Oracle Unbreakable

Oracle en security (4)

Een van de reclame-slogans waarmee Oracle9i geïntroduceerd werd, was "Oracle Unbreakable". Toch stond er op 3 augustus jongstleden een artikel in de Wall Street Journal, getiteld 'Security flaws found in Oracle software', waarin melding wordt gemaakt van 34 onvolkomenheden op het gebied van security, die gevonden zijn door David Litchfield. Bij het noemen van een dergelijk aantal onvolkomenheden ga je er eigenlijk bij voorbaat van uit dat de ontdekker hiervan de reclame slogan 'Oracle Unbreakable' niet alleen onwaar, maar op zijn minst ook misleidend zal vinden. Toch blijkt dit niet het geval te zijn.

In een artikel van 10 januari 2002 getiteld 'Hackproofing Oracle Application Server' stelt David weliswaar dat Oracle 'breakable' is maar dat de reclame slogan waarschijnlijk mee begrepen moet worden als een bewijs voor Oracle's commitment tot het leveren van een veilig product. Hij roemt hierbij de veertien onafhankelijke security assessments en geeft aan dat er onder de concurrenten geen enkel product is dat wat het security aspect betreft ook maar in de buurt komt. David Litchfield is overigens een van de oprichters van NGS, een bedrijf dat gespecialiseerd is het maken van software waarmee andere software (zoals bijvoorbeeld een Oracle database) kan worden doorgelicht op security risico's.

Onvolkomenheden

Het artikel trekt veel aandacht onder automatiseerders. Er is bijna geen automatiseringsblad te vinden of het artikel wordt in enige vorm geciteerd. Bij het nalezen van al deze artikelen blijkt echter dat er niet veel concrete informatie te vinden is over de exacte aard van deze beveiligingsproblemen. Je zou juist verwachten, dat de wat meer technisch georiënteerde tijdschriften zouden aangeven op welk gebied deze onvolkomenheden zich precies voordoen. Niets is echter minder waar: slechts enkele artikelen melden dat het onder andere problemen zijn die te maken hebben met buffer overflow, maar daar blijft het dan ook bij. Er staat nergens welke problemen het nu precies zijn en hoe men zich er tegen kan wapenen. Over dit laatste is niets anders bekend dan de mededeling van Oracle, dat ze druk bezig zijn met het vervaardigen van patches en de schatting dat dit een paar maanden kan duren. Volgens Hans Bos, marketing manager Technology van Oracle Nederland, heeft de duur van het vervaardigen van dergelijke patches in het verleden van vier tot maximaal negen maanden gevarieerd. Als verklaring voor deze doorlooptijd wordt aangegeven dat Oracle op veel platforms draait en dat een voortijdige publicatie van een patch voor één platform de aanleiding kan zijn voor een aanval op de andere platforms. De publicatie van een patch betekent immers dat er informatie beschikbaar komt over het veiligheidslek dat ermee gepatched wordt.

Slammer

Dat dit laatste een overweging is die serieus genomen moet worden, blijkt uit het optreden van de Sapphire/Slammer-worm van begin vorig jaar. In een artikel genaamd 'Expert weighs code release in wake of Slammer worm' van Paul Roberts in IDG News in januari van dit jaar wordt aangegeven dat code van de Slammer worm gebaseerd was op een publicatie over het security lek waarvan de worm gebruik maakte. Deze code kwam van David Litchfield, dezelfde die nu de Oracle security leaks heeft gevonden.

Het aantal geinfecteerde computers verdubbelde in het begin elke 8,5 seconde

'Bij analyse van de code van de Slammer worm is het duidelijk dat mijn code als een template is gebruikt' was het citaat van Litchfield. Bij een nadere analyse bleek overigens ook dat de schrijver van de worm wel degelijk zelf verstand moet hebben van de technieken om wormen te schrijven. (ook in dit geval werd gebruik gemaakt van een overflow van een buffer).

Afbeelding UC Regents

In januari 2003 werden binnen 10 minuten 75.000 machines geïnfecteerd met de Slammer-worm

Volgens een schatting van Litchfield heeft de code, die hij als 'proof-of-concept' heeft gepubliceerd, de worm-schrijver hooguit twintig minuten bespaard.
Die twintig minuten, die de ene deskundige op het gebied van wormen (David Litchfield), de andere (de worm-schrijver) heeft bespaard, interesseert natuurlijk niemand. Het werkelijke probleem is echter het publiceren van het lek, compleet met specificaties. Direct na het optreden van de Slammer-worm stelde Litchfield dan ook, dat hij waarschijnlijk niet langer dergelijke code zou publiceren, gezien de schade die de worm had aangericht. Later is hij daar overigens weer op teruggekomen.

