

IX Seminarium PLOUG
Warszawa
Maj 2004

Podnoszenie poziomu bezpieczeństwa instalacji i baz danych Oracle

Krzysztof Mikołajczyk

Bull Polska

e-mail: krzysztof.mikolajczyk@bull.com.pl

Abstrakt

Standardowe instalacje produktów Oracle zawierają wiele podatności wiążących się z „fabrycznymi” ustawieniami. Poziom bezpieczeństwa instalacji daje się jednak stosunkowo łatwo podnieść. Prezentacja ma na celu przedstawienie najważniejszych kroków, jakie należy wykonać w celu zwiększenia i zweryfikowania poziomu bezpieczeństwa instalacji bazy Oracle. Zostaną omówione najważniejsze cechy i funkcje związane z bezpieczeństwem Oracle. Na co zwracać uwagę przy instalacji i utrzymaniu bazy danych? Co zrobić aby zabezpieczyć się przed ewentualnymi zagrożeniami?

Omówienie bazuje na instalacji Oracle 9iR2, ale większość poruszanych zagadnień będzie również dotyczyć Oracle 8i (a także Oracle10g).

Rysunki pochodzą z programu instalatora Oracle lub ze stron internetowych oracle.com.

Wstęp

Ochrona informacji jest jednym z istotnych czynników obsługi informatycznej. Zabezpieczenie przed nieautoryzowanym dostępem do danych jest kluczowym zadaniem działu IT. W większych firmach i organizacjach jest wręcz wydzielony dział, który zajmuje się wyłącznie zapewnieniem bezpieczeństwa i ochroną zasobów z danymi. Oracle w kolejnych wersjach systemu zarządzania relacyjną bazą danych oferuje najnowocześniejsze cechy i metody ochrony. Ale warto wiedzieć, jak z tej funkcjonalności skorzystać i na co zwrócić uwagę, aby zwiększyć jeszcze bardziej bezpieczeństwo danych.

Generalnie materiał opisany w tej prezentacji dotyczy bazy danych Oracle 9iR2 (9.2), ale większość punktów dotyczy również wersji poprzednich, jak i następnych (10g).

Dostęp do serwera

Fizyczny dostęp do serwera zawsze gwarantuje możliwość uzyskania zwiększonego dostępu do danych. Po starcie systemu z CD można dostać się do systemu w znacznie łatwiejszy sposób. Dlatego też dostęp do każdego serwera powinien być ograniczony.

Nie należy również zapominać o ochronie nośników – kopii zabezpieczających, plików eksportu, raportów wygenerowanych z bazy. W tym wypadku dane nie są chronione – wszystko zależy od organizacji. Tasiemka z wygenerowanym plikiem eksportu systemowego, leżąca w ogólnie dostępnym miejscu, może dać dostęp do danych bez potrzeby włamywania się do systemu.

System operacyjny

Bezpieczeństwo systemu operacyjnego przekłada się pośrednio na bezpieczeństwo pracujących na nim aplikacji. Jeśli system operacyjny nie jest prawidłowo chroniony, to nieupoważniony dostęp do danych aplikacji jest wydatnie ułatwiony. Jednak nie należy przesadzać z utrudnianiem dostępu, gdyż efekt może być wręcz przeciwny - nagromadzenie różnych zabezpieczeń może spowodować zmniejszenie odporności użytkowników i ułatwiony dostęp do danych.

Warto jest podzielić role administracyjne, zarówno na poziomie administratorów systemu operacyjnego, jak i baz danych (również dla innych aplikacji). Generalnie konta systemu operacyjnego dla administratorów różnych poziomów powinny być różne (nawet jeśli daną rolę pełni ta sama osoba). Różne funkcje, różny zakres odpowiedzialności, różne konta w systemie - taką zasadę należałoby przyjąć. Jednak z drugiej strony należy pamiętać, że każde dodatkowe konto potencjalnie stanowi zwiększoną możliwość uzyskania dostępu do systemu.

Jeśli w systemie jest więcej niż jedna (produkcyjna) baza danych - dla każdej z nich powinno się rozważyć utworzenie dedykowanego konta dla administratora - jeśli bazy i administratorzy są inni, wtedy powinno być to obowiązkowe.

W systemie operacyjnym należy wyłączyć niepotrzebne i nieużywane usługi (zarówno dla protokołu TCP jak i UDP). Pozostawienie takiej usługi (lub możliwości jej uruchomienia) jest dodatkowym zagrożeniem. Dodatkowym zabezpieczeniem może być zmiana portów dla niektórych usług.

Od czasu do czasu pojawiają się uaktualnienia systemu. Mogą one dotyczyć znalezionych błędów, jak również bezpieczeństwa. Na ogół w opisie znajduje się informacja, co dana poprawka zmienia. Bezwzględnie należy śledzić informacje o dostępnych uaktualnieniach (szczególnie dotyczących bezpieczeństwa) i ewentualnie instalować je. Należy pamiętać, że niektóre nowe wersje

mogą mieć zmienione działanie (np. w celu zwiększenia bezpieczeństwa systemu) – ważne jest zapoznanie się z opisem PRZED zainstalowaniem poprawki. I nie należy zapominać o wykonaniu kopii systemu na wypadek, gdyby instalacja się nie powiodła.

