Zentrale Berechtigungssteuerung

Vorstellung

* Thomas Schild
* Studium am b.i.b. International College Paderborn
* NOWEDA e.G seit 2009
* Pharmaziegroßhandel mit 16 Niederlassungen in Deutschland

Problemstellung

* Eine seit 15 Jahre gewachsene Datenbank
* Kein einheitliches Konzept für APEX Anwendungen
* Erschwerte Wartung und Pflege
* Fehlende Transparenz

Angestrebtes Ziel

* Einheitliche Steuerung auf Basis des Active Directory
* Zentrale Zugriffssteuerung auf Anwendungs-, Seiten- und Itemebene
* Automatisierte Aktualisierung der Berechtigungen
* Sich ergänzende Berechtigungen
* Keine Änderung in der Anwendung bei neuen Berechtigungen
* Transparenz!

Beispielanwendung

* Beispielanwendung
  * Rechtesteuerung
  * Testanwendung

Umsetzung

Anwender- und Rollendaten

* Loginnamen der Anwender
* Berechtigungsrollen
* Zuordnung zwischen Berechtigungsrollen und Anwender

Umsetzung

Technische Daten

* Zuordnung von Rollen und den APEX-Elementen
  * Anwendung
  * Anwendungsseite
  * Region

Umsetzung

Apex Prozesse

Berechtigungskontrolle für:

* Anwendung
* Anwendungsseiten
* Tabreiter
* Regionen

Umsetzung

Itemtyp	Auflistung	Conditionparameter
Textfeld, Selectlist, …	apex_application.g_item_name	apex_application.g_item_display_when_type
Tabreiter	apex_application.g_tab_name	apex_application.g_tab_plsql_condition_type
Page Button	apex_application.g_button_id	apex_application.g_button_condition_type
…	…	…

Weitere Parameter im Packagespec APEX_0xxxxx.wwv_flow

Ende

Process [Download Source]

```
DECLARE
   cRollen VARCHAR2(2000);
BEGIN
   apex_authentication.login(
       p_username => :P101_USERNAME,
       p_password => :P101_PASSWORD );

   -- Ist der Benutzer einer der benötigten Rollen zugeordnet?
   FOR rec IN (SELECT a.rolle_id rolle
               FROM berechtigung b INNER JOIN rolle_application a ON
                      b.rolle_id = a.rolle_id
               WHERE UPPER(b.benutzer) = UPPER(:P101_USERNAME) AND
                      a.application_id = :APP_ID)
   LOOP
      cRollen := cRollen  || ';' || rec.rolle;
   END LOOP;
   :P0_ROLLEN := cRollen || ';';

   -- Wenn nicht, dann soll er direkt wieder abgemeldet werden
   IF cRollen IS NULL THEN

     owa_util.mime_header('text/html', FALSE);
     owa_cookie.send(name => 'LOGIN_USERNAME_COOKIE',
                     value => NULL,
                     expires => SYSDATE - 365);

     apex_custom_auth.logout(p_this_app => v('APP_ID'),
                             p_next_app_page_sess => v('APP_ID') || ':101');
     -- tell APEX engine to quit
     wwv_flow.g_unrecoverable_error := true;
   END IF;
END;
```

Authorization Scheme

* Scheme Type: PL/SQL Function Returning Boolean

* PL/SQL Function Body:

```
DECLARE
   nCount NUMBER;
BEGIN
   -- Ist der Benutzer für die Seite berechtigt?
   SELECT COUNT(1) INTO nCount
   FROM rolle_page
   WHERE application_id = :APP_ID AND
         page_id = :APP_PAGE_ID AND
         INSTR(:P0_ROLLEN, ';'|| rolle_id||';') > 0;

   IF nCount = 0 AND :APP_PAGE_ID NOT IN (0, 101) THEN
      RETURN false;
   ELSE
      Return true;
   END IF;
END;
```

* Process Point: On Load: Before Header (page template header)

* Name: Tabberechtigung

* Type: PL/SQL Anonymous Block

Source

* Process Text

```
BEGIN
     -- alle Applicationseiten durchlaufen, für die der User keine Berechtigung hat
     FOR rec1 IN (SELECT a.application_id, a.page_id
                  FROM Apex_Application_Pages a
                  WHERE a.application_id = :APP_ID AND
                        a.page_id NOT IN (0) AND
                        NOT EXISTS( SELECT 1
                                    FROM rolle_page p
                                    WHERE p.application_id = a.application_id AND
                                          p.page_id = a.page_id AND
                                          INSTR(:P0_ROLLEN, ';'|| p.rolle_id||';') > 0)
                  )
     LOOP
        -- Entsprechenden Tabreiter sperren
        FOR rec2 IN (SELECT tab_name FROM apex_application_tabs
                     WHERE application_id = rec1.application_id AND
                     tab_page = rec1.page_id)
        LOOP
            FOR i in 1 .. apex_application.g_tab_name.COUNT LOOP
                IF apex_application.g_tab_name(i) = rec2.tab_name THEN
                    apex_application.g_tab_plsql_condition_type(i) := 'NEVER';
                END IF;
            END LOOP;
        END LOOP;
     END LOOP;
END;
```