Bad guys

De code werd gepubliceerd tijdens een conferentie van security officers (de Blackhat Briefings). De bedoeling van deze conferenties is dat de 'good guys' op de hoogte blijven van wat de 'bad guys' uitspoken en dat ze in staat zijn om indien er nieuwe wormen en virussen ontstaan, deze zo snel mogelijk onschadelijk te maken. Dat dit ook inderdaad zo werkt blijkt uit de gebeurtenissen rondom de Slammer-worm[1]:

Juli 2002
Het security leak wordt ontdekt en pas gepubliceerd nadat Microsoft er een patch voor heeft gereleased.

Najaar 2002
De Blackhat Briefing waarop de bewuste code wordt besproken.

Zaterdag 25 januari 2003, iets voor 05:30 UTC
De verspreiding van de worm begint. Hierna gaat het snel. Het aantal geïnfecteerde computers verdubbelt in het begin ongeveer elke 8,5 seconde. Na ongeveer 3 minuten is de verspreidingssnelheid maximaal. Binnen 10 minuten zijn alle kwetsbare machines (die aan het internet hangen en niet gepatched waren) geïnfecteerd. Dit waren overigens meer dan 75.000 machines.

Zondag 26 januari 2003, 04:49
De verspreiding van een scanner om de problemen te kunnen inventariseren. De reden dat deze zo snel voorhanden was moge blijken uit een bijschrift van een scanner om het virus te identificeren en onschadelijk te maken: 'Dank aan NGSSoftware (N.B. het bedrijf van David Litchfield) voor het ontdekken van de onvolkomenheid waar de SQL-worm gebruik van maakt en voor het publiceren van een technische beschrijving hiervan, zonder welke deze scanner niet mogelijk geweest was.' (Marc Maiffret)

Een aantal conclusies uit bovenstaand tijdschema:
- De security leak is pas gepubliceerd nadat er een patch voor beschikbaar was.
- De code waarop de worm gebaseerd is, werd pas daarna besproken op de Blackhat Briefing.
- De code heeft ook geholpen om de worm zo snel mogelijk te kunnen bestrijden.

Binnen de security-wereld is het een algemeen aanvaarde praktijk, dat het lek pas gepubliceerd wordt nadat er een patch voor beschikbaar is gekomen. Dit staat dan ook los van de in het begin van het artikel genoemde overweging om wel of niet code als proof-of-concept te publiceren. Naar mijn mening heeft de internet gemeenschap er meer baat bij gehad, dat de problemen van de Slammer-worm snel en efficiënt konden worden aangepakt, dan dat de schrijver van de Slammer-worm hem iets sneller heeft kunnen schrijven. Gezien de technische expertise die nodig was om deze worm te schrijven, is het hebben van een stukje voorbeeldcode geen overweging geweest die bepalend was om er al of niet aan te beginnen.

"Bij analyse van de code van de Slammer wordt duidelijk, dat mijn code als een template is gebruikt"

Zware straffen

Er zijn een aantal zaken die bijdragen tot het ontstaan van wormen en virussen:
1. Software schrijvers maken fouten
2. Er zijn mensen die het spannend vinden wormen en virussen te schrijven
3. Er bestaan bekende security leaks
4. Bedrijven zijn laks met het aanbrengen van security patches

Aan de eerste drie punten kunnen eindgebruikers weinig tot

1 *The Spread of the Sapphire/Slammer Worm*
David Moore, Vern Paxon, Stefan Savage, Colleen Shannon, Stuart Staniford en Nicolas Weaver

niets bijdragen, maar het vierde hebben ze in eigen hand. Ook aan de andere drie punten kan wat gedaan worden:

Ad 1) Software schrijvers kunnen worden gewezen op de technieken waarmee virusschrijvers hun software misbruiken. Hierdoor zullen ze software leveren met minder security leaks. Dit is met name het doel van bijeenkomsten als de Blackhat Briefings.

Ad 2) Maak het onaantrekkelijk door een hoge pakkans en zware straffen. In iedere geval zwaar genoeg om duidelijk te maken dat het geen spelletje is.

Ad 3) Hoe paradoxaal het ook klinkt: indien bestaande security-patches ook inderdaad aangebracht worden, is het onderzoeken van software op security leaks een bezigheid die de veiligheid verhoogt. Hoe meer er bekend zijn, des te complexer zullen de overgebleven lekken zijn en des te moeilijker voor een hacker om ze te vinden. Maar dit staat of valt met het aanbrengen van bestaande security-patches. Indien dit niet gebeurt, wordt de situatie onveiliger. De hacker hoeft er zelf geen meer te zoeken.

