

# **Biuletyn Bezpieczeństwa PLOUG – Podsumowanie 2003. Przegląd zagrożeń dla bezpieczeństwa produktów Oracle, które ujrzały światło dienne w ostatnim roku**

*Wojciech Dworakowski*

SecuRing

W Biuletynie Polskiej Grupy Użytkowników Systemów Oracle - PLOUG'tki, od numeru 24 ukazuje się stała rubryka - Biuletyn Bezpieczeństwa PLOUG. Celem biuletynu jest przybliżanie administratorom zagrożeń związanych z produktami Oracle, które zostały ujawnione w ostatnich miesiącach. •ródłem dla publikowanych tam informacji są nie tylko dokumenty Oracle Security Alert, ale przede wszystkim źródła niezależne, często związane z osobami, które wykryły opisywane zagrożenia.

Podczas wykładu zostaną przedstawione najważniejsze zagrożenia dla bezpieczeństwa produktów Oracle, które ujrzały światło dzienne w ostatnim roku. Duży nacisk zostanie położony na praktykę administracyjną, tzn. - kiedy dane zagrożenie się uaktywnia, jak wielkie ryzyko się wiąże z danym zagrożeniem, czy istnieje publicznie dostępny exploit, jakie metody dodatkowe i obejścia można zastosować by ustrzec się przed podobnymi zagrożeniami w przyszłości.

Ponadto zostaną przybliżone typowe podatności takie jak np. buffer overflow, które często ujawniają się w produktach Oracle i innych.

## **Informacja o autorze:**

Konsultant bezpieczeństwa IT w firmie SecuRing. Koordynator prac zespołu i osoba odpowiedzialna za sporządzanie raportów. Siedem lat doświadczenia praktycznego w zakresie bezpieczeństwa IT. Od trzech lat zajmuje się również testowaniem bezpieczeństwa produktów Oracle. Prelegent na licznych konferencjach poświęconych bezpieczeństwu IT (m.in. CERT Secure, PLOUG, Open Source Security, Windows Security). Prowadzi rubrykę poświęconą bezpieczeństwu w Biuletynie PLOUG.

W Biuletynie Polskiej Grupy Użytkowników Systemów Oracle – PLOUGtka, od numeru 24, a więc od blisko roku ukazuje się stała rubryka – Biuletyn Bezpieczeństwa PLOUG. Celem Biuletynu jest dostarczanie rzetelnej i praktycznej informacji na temat zagrożeń dla produktów Oracle, które ujrzały światło dzienne w ostatnim czasie. Podczas wykładu przedstawię kilka najistotniejszych zagrożeń, jakie były opisywane w ostatnim roku.

Na początek – kilka słów o źródłach informacji zamieszczanych w Biuletynie Bezpieczeństwa PLOUG. Bazą do konstruowania informacji o błędach i zagrożeniach występujących w produktach Oracle są Oracle Security Alerts (<http://otn.oracle.com/deploy/security/alerts.htm>) oraz informacje od innych niezależnych badaczy bezpieczeństwa Oracle.

Problemom bezpieczeństwa produktów Oracle przyglądam się od dłuższego czasu. Zadowolająca jest niewątpliwie zauważalna poprawa sposobu podejścia producenta do problemów bezpieczeństwa Oracle w ciągu ostatnich kilku lat. Jednakże można również wyraźnie zauważyć, że większość odkryć i doniesień o nowo odkrywanych „dziurach” nie pochodzi bezpośrednio od samego Oracle i nie jest wynikiem działania jakiegoś zespołu badaczy wewnątrz struktur firmy Oracle. Obecnie większość tego typu odkryć jest dokonywana przez badaczy niezależnych. Od 2 lat postacią zdecydowanie dominującą na tym polu jest David Litchfield i jego firma – NGSSoftware (<http://www.nextgenss.com/>). Właśnie z tego względu, informacje do Biuletynu Bezpieczeństwa PLOUG staram się pobierać bezpośrednio „u źródła” i korzystam przede wszystkim z informacji publikowanych przez odkrywcę danej luki. Dzięki temu mogę przekazać znacznie więcej szczegółów technicznych niż zawierają Oracle Security Alerts.

Głównym celem Biuletynu Bezpieczeństwa, jest pomoc administratorom Oracle w ocenie, czy dany błąd może dotyczyć ich instalacji i czy stanowi w ich konkretnym środowisku duże zagrożenie. Informacje publikowane w Biuletynie mają pomóc w ocenie ryzyka i podjęciu decyzji o tym kiedy należy zainstalować poprawkę lub podjąć kroki ograniczające ryzyko. Natychmiast, w następnym cyklu wprowadzania poprawek czy – być może – wogóle?