Koniecznym jest śledzić, co się dzieje w systemie. W szczególności dotyczy to plików z rozszerzonymi uprawnieniami. Każdy nowy plik z rozszerzonymi uprawnieniami powinien być starannie sprawdzony, czy rzeczywiście musi mieć takie uprawnienia (a także skąd się wziął w systemie!). Szczególną uwagę należy zwrócić na pliki, których właścicielem jest **root**. Do wygenerowania listy plików z rozszerzonymi uprawnieniami może posłużyć poniższe polecenie:

```
# find / -perm -4000 -exec ls -l {} \;
```

Należy również pilnować praw dostępu. Dotyczy to nie tylko praw do plików Oracle, ale również plików bazy danych i plików tworzonych w trakcie funkcjonowania bazy danych (pliki te tworzone są zgodnie ze zdefiniowaną maską **umask** – domyślnie **022** – czyli wszyscy mogą je czytać). Chodzi tu w pierwszej kolejności o pliki typu log (np. `background_dump_dest`), jak również wszelkiego rodzaju skrypty, które mogą zawierać wpisane wprost hasła. Należy również zabezpieczyć poprzez odpowiednie prawa dostępu raporty z bazy oraz pliki eksportu. Przy ustalaniu odpowiednich praw dostępu należy również zwrócić uwagę na katalogi i podkatalogi, a nie tylko na pliki końcowe. Poniższe polecenie pokazuje aktualne ustawienia dla plików logowania:

```
SQL> show parameter dump_dest
```

NAME	TYPE	VALUE
background_dump_dest	string	/oracle/9/admin/ora9/bdump
core_dump_dest	string	/oracle/9/admin/ora9/cdump
user_dump_dest	string	/oracle/9/admin/ora9/udump

Odpowiednie prawa dostępu chronią również przed nieuprawnionym dostępem z poziomu programu nasłuchu LISTENER, aczkolwiek wymaga to odpowiedniego uruchomienia tego procesu.

„Tradycyjnie” uruchomiony proces LISTENER:

```
oracle9@linux:~> ps -ef|grep LIST
oracle9  3082      1  0 23:02 pts/2    00:00:00 /oracle/9/product/bin/tnslsnr
LISTENER -inherit
```

Bezpiecznie uruchomiony proces LISTENER:

```
oracle9@linux:~> ps -ef|grep LIST
daemon   3356      1  0 23:08 pts/3    00:00:00 /oracle/9/product/bin/tnslsnr
LISTENER -inherit
```

Sieć

Problem sieci jest jednym z poważniejszych problemów związanych z bezpieczeństwem. Wiadomo, że dane (również hasła dostępu do różnych serwerów) wędrują po sieci i mogą być przechwycone - wystarczy fizyczny dostęp do sieci, aby móc przejąć ramki. Aby się przed tym zabez-

pieczyć, należy przede wszystkim postarać się o odseparowanie ruchu obcego poprzez utworzenie wydzielonej podsieci (czy to poprzez użycie ściany ogniowej, czy też przez sieci prywatne VPN). W tym momencie wiadomo, że potencjalny wyciek danych oznacza dostęp do takiej sieci, co wydatnie ogranicza możliwości przechwycenia danych. Przy użyciu firewalla należy pamiętać, aby nie pozostawiać otwartych portów do usług sieciowych Oracla.

Odseparowanie serwera http i serwera bazy danych jest wielce wskazane. Dzięki temu osoby korzystające z protokołu http nie mają bezpośredniego dostępu do serwera bazodanowego. Daje to dodatkową korzyść w postaci lepszego rozłożenia obciążenia.

Dodatkowo, lub też gdy nie jest możliwe utworzenie sieci wydzielonej, można dane zaszyfrować. Na ogół powoduje to zmniejszoną przepustowość (wszystkie dane muszą być najpierw szyfrowane, a następnie deszyfrowane), jednak zastosowanie akceleratorów sprzętowych znacznie zmniejsza ten problem. Szyfrowanie na poziomie Oracla oferuje opcja Oracle Advanced Security (dostępna w wersji Enterprise Edition).

Zmniejszenie problemów związanych z przesyłaniem haseł można osiągnąć również przez zastosowanie serwera logowania i metody Single Sign-On.

Środowisko bazy danych

W optymalnej sytuacji są dostępne trzy serwery - dla prowadzenia prac rozwojowych, do celów testowych i serwer produkcyjny. Zakładając, że dla użytkownika końcowego prace rozwojowe i testy nie są prowadzone w sposób ciągły, można zredukować te dwa serwery do jednego.

Odrębne serwery

W tym przypadku jeden z serwerów jest przeznaczony do działania produkcyjnego, drugi do prac testowych i pewnych prac związanych z modyfikacją aplikacji. Generalnie oba serwery powinny być takiej samej klasy. Moc obliczeniowa i inne zasoby nie muszą być takie same, aczkolwiek testy wydajnościowe najlepiej byłoby przeprowadzać w identycznych warunkach (takie same dane, takie same rozłożenie plików, taka sama moc obliczeniowa) - wtedy wnioski są bezpośrednio przekładane na bazę produkcyjną. Dodatkowo serwer testowy może stanowić dodatkowe zabezpieczenie w przypadku awarii serwera głównego. Jednak zazwyczaj jest to rozwiązanie zbyt kosztowne i serwer testowy jest mniejszy.