Voorbeeldcode

Samenvattend denk ik te kunnen stellen, dat niet het publiceren van een stukje voorbeeldcode geleid heeft tot de Slammer-worm (met al zijn kwalijke effecten), maar de laksheid van de internetgemeenschap om bestaande security-patches aan te brengen, met name degenen die zich professioneel op het internet begeven. De Slammer-worm kon nog beschouwd worden als een relatief onschuldig virus. Het deed niets anders dan zich te verspreiden. Alleen de snelheid waarmee dit gebeurde zorgde ervoor dat er complete netwerken uitvielen, met veel onaangename gevolgen: uitvallen van vluchten van luchtvaartmaatschappijen, uitvallen van geldautomaten, problemen bij verkiezingen et cetera. Uit de gebruikte technieken blijkt overigens wel dat deze snelheid bewust nagestreefd is. Vooralsnog mijn dank aan David Litchfield voor het nog niet publiceren van de technische beschrijving van de vierendertig security leaks. Maar ik hoop wel dat het hier de komende maanden bij blijft, totdat Oracle de security patches gepubliceerd heeft. Vervolgens hoop ik ook dat wij allen deze dan ook zo snel mogelijk zullen aanbrengen.

Gerard Uiterwaal is Oracle- en security expert en werkzaam bij Motiv IT Masters. Indien u in een van de volgende uitgaven van Optimize graag een specifiek onderdeel belicht zou willen zien dan kunt u dit aangeven via e-mail: gerard.uiterwaal@motiv.nl. De auteur streeft ernaar zoveel mogelijk vragen te beantwoorden.

NIEUWS

Artikelen met praktische informatie, geschreven door en bestemd voor Oracle-professionals vindt u in het Online Archief van Array Publications. Vaktijdschriften als Database Magazine, Software Release en Java Magazine hebben hun artikelenarchief online gezet. Met een heldere zoekstructuur vindt u snel wat u zoekt op www.optimize.nl.

IDC: Bedrijfscontinuïteit in Nederland nog geen gemeengoed

"De effectiviteit van calamiteitenplannen wordt zwaar overschat." Dat is één van de conclusies uit de jaarlijkse IDC Business Continuity Benchmark 2004. In dit jaarlijkse onderzoek onder bijna 300 grote bedrijven staat de vraag centraal hoe het met de bedrijfscontinuïteit van ondernemingen in Nederland gesteld is. Hoe zorgen zij ervoor dat het bedrijf operationeel blijft of snel herstelt onder alle mogelijke omstandigheden? In het onderzoek is binnen de algemene continuïteitsplanning extra aandacht geschonken aan informatiebeveiliging.

Bedrijven zijn in het bedrijfsproces erg afhankelijk geworden van hun IT-systemen. Meest voorkomende bedrijfsschade wordt veroorzaakt door systeemuitval, door welke oorzaak dan ook. 22% loopt al schade op bij systeemuitval van minder dan 1 uur. Dit is niet verwonderlijk als 31% van de bedrijven aangeeft 24 uur per dag beschikbaar te zijn voor klanten. De effectiviteit van het calamiteitenplan, waar informatiebeveiliging een onderdeel van hoort te zijn, wordt echter door veel bedrijven zwaar overschat. Gemiddeld denken de meeste bedrijven dat ze met hun huidige plan na een grote calamiteit binnen één dag weer volledig operationeel kunnen zijn. Tegelijkertijd geeft 38% van de respondenten aan dat de werknemers onvoldoende op de hoogte zijn van wat zij moeten doen volgens datzelfde plan. Het onderzoek toon aan dat de gezondheidszorg het grootste belang heeft bij business continuïteit. Het niveau van maatregelen op het gebied van continuïteit in de zorg blijft echter sterk achter bij andere sectoren als industrie, financiële dienstverlening en IT & telecommunicatie.

Donderdag 30 september aanstaande organiseert IDC in samenwerking met Infosecurity.nl/Storage Expo De Business Continuity Management Track waarin de resultaten van dit onderzoek uitgebreid gepresenteerd worden. Eén van de gastsprekers tijdens dit seminar is Kevin Mitnick, voormalig hacker en CEO en medeoprichter van Defensive Thinking LLC. Meer informatie hierover is te vinden op www.idcresearch.com/benelux.