## Przegląd nowych podatności w produktach Oracle

Poniżej przedstawię kilka zagrożeń dla różnych produktów Oracle jakie ujrzały światło dzienne w ostatnim roku. Starałem się wybrać zagrożenia wiążące się z największym ryzykiem oraz zagrożenia ciekawe z punktu widzenia konsultanta bezpieczeństwa IT. Na ich przykładzie postaram się przybliżyć problemy związane z odkrywaniem nowych podatności i szacowaniem ryzyka z nimi związanego.

### Zawieszenie Listenera przez komendę **SERVICE\_CURLOAD**

Błąd dość trywialny, łatwy do wykorzystania i zrozumienia.

Oracle Listener posługuje się protokołem TNS (Transport Network Substrate). Jest to stosunkowo prosty protokół posługujący się zleceniami przesyłanymi w trybie ASCII. Protokół TNS operuje kilkoma komendami. Interesująca dla nas jest komenda **SERVICE\_CURLOAD**. Jest to komenda nieudokumentowana przez Oracle i nie jest znane mi jej przeznaczenie. Jedynym skutkiem wydania takiej komendy do zdalnego Listenera jest to, że Listener obciąża nagle procesor w 100%. Wydanie kilku takich zleceń pod rząd, z reguły prowadzi do całkowitego zawieszenia procesu Oracle Listener oraz maksymalnego obciążenia maszyny na której jest uruchomiony Listener.

Do przeprowadzenia skutecznego ataku wystarczy, żeby atakujący posiadał możliwość bezpośredniej komunikacji z portem TCP Oracle Listener (standardowo 1521). Tak więc w większości przypadków atak może być wykonany z sieci LAN w której jest obecny atakowany serwer.

Zagrożenie jest obecne w prawie wszystkich wersjach bazy Oracle 8 i 9 (łącznie z 9.2). Błąd usuwa nałożenie poprawki numer 2467947 i 2540219 (w zależności od wersji i platformy).

Pełen opis podatności znalazł się w Biuletynie Bezpieczeństwa w 24 numerze PLOUGtek: <http://www.ploug.org.pl/gazetka/24/11.htm>.

## Niebezpieczna konfiguracja standardowa WebDAV w Oracle Application Server

Błąd wynikający z nieprawidłowej konfiguracji standardowej jednego z modułów. Charakterystyczne jest to, że podatny moduł nie jest potrzebny do normalnej pracy serwera a nawet śmiem zaryzykować stwierdzenie, że funkcjonalność z nim związana jest stosunkowo rzadko wykorzystywana w instalacjach 9iAS.

Oracle Application Server 9i a dokładniej serwer HTTP Apache na którym bazuje serwer HTTP wchodzący w skład 9iAS, wspiera WebDAV (Web Distributed Authoring and Versioning). W uproszczeniu - rozszerzenie to pozwala na korzystanie z serwera WWW, tak jak z serwera plików. WebDAV jest włączone w instalacji standardowej. Błędna konfiguracja standardowa powoduje że dowolny anonimowy użytkownik, który ma dostęp do serwisu WWW obsługiwanego przez 9iAS, może wgrzywać dowolne pliki na serwer do katalogu /dav\_public.

Błąd jest obecny w wersji 9.0.2. Jego usunięcie polega na wyedytowaniu pliku konfiguracyjnego \$ORACLE\_HOME/Apache/oradav/conf/moddav.conf i zmianie wpisu „DAV on” na „DAV off” dla katalogu /dav\_public. Jeśli funkcjonalność WebDAV jest nieużywana to zaleca się wyłączenie stosownego modułu w pliku konfiguracyjnym Apache: \$ORACLE\_HOME/Apache/Apache/conf/oracle\_apache.conf.

Pełen opis podatności znalazł się w Biuletynie Bezpieczeństwa w 26 numerze PLOUGtek: <http://www.ploug.org.pl/gazetka/26/11.htm>.

## Zdalne pozyskanie kodu źródłowego stron JSP

Celem tego ataku nie jest bezpośrednio przejęcie kontroli nad serwerem, lecz pozyskanie dodatkowej, cennej informacji, która może posłużyć do o wiele łatwiejszego odnajdywania błędów w aplikacjach bazujących na JSP.