Odrębne instalacje (różne ORACLE_HOME)

W tym wypadku nie ma odrębnego serwera testowego, zarówno baza produkcyjna, jak i testowa pracują na tym samym fizycznym serwerze - podstawowym problemem są testy związane z modyfikacją elementów systemu operacyjnego, których nie da się bezboleśnie wykonać, jak również problemy związane z wydajnością - baza testowa i baza produkcyjna "przeszkadzają" sobie nawzajem. To rozwiązanie pozwala przetestować zachowanie bazy danych po wgraniu poprawki.

Odrębne bazy – grupy administratorów (różne ORACLE_SID)

Utworzone są dwie odrębne bazy danych, jednak obie korzystają z tej samej instalacji. Jest to kolejne ograniczenie - każda modyfikacja instalacji testowej oznacza również modyfikację instalacji produkcyjnej, co w pewnych przypadkach może być ryzykowne. Dodatkowo grupa administratorów obu baz danych jest taka sama (**dba**).

Odrębne schematy (różne USER1) - uwaga na PUBLIC

Zarówno baza produkcyjna, jak i testowa istnieją w tej samej fizycznej bazie, tylko w obszarze innego użytkownika. Jest to rozwiązanie bardzo tanie, ale jednocześnie bardzo ryzykowne (np. użycie przywilejów typu PUBLIC w bazie testowej ma od razu przełożenie na bazę produkcyjną. Rozwiązanie bardzo ryzykowne.

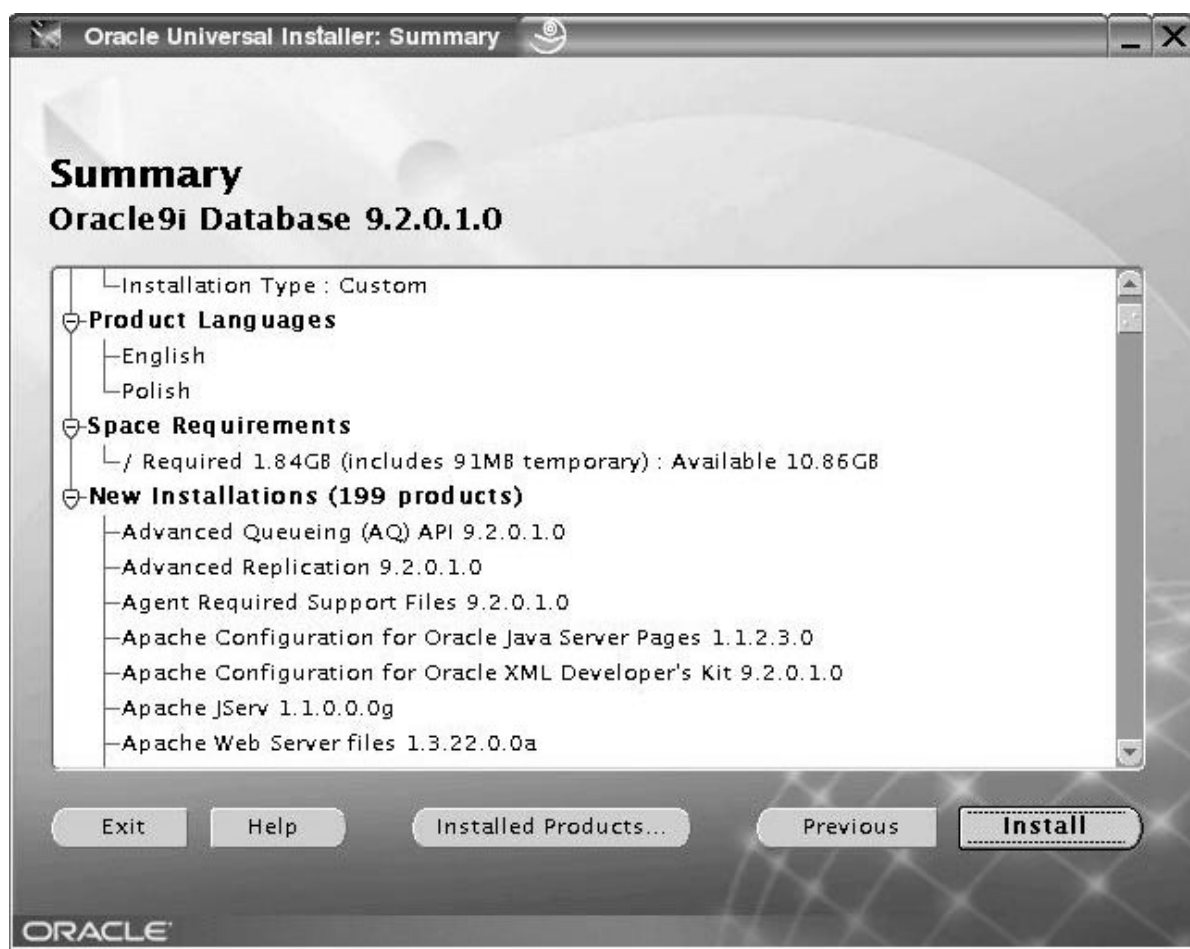
Przygotowanie instalacji

Przed rozpoczęciem instalacji należy określić kilka parametrów, od których w istotny sposób będzie zależeć bezpieczeństwo systemu:

