

VIII Konferencja PLOUG
Kościelisko
Październik 2002

Oracle

– „Nie możesz jej złamać, nie możesz się włamać” – fakt czy chwyt marketingowy

Wojciech Dworakowski

wojtekd@ipsec.pl

Oracle – „Nie możesz jej złamać, nie możesz się włamać” – fakt czy chwyt marketingowy

W listopadzie 2001, Oracle rozpoczęło intensywną kampanię marketingową, pod hasłem: „Oracle 9i – The Unbreakable: You can't break it, you can't break in”, czyli „Nie możesz jej złamać, nie możesz się włamać”. Tyle mówi hasło reklamowe, a co mówią fakty? Podczas tegorocznego spotkania PLOUG postaram się przedstawić kilka problemów związanych z bezpieczeństwem systemów Oracle i uczulić na nie projektantów i administratorów.

Już na wstępie chcę zaznaczyć, że nie zamierzam odsądzać firmy Oracle od czci i wiary, potępiać posunięć marketingowych i sugerować że DBMS Oracle jest niebezpieczny czy też „dziurawy” jak przysłowiowe rzeszoto. Moim celem jest wykazanie, że każdy produkt jest w większym lub mniejszym stopniu narażony na błędy programistów i projektantów, niezależnie od tego co twierdzą hasła na sztanarach. Oracle 9i to dobry produkt bazodanowy, świadczy o tym chociażby udział w rynku, jednak nawet tak dużym firmom i tak renomowanym produktom zdarzają się wpadki. Należy o tym pamiętać i nie ufać bezgranicznie hasłom reklamowym. Nie bez kozery motto Narodowej Agencji Bezpieczeństwa USA brzmi „Ufamy Bogu – Wszystko inne sprawdzamy”.

Zaraz po rozpoczęciu kampanii „The Unbreakable” podniosły się głosy, że jest to sformułowanie co najmniej mocno przesadzone w przypadku Oracle. Zaraz po premierze 9i nie było oczywiście jeszcze znanych zagrożeń związanych z tą wersją, ale znając kłopoty z bezpieczeństwem jakie miała poprzednia wersja – 8i, większość ekspertów, przez analogie i znając zasady rządzące korporacjami zakładało, że rewolucja w podejściu do bezpieczeństwa raczej nie dokonała się. Jeszcze raz powtórzę – we współczesnej informatyce, nie ma produktów które nie posiadałyby błędów, jednakże w przypadku Oracle 8i charakterystyczna była ich ilość i trywialność¹. Można było stąd wysnuć wniosek, że błędy te wynikają z niewłaściwego podejścia do zagadnień bezpieczeństwa w samej kulturze firmy. Jak się okazało – eksperci się nie mylili.

Hasło „The Unbreakable” podziało jak płachta na byka na społeczność badającą bezpieczeństwo aplikacji i wkrótce w Internecie pojawiły się liczne doniesienia o błędach w Oracle 9i i wynikających z nich zagrożeniach.

Co mówią eksperci i media?

Poniżej kilka cytatów z wypowiedzi ekspertów na temat tego co twierdzi departament marketingu Oracle i jakie to ma pokrycie w rzeczywistości:

CERT – Organizacja zajmująca się bezpieczeństwem IT:

Multiple vulnerabilities in Oracle Application Server and Oracle Database have recently been discovered. These vulnerabilities include buffer overflows, insecure default settings, failures to enforce access controls, and failure to validate input. The impacts of these vulnerabilities include the execution of arbitrary commands or code, denial of service, and unauthorized access to sensitive information.

CERT® Advisory CA-2002-08
Multiple Vulnerabilities in Oracle Servers
<http://www.cert.org/advisories/CA-2002-08.html>

¹ Uczestnicy konferencji PLOUG 2001 mogli się z nimi zapoznać podczas warsztatu „Wybrane metody ataku na systemy Oracle”.

Oracle

Utilizing an Oracle Listener configured with a TCP protocol address, a knowledgeable and malicious user can write an exploit that connects to an Oracle Database server's EXTPROC OS process without having to authenticate himself. As such, he will be able to make arbitrary calls to the underlying OS and potentially gain unauthorized administrative access to the machine hosting the Oracle Database server.