Błąd w module odpowiadającym za obsługę skryptów JSP pozwala na nieautoryzowane pozyskanie kodu źródłowego skryptów. Manipulując w odpowiedni sposób nazwą i rozszerzeniem skryptu JSP w zleceniu HTTP, intruz może spowodować, że serwer źle interpretuje jego zlecenie. Serwer nie rozpoznaje, że wydawany plik jest skryptem JSP, który należy wykonać i zwrócić rezultat wykonania skryptu. Serwer zwraca źródło JSP a nie rezultat jego wykonania.

Błąd jest usuwany przez upgrade do wersji 9.0.2.0.1 bądź wyższej.

Pełen opis podatności znalazł się w Biuletynie Bezpieczeństwa w 25 numerze PLOUGtek: <http://www.ploug.org.pl/gazetka/25/9.htm>.

## Błędy w E-Business Suite

Rok 2003 obfitował w doniesienia związane z Oracle E-Business Suite. Przedtem informacje o błędach w tym oprogramowaniu pojawiały się bardzo sporadycznie. Spowodowane to jest tym, że dopiero w tym roku produkt ten został poddany fachowej i wnikliwej ocenie bezpieczeństwa, przez jednego z badaczy. Jest nim Stephen Kost z firmy Integrigy.

Przykład Oracle E-Business Suite świetnie pokazuje, że fakt iż w danym oprogramowaniu nie wykryto większych błędów wcale nie świadczy o tym, że jest ono bezpieczniejsze od konkurencji. Moim zdaniem, jest raczej na odwrót. To że producent publikuje na bieżąco informacje o odkrywanych błędach świadczy o tym, że ktoś się tym zajmuje, odkrywa błędy i są one na bieżąco eliminowane. Niestety – nie ma współczesnego oprogramowania bez błędów. Brak doniesień dotyczących danego produktu powinien być raczej sygnałem ostrzegawczym dla administratorów. Pamiętajmy, że informacja o istnieniu błędu dużo wcześniej jest rozpowszechniana w środowisku hackerów.

Więcej informacji o błędach odkrytych w E-Business Suite znajduje się w 27 numerze PLOUGtek: <http://www.ploug.org.pl/gazetka/27/13.htm>.

## Błąd w kodzie zabezpieczającym przed atakami na bazy Oracle (EXT-PROC)

Podwójnie ciekawa podatność. Po pierwsze – został wykryty błąd w kodzie zabezpieczającym przed innym błędem. Dotychczas powszechnie była znana podatność obecna w każdej standardowej instalacji Oracle. Wiązała się ona z modułem odpowiedzialnym za wykonywanie External Procedures. Problem był jak dotychczas nie do usunięcia, gdyż wynikał z błędnie przyjętego projektu. Dokładniej – chodzi o to, że w momencie wywołania procedury zewnętrznej, przez Oracle Listener, Listener wywołuje proces Extproc i przekazuje do niego parametry wywołania. Błąd polega na tym, że między tymi procesami nie ma żadnego uwierzytelnienia. W związku z tym intruz może podpiąć się bezpośrednio do procesu Extproc, udąć Oracle Listener i wywołać dowolną procedurę zewnętrzną. Błąd ten jest znany od dłuższego czasu i jest dobrze udokumentowany.

W wersji bazy 9.2.0.2 i 9.2.0.3, Oracle wprowadziło nowy mechanizm polegający na tym, że próby załadowania biblioteki zewnętrznej są logowane i odrzucane, chyba że wywołanie pochodzi z tego samego serwera. W kodzie obsługującym to logowanie, popełniono błąd, który umożliwia klasyczny atak przez przepełnienie bufora wejściowego (buffer overflow). Co za tym idzie – możliwe jest wykonanie dowolnego kodu na serwerze, z uprawnieniami użytkownika systemowego będącego właścicielem instalacji Oracle (użytkownik "oracle" na unixach lub LOCAL\_SYSTEM na Windows) i przejęcie kontroli nad instalacją Oracle.

Drugą ciekawostką związaną z tą podatnością jest to, że pierwotnie w Oracle Security Alert #57 z dn. 23.07.2003 (<http://otn.oracle.com/deploy/security/pdf/2003alert57.pdf>), producent ocenił zagrożenie związane z omawianym błędem jako niskie. Wynikało to z fałszywego przekonania, iż błąd ten może wykorzystać tylko użytkownik posiadający przywileje "CREATE LIBRARY". Istotnie – tak było w przypadku pierwotnego błędu, jednak wprowadzenie błędu w poprawce doprowadziło do tego że tym razem błąd mógł zostać wykorzystany przez atakującego działającego bez żadnych początkowych przywilejów. Odkrywczy opisywanej luki - David Litchfield i Chris Anley, wielokrotnie podkreślali, że firma Oracle źle oceniła zagrożenie i może ono być wykorzystane przez intruza nie posiadający nawet żadnego konta w bazie. Jako dowód swojej tezy, zaprezentowali taki atak w praktyce, podczas konferencji Black Hat Briefings Europe 2003. Ostatecznie Oracle zgodziło się ze stanowiskiem odkrywców błędu i zmieniło wycenę błędu na „wysokie”, jednakże nastąpiło to po długiej wymianie zdań, dopiero 04.09.2003. Co ciekawe informacja o zmianie oceny ryzyka z „Low” na „High” nie jest uwzględniona w historii zmian dokumentu (rozdział „Modification History”).