- właściciel instalacji (oracle),
- grupy administracyjne - o podwyższonych uprawnieniach (OSDBA, OSOPER),
- inne grupy użytkowników (oinstall, orainventory),
- katalog domowy, ORACLE_BASE, ORACLE_HOME, oraInventory, oui, jre i inne katalogi (i określenie praw dostępu do nich)
- OEM - gdzie zostanie umieszczone repozytorium
- jakie produkty są niezbędne do instalacji
- wybór protokołów sieciowych

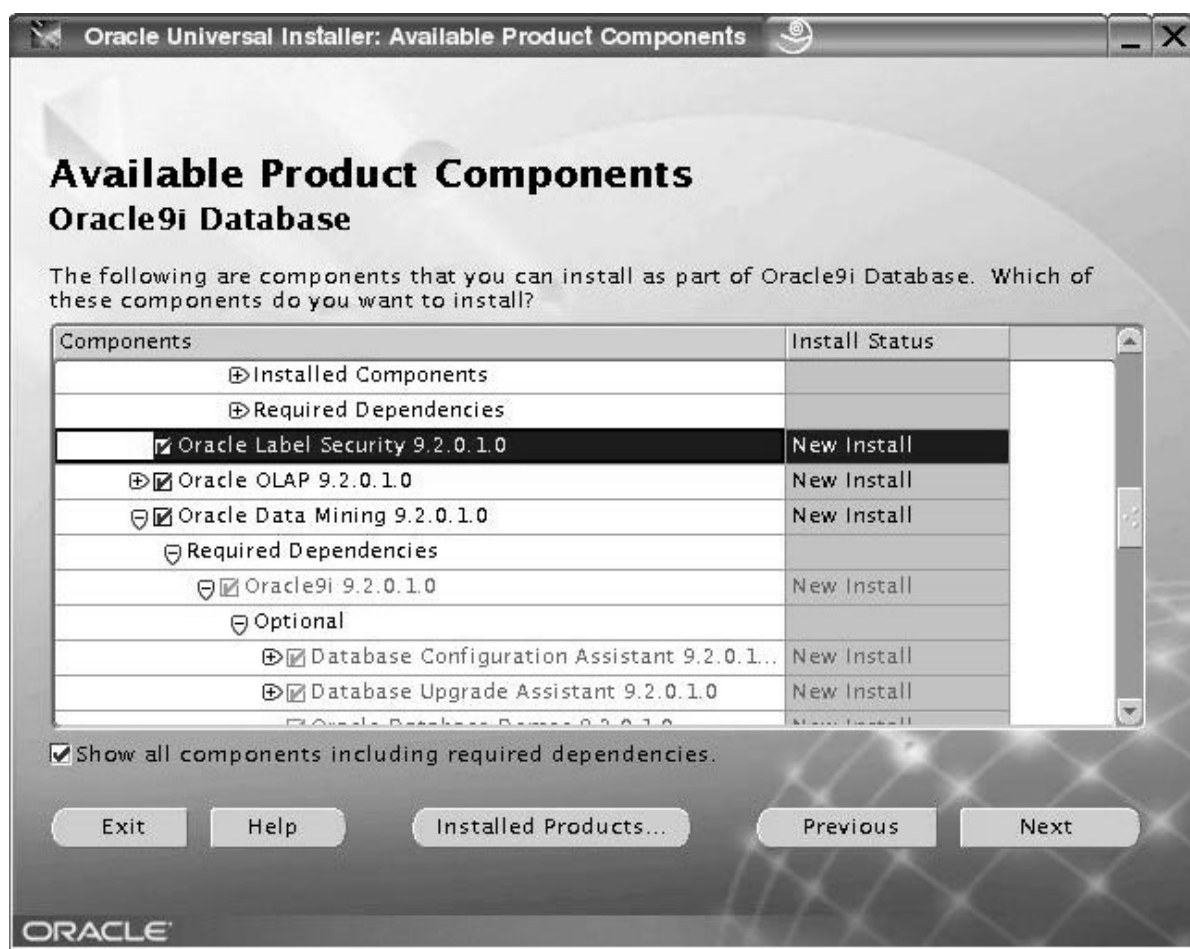
Instalacja

Istnieją dwa podejścia: instalacja wszystkiego (np. domyślna) , a następnie deinstalacja zbędnych opcji i produktów i druga opcja, która zaleca wybór produktów do instalacji (uwaga na niektóre opcje, które występują kilkakrotnie). Nawet prosta instalacja może zawierać kilkaset produktów.



Rys. 1

Zawsze przy instalacji należy zwrócić uwagę na zależności między produktami i na konieczność relinkowania produktów, gdyż może to wydatnie zwiększyć czas potrzebny do instalacji.



Rys. 2

Należy zwrócić również uwagę na wybór protokołów sieciowych - wybór należy ograniczyć tylko do tych, które rzeczywiście będą używane.

