

PDA (Private Defense Attorney) Metadata & Policy technical sheet

The following minimum required attributes were used for this implementation:

- gfipm:2.0:user:GivenName
- gfipm:2.0:user:SurName
- gfipm:2.0:user:FederationId
- gfipm:2.0:user:TelephoneNumber
- gfipm:2.0:user:EmailAddressText
- gfipm:2.0:user:EmployerName
- gfipm:2.0:user:IdentityProviderId

Additional attributes used for PDA specific policies

- gfipm:2.0:user:EmployeePositionName
- gfipm:2.0:user:SecurityClearanceLevelCode
- gfipm:2.0:user:SecurityClearanceExpirationDate
- gfipm:2.0:user:EmployeeId

For the PDA project, all the hard coded rules were removed and replaced with the flexible policies below. Under each policy is an explanation.

1 Policies

1.1 Private Attorney Expiration Policy

An attorney pays an annual renewal fee of \$100 dollars and is given a system username and password. This account expires annually. Each time the attorney logs into the system they receive a message on the home page indicating when their account will expire. If the expiration date is within 15 days of expiration, they will receive the same message but with a "Warning!" pre-pended to the message. If the attorney does not pay the renewal fee and their account expires, they will see a message stating that their account has expired and access to the system is locked down.

Future Expiration policy:

- UI/AccountExpirationAdvanceNoticePolicy
 - Warn_of_future_expiration_date(SecurityClearanceExpirationDate,-15) and EmployeePositionName = "Private Attorney"
- AppMessages/AttorneyFutureExpMsg
 - Report_as("AttorneyFutureExpMsg","Warning! Your subscription will expire on \$1") and report(SecurityClearanceExpirationDate)

The *UI/AccountExpirationAdvanceNoticePolicy* name is referenced in the java code and is used to evaluate a policy. The sub-bullet is the actual policy constraint. The *Warn_of_future_expiration_date* is a custom function created to evaluate when a future date is imminent. The *-15* in the *Warn_of_future_expiration_date(SecurityClearanceExpirationDate,-15)* function will become true if the *SecurityClearanceExpirationDate* is 15 days away. This allows the application to give a warning message when the date is near.

The *AppMessages/AttorneyFutureExpMsg* name is referenced in the java code and used to pull in two values: *Report_as("AttorneyFutureExpMsg", "Warning! Your subscription will expire on \$1")* and *report(SecurityClearanceExpirationDate)*. The *Report_as* built-in function is used to retrieve custom text and the *Report* built-in is used to retrieve any user attributes. After both *Report_as("AttorneyFutureExpMsg", "Warning! Your subscription will expire on \$1")* and *report(SecurityClearanceExpirationDate)* is retrieved by the java code, the \$1 place holder will be replaced with the *report(SecurityClearanceExpirationDate)*. The final message will be "Warning! Your subscription will expire on 04/22/2010." The \$1 wild card strategy was created by our team to allow a dynamic message. Oracle Entitlement Server does not have this capability built-in.

Non-expired policy:

- *UI/AccountExpirationAdvanceNoticePolicy*
 - *Warn_of_future_expiration_date(SecurityClearanceExpirationDate,-15)* and *EmployeePositionName = "Private Attorney"*
- *UI/AttorneyPolicy*
 - *EmployeePositionName = "Private Attorney"*
- *AppMessages/AttorneyNormalExpMsg*
 - *Report_as("AttorneyNormalExpMsg", "Your subscription will expire on \$1")* and *report(SecurityClearanceExpirationDate)*

If the *UI/AccountExpirationAdvanceNoticePolicy* is false then in the Java code checks to determine if the *UI/AttorneyPolicy* is true. If so, display the *AppMessages/AttorneyNormalExpMsg*. The final message will look like the following "Your subscription will expire on 04/22/2010." The only difference between the first policy message and the second policy message is the "Warning!" prefix is excluded in the later.

Expiration policy:

- *UI/AccountExpirationPolicy*
 - *Warn_of_future_expiration_date(SecurityClearanceExpirationDate,0)* and *EmployeePositionName = "Private Attorney"*
- *AppMessages/AttorneyExpMsg*
 - *Report_as("AttorneyExpMsg", "The account has been expired. Please renew your account")*

The *UI/AccountExpirationPolicy* uses the same function as the *UI/AccountExpirationAdvanceNoticePolicy* policy except instead of a -15 value passed, the value of 0 is passed. This will evaluate to true if the users expiration date is on the current date. If the *UI/AccountExpirationPolicy* evaluates to true, then use the *AppMessages/AttorneyExpMsg*.

1.2 Private Attorney Policy

Private Attorney Policy:

- *UI/AttorneyPolicy*
 - *EmployeePositionName = "Private Attorney"*
- *EmployeeId*
 - *Report(EmployeeId)*

If the *UI/AttorneyPolicy* is true, then pass the *EmployeeId* value to the query in the Java code. This will restrict attorney queries to cases they are associated to. If the *UI/AttorneyPolicy* evaluates to false, then the *EmployeeId* is not passed to the query which gives the user unrestricted query access.

1.3 Administrator Policy

The administrator of the PDA application has the ability to setup new users in (OID) Oracle Internet Directory and to run a user Audit Report.

Administration Policy:



- UI/AdministrationPolicy
 - SecurityClearanceCodeLevel = "PDA Administrator" and EmployerName = "Superior Court"

If the *UI/AdministrationPolicy* evaluates to true, then display two new admin buttons to the user and allow them to setup new users and run audit reports.