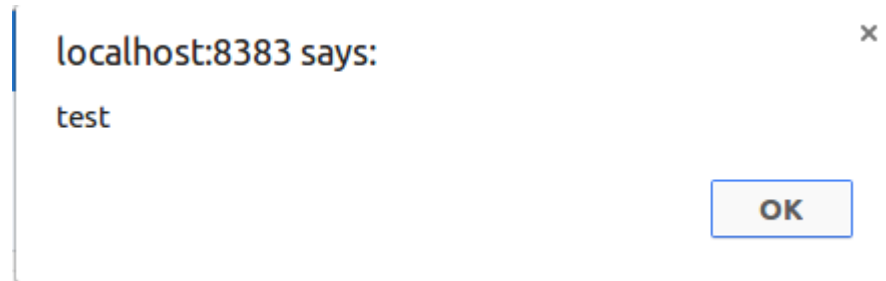
Title

Type ⌵ ☰

▼ 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

Session state protection

- Enabling Session State Protection can prevent hackers from tampering with URLs within your application
- URL tampering can adversely affect program logic, session state contents, and information privacy.
- When enabled, Session State Protection uses the Page Access Protection attributes and the Session State Protection item attributes with checksums positioned in f?p= URLs to prevent URL tampering and unauthorized access to and alteration of session state



Session state protection

- You can enable session state protection from either the Edit Security Attributes page or the Session State Protection page
- Enabling Session State Protection is a two-step process.
 - First, you enable the feature.
 - Second, you set page and item security attributes

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
- Checksum Required: User Level
- Checksum Required: Session Level
-

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.



PREPARE_URL Function

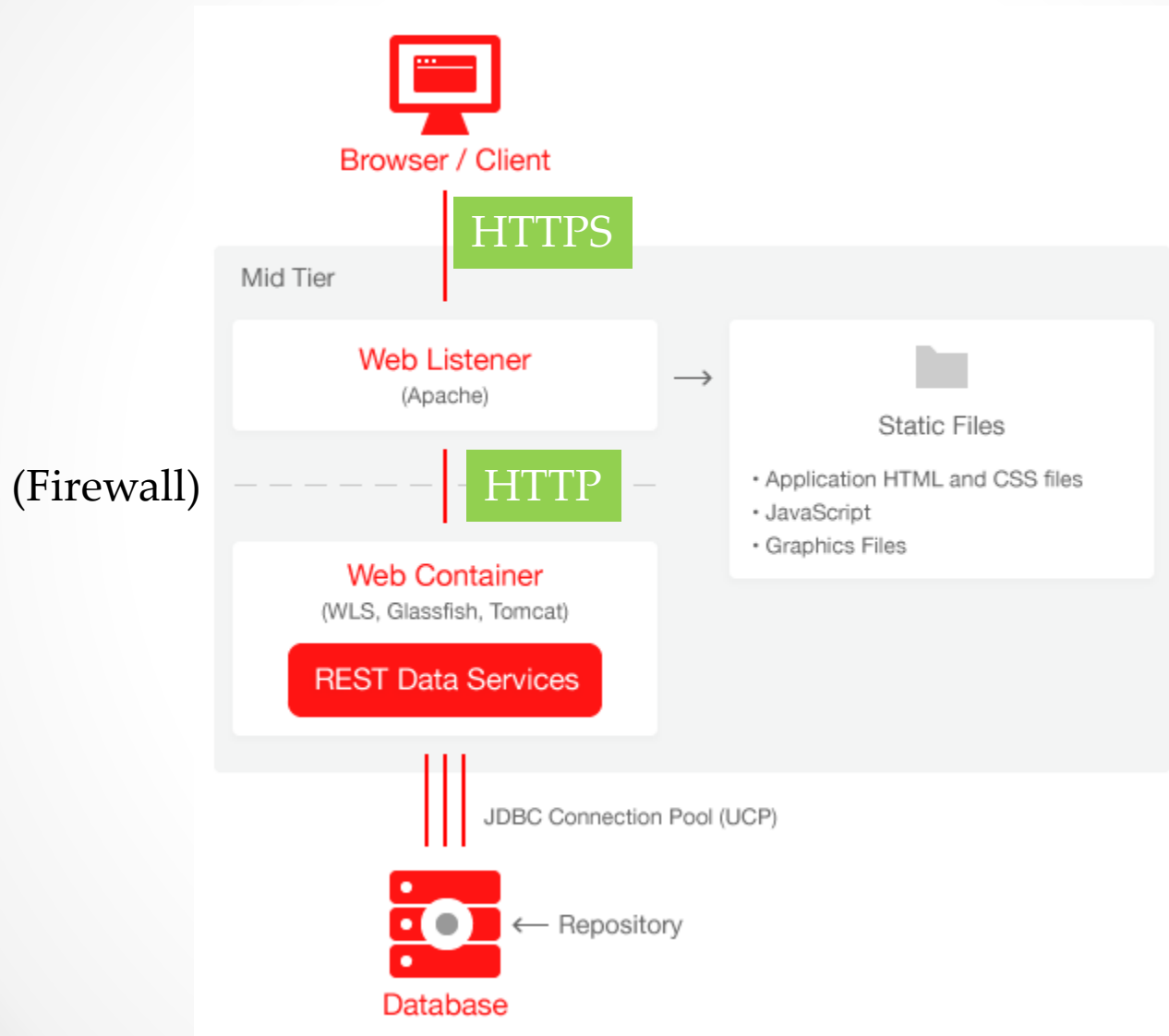
```
APEX_UTIL.PREPARE_URL (  
  p_url      IN VARCHAR2,  
  p_url_charset  IN VARCHAR2 default null,  
  p_checksum_type IN VARCHAR2 default null)  
RETURN VARCHAR2;
```

```
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

SSL & Reverse Proxy

Architecture



Why HTTPS?

- HTTPS protects the integrity of your website/APEX app
- HTTPS protects the privacy and security of your users
- HTTPS is the future of the web; many new technologies only work with HTTPS (for example Service Workers)
-

Use of HTTPS

- Getting a certificate (SSL)
- Webserver side configuration
- APEX side - optional
(limit apps to only use https)

Free SSL certificates (HTTPS)



LINUX FOUNDATION COLLABORATIVE PROJECTS

[Documentation](#)

[Get Help](#)

[Donate](#) ▾

[About Us](#) ▾

Let's Encrypt is a new Certificate Authority:
It's free, automated, and open.

[Get Started](#)

• <https://letsencrypt.org> •

Reverse Proxy

- A reverse proxy can act as a gateway service allowing access to servers on your trusted network from an external network.

Reverse Proxy Benefits

- Give APEX a nice URL
- Use HTTPS to the outside world
- Access other sites over HTTPS
- Performance

Use of Reverse Proxy