Niezainstalowane opcje mogą być w razie potrzeby doinstalowane.

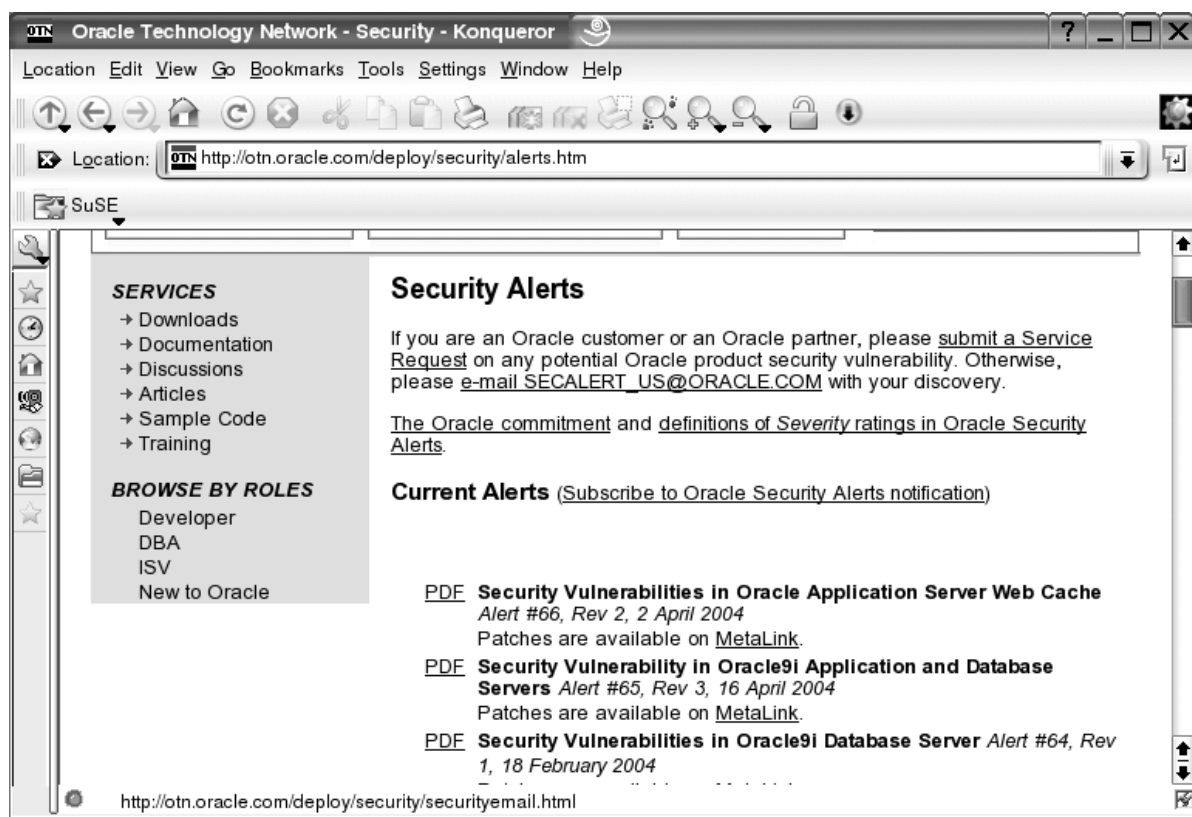
Uaktualnienia systemu i bazy danych

Nie należy również zapominać o uaktualnieniach systemu. Często dotyczą one znalezionych błędów, czasami również bezpieczeństwa. Na ogół w opisie znajduje się informacja, co dana poprawka zmienia lub poprawia. Bezwzględnie należy śledzić informacje o dostępnych uaktualnieniach (alerty dotyczące naruszenia zasad bezpieczeństwa można znaleźć na stronie: <http://otn.oracle.com/deploy/security/alerts.htm>) i ewentualnie instalować je (poprawki dostępne są ze strony <http://metalink.oracle.com>). Należy pamiętać, że wersja po uaktualnieniu może inaczej działać - PRZED zainstalowaniem poprawki należy się dokładnie zapoznać z jej opisem (i oczywiście wykonać kopię bezpieczeństwa zarówno systemu, jak i bazy danych). Dobrym zwyczajem jest wykonanie instalacji poprawki najpierw w środowisku testowym, można wtedy zapoznać się z potencjalnymi problemami w trakcie instalacji w środowisku produkcyjnym.

Przy aktualizacji zawsze trzeba pamiętać, że nie wystarczy aktualizacja oprogramowania. Często trzeba wykonać dodatkowo aktualizację bazy danych (lub kilku, gdy jest ich więcej).

Poprawki dotyczące bezpieczeństwa powinny być instalowane. Opisy zagrożeń można znaleźć na stronie <http://otn.oracle.com/deploy/security/alerts.htm>. Można tam wpisać się na listę i infor-

mację o pojawiających się alertach otrzymywać na bieżąco (ewentualne poprawki można pobrać ze strony <http://metalink.oracle.com>).