## Powszechnie dostępny exploit pozwalający na przejęcie kontroli nad instalacją Oracle

W ostatnim czasie (po publikacji ostatniego Biuletynu) została upubliczniona informacja o kolejnej serii błędów typu buffer overflow w bazie Oracle. Tym razem jednak odbyło się to w nieco inny sposób niż zazwyczaj oraz co istotniejsze został udostępniony kod (exploit) pozwalający na skuteczne atakowanie baz Oracle.

Istotą ataku wykorzystującego podatność typu buffer overflow jest napisanie programu, który komunikując się z atakowanym programem, będzie w stanie przepełnić bufor przeznaczony na którąś ze zmiennych i w dalszej kolejności nadpisać adres powrotu z procedury na stosie procesora. Stworzenie exploita, to zadanie dość złożone, wymaga sporej wiedzy, dysponowania instalacją testową atakowanego programu uruchomioną pod debuggerem, dużej ilości czasu oraz umiejętności programowania w językach niskopoziomowych. Jeśli atakujący dysponuje exploitem – gotowym programem, to jego wiedza techniczna może być ograniczona do minimum. W związku z powyższym, z reguły nie udostępnia się exploitów, chyba że w celach edukacyjnych.

Na konferencji Black Hat Briefings 2003, David Litchfield przedstawił referat w którym omawiał różnice w atakowaniu i pisaniu exploitów na platformy Linux i Windows („Variations in Exploit methods between Linux and Windows”, <http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-litchfield-paper.pdf>). Przykładowym atakowanym programem była właśnie baza

Oracle w wersji 9.2, a dokładniej moduł XDB (XML Database). Częścią artykułu towarzyszącego prezentacji są przykłady exploitów na platformy RedHat Linux i Windows.

Oracle nadało zagrożeniu wiążącemu się z tą podatnością priorytet „Wysokie”. Informacje o błędzie są dostępne w Oracle Security Alert 58 (<http://otn.oracle.com/deploj/security/pdf/2003Alert58.pdf>). Błąd jest usuwany przez nałożenie poprawki nr 3058991.

## Jak chronić się przed skutkami błędów obecnych w Oracle

W większości wypadków wystarczy po prostu zastosować się do zaleceń producenta. Nałożyć dostępne poprawki, lub zastosować proponowane obejścia problemu. Wydaje się to oczywiste jednak, jak zdążyłem się zorientować administratorzy Oracle dość niechętnie nakładają poprawki na systemy które mają pod swoją opieką. Według obiegowej opinii – nakładanie poprawek (pachy) wiąże się z ryzykiem destabilizacji systemu. Według mnie – tego typu ryzyko występuje w większości współczesnych systemów informatycznych i nie dotyczy wyłącznie Oracle. Skutecznym i praktycznym rozwiązaniem jest wprowadzenie procedur zarządzania zmianami. W wielkim skrócie: zwykle proces zarządzania zmianami składa się z następujących etapów:

1. Zdobywanie informacji o nowych podatnościach.
2. Ocena ryzyka związanego z podatnością, w środowisku eksploatowanego systemu.
3. Testowanie poprawek.
4. Instalowanie poprawek w cyklach, w zależności od stopnia ryzyka.

Dokładniej te problemy opisałem w artykule „Dylematy administratora bezpieczeństwa Oracle”, który został opublikowany w 27 numerze PLOUG’tek i jest dostępny pod adresem: <http://www.ploug.org.pl/gazetka/27/12.htm>.

Drugim sposobem unikania zagrożeń zanim przerodzą się w konkretne ryzyko jest ograniczanie możliwości wystąpienia błędu przez eliminację zbędnej funkcjonalności. Przykładowo – jeśli w instalacji Oracle9iAS nie korzystamy z modułu WebDAV czy XDB, to należy je wyłączyć. W ten sposób nawet jeśli w przyszłości nowo odkryte błędy w tych modułach pozwolą na przejęcie kontroli nad serwerem aplikacji, to problem ten nie będzie nas dotyczyć.