Oracle Security Alert #29, 06 luty 2002
Oracle PL/SQL EXTPROC in Oracle 9i Database

Bruce Schneier – światowej sławy kryptolog i ekspert od bezpieczeństwa IT. Wynalazca szyfrów Blowfish i Twofish. Szef firmy Counterpane zajmującej się zarządzaniem bezpieczeństwem:

"Unbreakable" has a meaning. It means that it can't be broken.(...) I don't care who Larry Ellison is; he can't rewrite the dictionary.

Crypto-Gram Newsletter, 15 luty 2002
<http://www.counterpane.com/crypto-gram-0202.html#6>

Co mówią fakty? – Przykłady zagrożeń

Poniżej przedstawię kilka przykładów zagrożeń obecnych w standardowych instalacjach Oracle 9i/AS. Więcej szczegółów, oraz praktyczne demonstracje tych zagrożeń przedstawię podczas warsztatów – „Metody atakowania systemów Oracle 9i/AS”. Przy każdym z zagrożeń opisuje też krótko jak się przed nimi zabezpieczyć. Temat zabezpieczania systemów Oracle będzie dokładniej przedstawiony na warsztacie „Tuning bezpieczeństwa systemów Oracle”.

Konta i hasła standardowe

Wszystkie produkty spod znaku Oracle instalują bardzo dużą liczbę kont i haseł standardowych. Są to zarówno dobrze znane i udokumentowane konta takie jak konta administracyjne SYS oraz SYSTEM, konta testowe (np. SCOTT, BLAKE, JONES), jak i konta systemowe służące do realizacji określonych zadań. Zbędne konta w każdym systemie mogą stanowić doskonałą boczną furtkę dla intruza. Zwłaszcza gdy nazwy kont i hasła są powszechnie znane.

Co gorsza – producenci aplikacji rozszerzających funkcjonalność programów Oracle, także hołdują temu modelowi. W tej chwili jest znanych ponad **160 kont i haseł** (sic!), które można znaleźć w instalacjach Oracle i produktów towarzyszących.

PL/SQL External Procedures

PL/SQL pozwala na tworzenie wykonywalnych pakietów, które zawierają eksportowane procedury i funkcje. Ponadto funkcjonalność ta może być rozszerzona o wykonywanie funkcji dostępnych w API systemu operacyjnego albo bibliotek dynamicznych.

Żeby korzystać z funkcji i procedur dostępnych przez PL/SQL użytkownik musi posiadać uprawnienie CREATE LIBRARY. Jednakże, da się oszukać Oracle i w rezultacie wywołać dowolną procedurę z dowolnej biblioteki obecnej w systemie.

Szczegóły:

Gdy PL/SQL musi wykonać zewnętrzną procedurę, to proces „oracle” łączy się do Listenera, żąda załadowania odpowiedniej biblioteki i wywołuje pożądaną procedurę przekazując do Listenera parametry. Listener obsługuje to żądanie w ten sposób, że wywołuje nowy proces – extproc i każe procesowi „oracle” komunikować się dalej z tym nowym procesem. Proces „oracle”

łączy się z procesem „extproc” za pomocą mechanizmu *named-pipes* i ponawia żądanie załadowania biblioteki i wywołania procedury.

Problem polega na tym, że nigdzie nie ma autoryzacji procesów wobec siebie. W rezultacie atakujący może podszyć się pod proces `oracle` i wywołać dowolną procedurę z dowolnej biblioteki w systemie. Tą procedurą może być np. `system()` za pośrednictwem której można wywołać dowolny program zewnętrzny.

Mało tego – da się zmusić Oracle do tego, żeby zamiast komunikować się za pomocą mechanizmu *named-pipes* (który umożliwia tylko komunikację wewnątrz jednego systemu) używał socketów TCP.

W rezultacie intruz może zdalnie wykonać dowolny program na systemie z uprawnieniami jakie ma instancja Oracle (na Windows – SYSTEM, na unixach – oracle), bez konieczności posiadania konta w systemie.

Zabezpieczenie:

Podstawową obroną jest zablokowanie ruchu na port listenera (1521/tcp) na firewallach. Obroni to system przed tym i przed wieloma innymi atakami z zewnątrz.

Warto też ograniczyć liczbę systemów, które mogą łączyć się do Listenera. Robi się to modyfikując plik `$ORACLE_HOME\network\admin\sqlnet.ora` :

```
tcp.validnode_checking = YES
tcp.invited_nodes = (adres_ip1, adres_ip2, itd)
```

Jeżeli funkcjonalność PLSExtproc jest niepotrzebna, to należy ją wyłączyć, usuwając odpowiednie wpisy w `tnsnames.ora` i `listener.ora`. Dodatkowo - można też skasować program `extproc` (nie będzie potrzebny).