Rys. 3

Użytkownicy

W systemie operacyjnym są zdefiniowane dwie grupy (OSDBA i OSOPER), które gwarantują użytkownikom należącym do nich zwiększone uprawnienia. Nazwy tych grup są określane na etapie instalacji



Rys.4

Lista użytkowników domyślnych utworzonych na etapie tworzenia bazy danych jest dość długa. W większości przypadków konta zostaną zablokowane (status EXPIRED & LOCKED), jednak zawsze warto sprawdzić, czy stan ten wystąpił. Zwykle cztery konta, jeśli istnieją, (SYS, SYSTEM, SCOTT i DBSNMP) pozostają w trybie OPEN. W bazie produkcyjnej użytkownik SCOTT powinien być zablokowany.

Poniżej przedstawiona jest lista kont i domyślnym statusem po instalacji:

ADAMS	EXPIRED & LOCKED
AURORA\$JIS\$UTILITY\$	OPEN
AURORA\$ORB\$UNAUTHENTICATED	OPEN
BLAKE	EXPIRED & LOCKED
CLARK	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED

DBSNMP	OPEN
HR	EXPIRED & LOCKED
JONES	EXPIRED & LOCKED
LBACSYS	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
OE	EXPIRED & LOCKED
OLAPDBA	EXPIRED & LOCKED
OLAPSVR	EXPIRED & LOCKED
OLAPSYS	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OSE\$HTTP\$ADMIN	OPEN
OUTLN	OPEN
PM	EXPIRED & LOCKED
QS	EXPIRED & LOCKED
QS_ADM	EXPIRED & LOCKED
QS_CB	EXPIRED & LOCKED
QS_CBADM	EXPIRED & LOCKED
QS_CS	EXPIRED & LOCKED
QS_ES	EXPIRED & LOCKED
QS_OS	EXPIRED & LOCKED
QS_WS	EXPIRED & LOCKED
SCOTT	OPEN
SH	EXPIRED & LOCKED
SYS	OPEN
SYSTEM	OPEN

Po zainstalowaniu aplikacji należy ponownie sprawdzić, jacy nowi użytkownicy pojawili się w bazie danych i jaki jest ich status. Po instalacji jakichkolwiek uaktualnień (instalacji Oracle, bądź aplikacji) należy zweryfikować status użytkowników bazy.

Jeśli pojawi się w bazie użytkownik nieznanego pochodzenia, można go testowo wyłączyć (jednak sprawdzenie, czy jest używany przez aplikację, może wyjść dopiero po jakimś czasie).

Inny problem stanowią użytkownicy, którzy są właścicielami obiektów aplikacji. Zasadniczo powinni oni też być wyłączeni (kwestia praw dostępu do obiektów), a dostęp do danych powinien być realizowany przez inne konta, jednak jest to uzależnione od aplikacji.

Zarządzanie hasłami

Większość użytkowników jest instalowana (zakładana) z domyślnymi hasłami. Jest to duże ułatwienie dla administratorów (biorąc pod uwagę ilość definiowanych kont), jednak jest to również duże ułatwienie dla uzyskania nieautoryzowanego dostępu. Dlatego też trzeba pamiętać o zmianie haseł z domyślnych na inne (nawet dla kont zablokowanych – zawsze może zaistnieć konieczność odblokowania ich). Zarządzanie tymi hasłami jest o tyle proste, że uprzywilejowany użytkownik zawsze może te hasła zmienić.

Użytkownicy administracyjni (SYS i SYSTEM) powinni mieć zmienione hasło tak szybko, jak to jest możliwe (kreator bazy **dbca** wymusza zmianę tych haseł).

Przy zakładaniu użytkowników takie same hasła po zakodowaniu wyglądają inaczej:

```
SQL> create user u1 identified by u1 profile PASS_LOG
User created.
```

```
SQL> create user u2 identified by u1 profile PASS_LOG
User created.
SQL> select username, password, account_status, profile from dba_users where
username like 'U_' ;
USERNAME PASSWORD          ACCOUNT_STATUS  PROFILE
-----
U1          3E81B724A296E296 OPEN           PASS_LOG
U2          F1B3E20A8E59BF2F OPEN           PASS_LOG
```

Należy również pamiętać o włączeniu zabezpieczeń związanych z utrzymaniem haseł (analiza prostych haseł, długość haseł, częstość zmian haseł), jednak nie należy zbyt utrudniać życia użytkownikom – zbyt rygorystyczne reguły mogą doprowadzić do przechowywania haseł w sposób jawny (uwaga ta nie dotyczy kont administracyjnych – tu reguły powinny być bardziej krytyczne).

