

Writing Secure APEX Applications

BGOUG - 2010

sumnēva

Application Success

About Dimitri Gielis



Dimitri Gielis 



<http://dgielis.blogspot.com>

dimitri@sumneva.com

About this presentation

- What it is:
 - APEX tools and features to make applications more secure
 - Common security holes
- What it is not:
 - An exhaustive or definitive list
 - Replacement for the documentation
 - Not a copy/paste exercise

What is security?

What is security?

- Very subjective
 - (your idea of secure might be different to mine)
- Different metrics
 - (based on time and effort)
- Proportional to the value of the data and consequences

What is security?

My definition:

People can only see/do the things they
should be allowed to see/do.

People cannot see/do the things they
should not be allowed to see/do.

What is security?

- Often an after-thought
- Should be considered from Day 0
- Don't just see the application as a developer, see it as a user/hacker
- It's **OUR** responsibility



Specific Topics

Specific Topics

- APEX Features
 - Authentication
 - Authorization
 - Session State Protection
- Malicious User Input
 - SQL Injection
 - XSS (Cross Site Scripting)
- Database Features

APEX Features

APEX Architecture

- Review Architecture
 - Server and Operating System
 - Database and Webserver
 - APEX Installation
- Use latest patch of APEX
- Use common development practices
- Use Advisor (APEX 4.0) and automated tests

Authentication



Authentication

- One time check
 - Pass or Fail
- Identify a user in the system
 - Used later for auditing

Authentication

Create Authentication Scheme

Cancel

< Previous

Next >

Select a preconfigured authentication scheme. Making a selection here creates an authentication scheme that follows a standard behavior for authentication and session management. Even though you are choosing a preconfigured authentication scheme, you can change any aspect of it later.

Gallery:

- ☒ Show Built-In Login Page and Use Open Door Credentials
- ☐ Show Login Page and Use Application Express Account Credentials
- ☐ Show Login Page and Use Database Account Credentials
- ☐ Show Login Page and Use LDAP Directory Credentials
- ☐ No Authentication (using DAD)
- ☐ Oracle Application Server Single Sign-On (Application Express Engine as Partner App)
- ☐ Oracle Application Server Single Sign-On (My Application as Partner App)

Authentication Gallery Schemes

- APEX Accounts (Cookie Users)
 - Uses built in APEX accounts
 - Easy to administer repository of users
 - Useful for testing purposes
 - Rarely used in production

Authentication Gallery Schemes

- Open Door
 - User prompted only for username
 - No verification of credentials
 - Any username is valid!
- Useful for testing purposes

Authentication Gallery Schemes

- Database Accounts
 - Uses real DB usernames/passwords
 - Does not affect the parsing schema

Authentication Gallery Schemes

- Custom Authentication
 - 100% Control over how it works
 - Authenticate users via
 - Tables
 - Web Services
 - ...

Authentication Gallery Schemes

- LDAP Directory
 - Use (in theory!) any LDAP compliant server
 - Microsoft Active Directory (AD)
 - Oracle Internet Directory (OID)
 - Sun iPlanet
 - OpenLDAP
 - Delegate responsibility elsewhere!

Authentication

D E M O N S T R A T I O N

Authorization

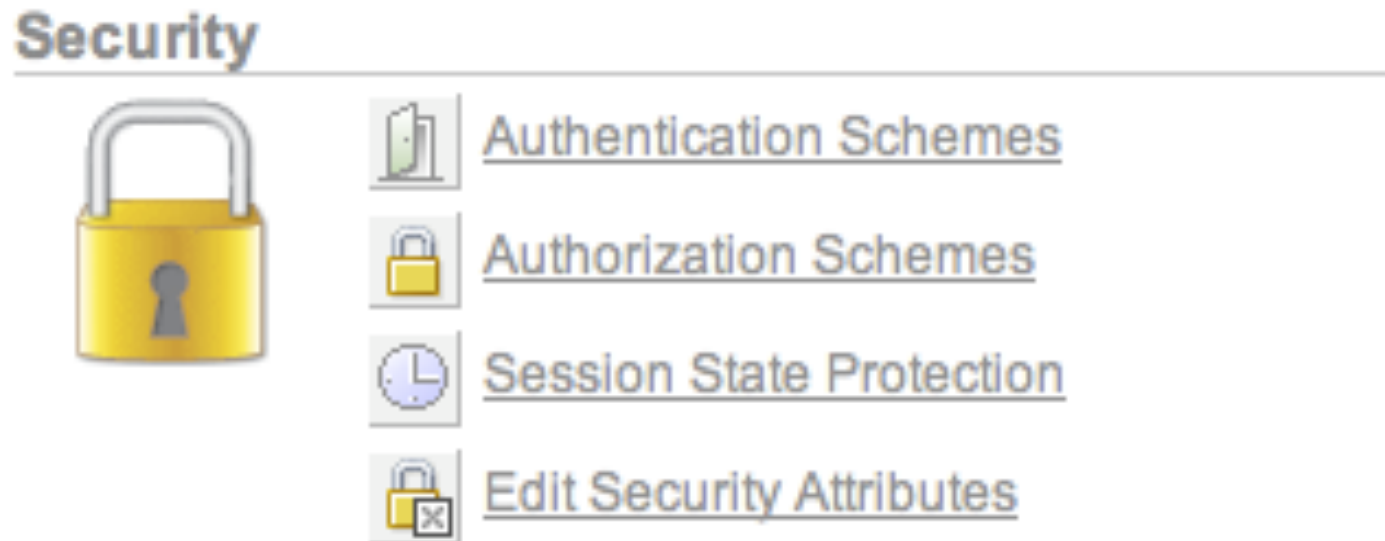


Authorization

- Multi time check
 - Pass or Fail
 - Contrast with Authentication
- Identify what a user can do (or not do)
 - User-specific view of the application/data
- Not the same as a condition!