```
ProxyRequests Off
```

```
Order deny,allow
```

```
Allow from all
```

```
ProxyPreserveHost On
```

```
ProxyPass /apex ajp://localhost:8009/apex
```

```
ProxyPassReverse /apex ajp://localhost:8009/apex
```

```
ProxyPass /i ajp://localhost:8009/i
```

```
ProxyPassReverse /i ajp://localhost:8009/i
```

Use of Reverse Proxy

```
<VirtualHost *:80>  
ServerName support.apexrnd.be  
ServerAlias www.support.apexrnd.be  
ErrorLog /var/log/httpd/support_error_log  
CustomLog /var/log/httpd/support_custom_log combined  
DocumentRoot /var/www  
ProxyPreserveHost On  
ProxyPass / http://127.0.0.1:8088/  
ProxyPassReverse / http://127.0.0.1:8088/  
</VirtualHost>
```

Use of Reverse Proxy

```
<VirtualHost *:443>
ServerName xior.rentmaster.be
ServerAlias xior.rentmaster.be
ErrorLog /var/log/httpd/xior_s_error_log
LogFormat "%{X-Forwarded-For}i\" %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-agent}i\"" combined-elb
CustomLog /var/log/httpd/xior_s_custom_log combined-elb
DocumentRoot /var/www/rentmaster

SSLEngine On
SSLCertificateFile /etc/letsencrypt/live/rentmaster.be/cert.pem
SSLCertificateKeyFile /etc/letsencrypt/live/rentmaster.be/privkey.pem
SSLCertificateChainFile /etc/letsencrypt/live/rentmaster.be/chain.pem

ProxyRequests Off
ProxyPreserveHost On
RewriteEngine On
RewriteRule ^/$ /ords/f?p=XIOR:LOGIN:0 [R=301,L]
</VirtualHost>
```

Use of Reverse Proxy

```
<VirtualHost *:80>
ServerName apexrnd.localdomain
ServerAlias apexrnd.localdomain
RewriteEngine On
ProxyVia On
ProxyRequests Off
# Facebook
SSLProxyEngine On
ProxyPass          /graph.facebook.com/    https://graph.facebook.com/
ProxyPassReverse   /graph.facebook.com/    https://graph.facebook.com/
# Google
ProxyPass          /www.google.com/    https://www.google.com/
ProxyPassReverse   /www.google.com/    https://www.google.com/
# APEX.ORACLE.COM
ProxyPass          /apex.oracle.com/    https://apex.oracle.com/
ProxyPassReverse   /apex.oracle.com/    https://apex.oracle.com/
# APEX Office Print
ProxyPass          /aop/    https://www.apexofficeprint.com/api/
ProxyPassReverse   /aop/    https://www.apexofficeprint.com/api/
</VirtualHost>
```


Not interested in managing?



< Cloud

Get Started

What's New

Tasks

Administer Users

Manage Your Service

Connect to Your Service

Develop Document Store Applications Using SODA

Develop and Manage Web Services

Develop on the Database

Develop and Manage Applications

Learn About Application End User Tasks

Oracle Database Exadata Express Cloud Service

Get Started

Welcome to Oracle Database Exadata Express Cloud Service. With this service you get your own Oracle Database 12c Release 2 Enterprise Edition plus options running on Exadata in Oracle Cloud.



Learn About Your Service

What is Exadata Express?

How do I begin with Exadata Express?



Get a Subscription

Buy a subscription

Set up cloud users, administrators, and SFTP users



Get Started

Create and manage users

Access your service

Access the Exadata Express Service Console

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