Do obsługi wymuszania haseł świetnie nadają się profile (nadają się one również do innych celów). Po utworzeniu odpowiedniego profilu można go przypisać do użytkownika (bądź grupy użytkowników).

```
SQL> create profile pass_log limit FAILED_LOGIN_ATTEMPTS 3  PASSWORD_REUSE_MAX
1 PASSWORD_REUSE_TIME 1;
Profile created.
```

```
SQL> select * from dba_profiles where profile='PASS_LOG';
PROFILE      RESOURCE_NAME                                RESOURCE LIMIT
-----
PASS_LOG     COMPOSITE_LIMIT                             KERNEL  DEFAULT
PASS_LOG     SESSIONS_PER_USER                           KERNEL  DEFAULT
PASS_LOG     CPU_PER_SESSION                             KERNEL  DEFAULT
PASS_LOG     CPU_PER_CALL                                KERNEL  DEFAULT
PASS_LOG     LOGICAL_READS_PER_SESSION                   KERNEL  DEFAULT
PASS_LOG     LOGICAL_READS_PER_CALL                      KERNEL  DEFAULT
PASS_LOG     IDLE_TIME                                    KERNEL  DEFAULT
PASS_LOG     CONNECT_TIME                                KERNEL  DEFAULT
PASS_LOG     PRIVATE_SGA                                 KERNEL  DEFAULT
PASS_LOG     FAILED_LOGIN_ATTEMPTS                       PASSWORD 3
PASS_LOG     PASSWORD_LIFE_TIME                           PASSWORD DEFAULT
PASS_LOG     PASSWORD_REUSE_TIME                         PASSWORD 1
PASS_LOG     PASSWORD_REUSE_MAX                         PASSWORD 1
PASS_LOG     PASSWORD_VERIFY_FUNCTION                     PASSWORD DEFAULT
PASS_LOG     PASSWORD_LOCK_TIME                           PASSWORD DEFAULT
PASS_LOG     PASSWORD_GRACE_TIME                         PASSWORD DEFAULT
```

Próba ponownego wprowadzenia tego samego hasła kończy się niepowodzeniem (Uwaga: znaki wprowadzonego hasła *u1*, wydrukowane tutaj pismem pochyłym, nie wyświetlają się):

```
SQL> password user u1
Changing password for user
New password:u1
Retype new password:u1
ERROR:
ORA-01017: invalid username/password; logon denied

Password unchanged
```

Na uwagę zasługują również wszelkie dodatkowe metody zwiększające bezpieczeństwo, jak włączenie Oracle Advanced Security lub włączenie mechanizmów Single Sign-On.

Przywileje użytkowników bazy danych

Przywileje użytkowników bazy danych powinny być jak najmniejsze, zarówno jeśli chodzi o przywileje systemowe (np. CREATE TABLE), jak i obiektowe (np. DELETE). Każdy użytkownik powinien otrzymać tylko taki zestaw przywilejów, jaki jest mu niezbędny do wykonania konkretnej pracy. Do łatwiejszego zarządzania przywilejami przydatne są role, które pozwalają zdefiniować zestawy przywilejów. Jednak jeśli jakiś użytkownik nie potrzebuje wszystkich przywilejów danej roli, to lepiej jest zdefiniować inną rolę dla niego lub nadać uprawnienia indywidualnie – nie można nadać przywilejów poprzez rolę, a następnie odwołać te, które są niepotrzebne. Należy również zwrócić uwagę na przywileje nadane wielokrotnie (np. poprzez różne role lub przez różnych użytkowników). Odebranie przywileju może być nieskuteczne, jeśli był on nadany wielokrotnie.

Bardzo uważnie należy nadawać przywileje z opcją do przekazywania ich innym użytkownikom (WITH ADMIN OPTION, WITH GRANT OPTION). W takim przypadku śledzenie rzeczywistych uprawnień poszczególnych użytkowników może być utrudnione. W takim wypadku dobrze jest mieć zdefiniowany zestaw przywilejów dla konkretnego użytkownika i okresowo weryfikować, czy nie pojawiają się jakieś odstępstwa.

Sprawdzenie listy użytkowników z przywilejem WITH ADMIN OPTION:

```
SQL> select grantee, privilege
  2  from dba_sys_privs
  3  where admin_option='YES'
  4  union
  5  select grantee, granted_role
  6  from dba_role_privs
  7  where admin_option='YES';
```

GRANTEE	PRIVILEGE
AQ_ADMINISTRATOR_ROLE	CREATE EVALUATION CONTEXT
AQ_ADMINISTRATOR_ROLE	CREATE RULE
AQ_ADMINISTRATOR_ROLE	CREATE RULE SET

... /wydruk obcięty/

Sprawdzenie listy użytkowników z przywilejem WITH GRANT OPTION:

```
SQL> select grantee, privilege, table_name
  2  from dba_tab_privs
  3  where grantable='YES'
  4  union
  5  select grantee, privilege, table_name
  6  from dba_col_privs
  7  where grantable='YES';
```

GRANTEE	PRIVILEGE	TABLE_NAME
CTXSYS	SELECT	ARGUMENT\$
CTXSYS	SELECT	DBA_CONS_COLUMNS
CTXSYS	SELECT	DBA_DB_LINKS
CTXSYS	SELECT	DBA_OBJECTS
CTXSYS	SELECT	DBA_ROLES
CTXSYS	SELECT	DBA_ROLE_PRIVS

... /wydruk obcięty/

Na szczególną uwagę zasługują przywileje typu ANY (np. DROP ANY TABLE). Te przywileje powinny być nadawane ze szczególną ostrożnością, gdyż ANY oznacza wszystkie obiekty danego typu, również obiekty słownikowe. Rzadko który użytkownik potrzebuje aż tak szerokie uprawnienia. W szczególności przywilej DROP ANY TABLE może spowodować dużo szkód (dotyczy on również tabel słownika danych). W celu ochrony słownika danych można dodać parametr inicjalizacyjny *init.ora* 07_DICTIONARY_ACCESSIBILITY=FALSE, który powoduje, że tylko użytkownik z przywilejem **dba** (po podłączeniu CONNECT / AS SYSDBA) może korzystać z przywilejów typu ANY na tabelach słownika.