Authorization

- Created as a Shared Component



- Can be copied from another application
 - Publish/Subscribe method very powerful !!!

Authorization

- Result of Authorization Scheme can be cached
 - Huge opportunity for performance/scalability
 - APP_USER rarely changes per page view

Evaluation Point

Validate authorization scheme

- ☒ Once per page view
- ☐ Once per session

- In APEX 4.0 Once per session is the default

Authorization

- Access Control List
- Predefined Roles

Application Administration

Set Application Mode

Application Mode

☒ Full access to all, access control list not used.

☐ Restricted access. Only users defined in the access control list are allowed.

☐ Public read only. Edit and administrative privileges controlled by access control list.

☐ Administrative access only.

Access Control List

DeleteApply Changes

Identify usernames which correspond to this application's authentication scheme.

Find Go

☐	<u>Username</u> ▲	<u>Privilege</u>	<u>Last Changed By</u>	<u>Date</u>
☐	ADMIN	Administrator ▼	admin	0 seconds ago
☐		Administrator ▼	(null)	(null)

1 - 2

Add User

Authorization

D E M O N S T R A T I O N

Session State Protection (SSP)



Session State Protection (SSP)

- Protects against end users tampering with the URL
 - Everybody can do that!
- Enable and configure SSP
 - Checksum passed along the URL
 - Checksum verified on page submission

Session State Protection (SSP)

D E M O N S T R A T I O N

Malicious User Input

General Validations



General Validations

- Where should validations be?
 - Server Side (PL/SQL, SQL)
 - Client Side (Javascript)

ANSWER: BOTH

General Validations

Don't trust client-side validations:

```
<script language="JavaScript" type="text/javascript">
  function ValidateTotal() {
    if ($v('P2_ORDER_TOTAL') == '') {
      alert('You must enter a total');
    };
  }
</script>
```

☐ Enable JavaScript

☒ Enable Java

Advanced...

Cross Site Scripting (XSS)



Cross Site Scripting (XSS)

- Code injection by a malicious user into the webpage which is viewed/executed by others
- Different forms of XSS
 - DOM-based - Manipulation DOM
 - Non-Persistent - Search result
 - Persistent - Data from DB

Cross Site Scripting (XSS)

D E M O N S T R A T I O N

Cross Site Scripting (XSS)

- Escape special characters when displaying user supplied input
 - Use `htf.escape`
 - `htf.escape_sc(:PI_DESCRIPTION);`
 - Use `dbms_assert`
 - `sql_object_name`
 - `enquote_name`
 - `enquote_literal`

SQL Injection



SQL Injection

- Incorrectly filtered user input used in an SQL operation leading to unintended side-effects

SQL Injection

D E M O N S T R A T I O N

Database Features

VPD

Database Features - VPD

- Virtual Private Database (VPD)
 - Enforces security at the database level
 - Obeyed regardless of client
 - Adds extra predicate information
 - PL/SQL function

Database Features - VPD

- Benefits
 - Separates security logic from application logic
 - Increases the degree of auditing
 - Different views of data depending on user
 - Easier to incorporate security logic changes
 - Across all applications/clients - not multiple places
 - Simplified maintenance
 - Shared policies across tables

Database Features - VPD

- Simple example: without VPD policy

```
SQL> select count(*) from demo_product_info;
```

```
      COUNT(*)  
-----  
              10
```

Database Features - VPD

- Simple example

```
SQL> create or replace function  
vpd_product_info(  
    p_schema in varchar2 default null,  
    p_object in varchar2 default null)  
    return varchar2 as  
begin  
    return ' (PRODUCT_AVAIL=' 'Y' ' ) ' ;  
end;
```

Database Features - VPD

- Simple example

```
begin
  dbms_rls.add_policy(
    object_schema => 'training',
    object_name   => 'demo_product_info',
    policy_name   => 'Product Info Policy',
    function_schema => 'training',
    policy_function => 'vpd_product_info',
    statement_types => 'select');
end;
```

Database Features - VPD

- Result: with VPD policy enabled

```
SQL> select count(*) from demo_product_info;
```

```
      COUNT ( * )  
-----  
              9
```

(We only see rows the policy allows us to see)

Automated Checks

APEX 4.0 Advisor

- Performs checks on your application
 - Programming Errors
 - Security Issues
 - QA and BP

ORACLE® Application Express

Welcome DIMITRI.GIELIS@APEX-EV

Home	Application Builder ▼	SQL Workshop ▼	Team Development ▼	Administration ▼
------	-----------------------	----------------	--------------------	------------------

Home > Application Builder > Application 12807 > Utilities > Advisor

Perform Check

Checks to Perform

☒ References with Substitution Syntax	☒ References with Column Syntax	☒ References with Bind Variable Syntax
☒ Referenced item is on current page	☒ Referenced item is page item of target page	☒ Declarative references of Application/Page Items/IR Filter
☒ References of Page Item in a String	☒ Referenced Page Number Exists	☒ Clear Cache Page Number equals Target- or Current Page
☒ Is Valid SQL/PLSQL Code	☒ Target Page Authorization equals current Component authorization	☒ Target Page Authorization is also set for current Component
☒ Fetch/DML/MR* Processes are valid	☒ Inappropriate use of Substitution Syntax	☒ V Function used in SQL Statement
☒ Hardcoded Application Id	☒ Unconditional Branch before other branches	☒ Associated Item of Validations
☒ Length of item name	☒ Referenced Button in When Button Pressed exists	☒ Report has default order
☒ Page Item has Help Text		

Summary



Summary

- Make use of tools and features to make your applications more secure
- Be aware of common security holes

sumnēva

Application Success

<http://sumneva.com>