Listener – atak przez przepełnienie bufora wejściowego

W wielu programach wchodzących w skład Oracle 9i/AS jest możliwe przepełnienie buforów wejściowych. Ta klasa ataków polega na podaniu do atakowanego modułu bardzo długiego ciągu danych w miejscu w którym aplikacja spodziewa się krótkiego ciągu. Jeżeli aplikacja nie sprawdza długości wprowadzonych danych, to jest możliwe nadpisanie pamięci i zakłócenie działania programu. Często prowadzi to też do możliwości wykonania dowolnego kodu wprowadzonego przez intruza.

Tego typu błąd istnieje m.in. w Listenerze. Listener przyjmuje zlecenia od innych procesów za pomocą protokołu TNS. Jednym z parametrów obecnych w każdym wywołaniu TNS jest parametr `SERVICE_NAME`. Jeśli atakujący poda bardzo długą i odpowiednio sformatowaną wartość tego parametru, to będzie on w stanie wywołać dowolny kod na systemie na którym działa Oracle. Kod ten zostanie wywołany z uprawnieniami procesu Listenera (SYSTEM – na Windows, Oracle na unixach).

Zabezpieczenie:

Poprawka numer 2367681.

Podobne błędy istnieją w innych częściach składowych Oracle. Np. w module PL/SQL do serwera Apache, w procedurach obsługujących help i w Oracle 9iAS Report Server (program `rwcgi60`). Na większość tych zagrożeń istnieją odpowiednie poprawki dostępne na Metalinku.

Oracle JSP w Oracle 9iAS

Serwer WWW – Apache, który jest zastosowany w Oracle 9i/AS ma kilka metod komunikowania się z bazą i ze środowiskiem zewnętrznym. Metody są używane do konstruowania aplikacji, które są udostępniane przez 9i Application Server. Jedną z tych metod są JSP (Java Server Pages). Funkcjonalność JSP została zaimplementowana przez Oracle, przez dodanie specjalnego modułu

do serwera WWW Apache. Moduł ten posiada kilka błędów, które stanowią poważne zagrożenie dla bezpieczeństwa serwera aplikacji uruchomionego na tym oprogramowaniu.

Gdy użytkownik prosi o stronę JSP, strona ta jest przetwarzana przez serwer 9iAS, (a konkretnie przez moduł OracleJSP) w następujący sposób:

- jest tłumaczona
- kompilowana
- wynikowy program jest wykonywany
- jego rezultaty są zwracane do użytkownika w postaci HTML

Podczas tego procesu są tworzone są pliki tymczasowe. Jeżeli plik JSP nazywał się np. katalog.jsp, to w podkatalogu /_pages zostaną utworzone następujące pliki:

```
_katalog$__jsp_StaticText.class  
_katalog.class  
_katalog.java2
```

Przetłumaczony plik _katalog.java zawiera kod źródłowy aplikacji JSP. Co gorsza pliki te są dostępne bezpośrednio.

W rezultacie intruz jest w stanie pozyskać kod źródłowy aplikacji JSP udostępnianej przez serwer. Stwarza to ryzyko poznania przez intruza cennych informacji, które bywają umieszczane bezpośrednio w kodzie, takich jak nazwy użytkowników i hasła, oraz daje dostęp do całej logiki biznesowej atakowanej aplikacji, co może prowadzić do dalszej penetracji.

W podobny sposób intruz może przejrzeć zawartość pliku ustawień global.jsa

Zabezpieczenie:

Przed tego typu atakiem można się obronić przez skonfigurowanie samego serwera Apache w ten sposób, żeby nie wydawał plików z katalogu /_pages oraz pliku o nazwie global.jsa

Komentarz

Jak widać w standardowej instalacji Oracle istnieją błędy. Na dodatek rodzaj tych błędów wskazuje na brak stosowania zasad bezpiecznego programowania (błędy związane z przepełnieniem bufora) czy wręcz na niestosowanie podstawowych pryncypiów bezpieczeństwa aplikacji przy projektowaniu architektury niektórych modułów (brak uwierzytelnienia procesów).