Sprawdzenie użytkowników posiadających przywilej typu ANY:

```
SQL> select grantee, privilege
2   from dba_sys_privs
3   where privilege like '%ANY%';
```

GRANTEE	PRIVILEGE
DBA	AUDIT ANY
DBA	ANALYZE ANY
DBA	DROP ANY ROLE
DBA	DROP ANY RULE

... /wydruk obcięty/

Drugą grupą niebezpiecznych przywilejów są te nadane dla użytkownika PUBLIC. Przywilej typu PUBLIC dotyczy wszystkich użytkowników w bazie danych. Nadawanie uprawnień tego typu powinno być ograniczone do minimum. Dotyczy to również obiektów systemowych, które mogą być domyślnie dostępne dla użytkownika typu PUBLIC. Szczegółowo należy przejrzeć przywilej EXECUTE nadany użytkownikowi PUBLIC na pakiety i procedury PL/SQL, gdyż niektóre z nich mogą być niebezpieczne (szczególnie niebezpieczny jest pakiet UTL_FILE, umożliwiający dostęp do plików systemu operacyjnego).

Następna grupa to przywileje właścicieli poszczególnych obiektów bazy danych – są one pełne (jeśli chodzi o uprawnienia obiektowe). Dlatego należy unikać podłączania się do bazy jako właściciel obiektów – lepiej jest realizować dostęp do bazy przez inne konta, do których można zdefiniować odpowiedni zestaw przywilejów, a konto właściciela zachować do szczególnych przypadków (np. modyfikacja obiektów).

Proces nasłuchu LISTENER

Ważną rzeczą jest zarządzanie procesem Oracle Listener. Koniecznie należy wprowadzić hasło na zdalne zarządzanie konfiguracją. Hasło powinno być wprowadzone poleceniem **change_password**, wtedy jest przechowywane w postaci zakodowanej. Można również wpisać je ręcz-

nie, ale wtedy jest podane jawnym tekstem (i przy nieodpowiednich prawach dostępu może być odczytane).

Przy nieprawidłowym haśle proces nie wykonuje żądanych poleceń:

```
LSNRCTL> reload
Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
TNS-01169: The listener has not recognized the password
```

Dodatkowo należy wprowadzić parametr (w pliku *listener.ora*) `ADMIN_RESTRICTIONS_nazwa_procesu=ON`, który zabezpieczy przed nieautoryzowaną administracją (komenda SET jest zablokowana, wszystkie zmiany dokonywane są w pliku *listener.ora*).

```
LSNRCTL> set log_directory /tmp
Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
listener parameter "log_directory" set to /tmp
The command completed successfully
```

Po włączeniu wyżej podanej opcji:

```
LSNRCTL> set log_directory /tmp
Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
TNS-12508: TNS:listener could not resolve the COMMAND given
```

Aby móc śledzić, czy dzieje się coś niedobrego z procesem nasłuchu, należy włączyć logowanie (i dodatkowo przeglądać logi !) W przypadku stwierdzenia nieprawidłowości lub wystąpienia jakichkolwiek wątpliwości należy sytuację wyjaśnić.

Następujące parametry pliku *sqlnet.ora* (lub *protocol.ora*) pozwalają na filtrowanie dostępu do baz danych:

```
TCP.VALIDNODE_CHECKING=YES
TCP.EXCLUDED_NODES={adresy_IP}
TCP.INVITED_NODES={adresy_IP}
```

Pierwszy z parametrów włącza sprawdzanie, drugi podaje listę adresów IP, z których nie można się podłączać do Oracle Listener (dostęp jest zakazany), natomiast trzeci podaje listę adresów IP, które są uprawnione do dostępu.

Warto również rozważyć uruchomienie procesu nasłuchu na innym porcie TCP (należy wybrać port, który nie jest używany). Czasami warto jest uruchomić kilka procesów nasłuchu (np. W celu odseparowania PLSExtProc).

W katalogu `$ORACLE_HOME/network/admin/samples` można znaleźć przykłady powyższych plików.

Źródła informacji

Podstawowym źródłem informacji na temat bezpieczeństwa bazy danych Oracle::

- dokumentacja (głównie *Administration Guide*)

- metalink (<http://metalink.oracle.com>) - strona wymagająca wykupienia asysty technicznej
- Oracle Technology Network (<http://otn.oracle.com> lub <http://technet.oracle.com>) - strona wymaga rejestracji, ale nie jest płatna
- Strona z alertami dotyczącymi bezpieczeństwa - można się zapisać na listę (<http://otn.oracle.com/deploy/security/alerts.htm>)
- CERT (<http://www.cert.org>)
- subskrypcja list dotyczących tematu bezpieczeństwa
- wyszukiwarki internetowe
- inne publikacje (np. materiały z konferencji, itp.)

PREZENTACJE