Tak więc w tym wypadku fakty znacząco odbiegają od haseł reklamowych. Zastanawiające jest wręcz co skusiło departament marketingu do przygotowania hasła tak dalece odbiegającego od rzeczywistości? Czytając wypowiedzi Larego Ellisona³ mam wrażenie że szefostwo Oracle i departament marketingu święcie wierzyło w to że ich aplikacje są ultra-bezpieczne. Cóż – paradoksalnie – w dużych korporacjach taki brak informacji jest całkiem możliwy.

Tak czy inaczej – czas i niezależni badacze (głowie David Litchfield i Pete Finigan) zweryfikowały hasła marketingowe.

A co na to Oracle?

Osobnego komentarza wymaga odpowiedź Oracle na oczywiste fakty którym nie da się zaprzeczyć. Otóż – Pani Mary Ann Davidson – „Chief security officer” w Oracle, stwierdziła, że słowo

² Schemat nazewnictwa tych plików jest opisany tutaj:

http://download-west.oracle.com/tndoc/oracle9i/901_doc/java.901/a90208/trandep1.htm

³ Np. wywiad dla Wprost – dodatek Intermedia:

<http://intermedia.wprost.pl/php/intermedia/index.php3?sz=1&id=11708>

Chciałem zwrócić redakcji uwagę na fakty, które przeczą wypowiedziom Elisona i wysłałem list otwarty do redakcji (dostępny jest pod adresem: <http://arch.ipsec.pl/art/wojtekd.html>), ale nie dostałem żadnej odpowiedzi ;)

„Unbreakable” nie mówi o tym że do systemów Oracle nie da się włamać (sic!), lecz o tym, że Oracle 9i przeszedł 14 niezależnych testów bezpieczeństwa. To są fakty podawane przez Oracle http://www.oracle.com/ip/dep/loay/database/oracle9i/index.html?se_dbcomp.html

Ale co to oznacza w praktyce? Co to za 14 niezależnych testów? Na czym one polegały? Czy można im ufać skoro nie wykryto podczas nich tak podstawowych błędów jak np. przepełnienie bufora (technika znana od roku 1996)?

Bruce Schneier z firmy Counterpane postanowił sprawdzić co to za 14 niezależnych sprawdzianów bezpieczeństwa. Niestety okazało się, że jest ich tylko pięć: TCSEC, ITSEC, Common Criteria, Russian Criteria i FIPS 140-1. Marketing Oracle rozmnożył je do 14, przez policzenie różnych poziomów ewaluacji TCSEC i ITSEC jako niezależnych testów. Na dodatek ilość testów dotyczy różnych wersji Oracle w sumie (Oracle Database, Trusted Oracle, Oracle Advanced Security). Poza tym wszystkie dotyczą wersji 7, 8 i 8i a nie reklamowanej wersji 9i oraz AS.

Podsumowanie

Mam nadzieję, że powyższy artykuł i wykład mu towarzyszący uzmysłowi państwu, że nie warto wierzyć hasłom reklamowym. Zwłaszcza w tak delikatnej materii jak bezpieczeństwo. Tym bardziej że jest to coś mierzalnego, coś co da się sprawdzić.

Niezależnie od tego z jak zaawansowanym i rozwiniętym produktem mamy do czynienia warto sprawdzić jego bezpieczeństwo. Proszę też pamiętać o tym, że bezpieczeństwo samych programów Oracle to nie wszystko. Nic nam nie da poprawnie zabezpieczony serwer, bezpiecznie skonfigurowane środowisko Oracle, firewalle i systemy IDS, jeżeli błąd popełnimy w naszej aplikacji działającej na platformie Oracle. Co gorsza o tych błędach nikt nie napisze w Internecie i nie opracuje za nas poprawki. A zapewniam Państwa, że te błędy – pomimo tego, że unikalne – dają się zidentyfikować i wykorzystać...

W Internecie:

<http://www.counterpane.com/crypto-gram-0202.html#6>
<http://www.nextgenss.com/advisories/oraplsxtproc.txt>
http://technet.oracle.com/dep/loay/security/pdf/ias_modplsql_alert.pdf
<http://www.nextgenss.com/advisories/oraplsbos.txt>
<http://www.nextgenss.com/advisories/orajsp.txt>
<http://www.cert.org/advisories/CA-2002-08.html>
<http://arch.ipsec.pl/art/wojtekd.html>
<http://www.securityfocus.com/columnists/45>
<http://news.com.com/2100-1001-831142.html>