Oracle en Security (10)

Het gevaar van de Oracle Voyager Worm

Op 31 oktober 2005 werd anoniem melding gemaakt van het bestaan van een Oracle-wormcode. Dat gebeurde door het publiceren van dit stukje PL/SQL-code waarmee het mogelijk is om zo'n Oracle-worm te maken (een zogenaamd proof-of-concept). Op 29 december 2005 werd ontdekt dat iemand dit stukje code ook daadwerkelijk gebruikt heeft om een Oracle-worm te maken. Dit heeft geresulteerd in een tweetal mailtjes, waarin Oracle zijn klanten op de hoogte stelt van het bestaan van deze worm. Tevens heeft Oracle aangegeven hoe men zich tegen deze worm moet beveiligen.

Wat is eigenlijk een computerworm? Een eenvoudige definitie luidt: een programma, dat in staat is om zichzelf te repliceren en dit replica over een netwerk te versturen naar een andere computer, waarna het proces zich kan herhalen. In het geval van een Oracle-worm, gaat een dergelijk programma niet van computer naar computer, maar van Oracle-database naar Oracle-database. Om het potentiële gevaar te beoordelen van een worm is het nodig om eerst te bekijken hoe een worm generiek is opgebouwd. Je kunt een worm beschouwen als een systeem, dat bestaat uit de volgende onderdelen :

- Verkenning
- Specifieke aanvalsmogelijkheden
- Commando interface
- Communicatie
- Gegevens verzamelen
- Potentiële aanvalsmogelijkheden

Deze onderdelen kunnen elkaar overlappen. De indeling is alleen gemaakt om de doelstellingen van de onderdelen van elkaar te kunnen isoleren, waarmee het mogelijk wordt om de functionaliteit van wormen beter te kunnen beschrijven.

Verkenning

Het wormsysteem moet in staat zijn om zijn omgeving te kunnen inventariseren. Aan de hand van deze inventarisatie kan dan bepaald worden wat de volgende doelwitten zijn. De Voyager-worm heeft hier slechts primitieve mogelijkheden. Hij probeert om alle IP-adressen in het subnet van de machine, waarop de geïnfecteerde database draait te benaderen.

Specifieke aanvalsmogelijkheden

Hieronder wordt verstaan de methoden waarmee de worm zijn doel aanvalt. Het doel hierbij is initieel om binnen te komen en vervolgens om genoeg privileges te krijgen om zoveel mogelijk van zijn doelen te kunnen gebruiken. De Voyager-worm in zijn huidige vorm maakt gebruik van een aantal methoden. Voor het binnenkomen wordt gebruik gemaakt van enerzijds bekende zwakke wachtwoorden en anderzijds door het aanmaken de file glogin.sql, waarin een dergelijk account wordt aangemaakt. Deze file wordt tijdens het inloggen gedraaid door iedereen die van sqlplus gebruik maakt. De hoop van de schrijver van de worm is, dat dit ook gebeurt bij het inloggen van geprivilegieerde gebruikers die over voldoende privileges beschikken om een nieuw account aan te maken. Het hierbij aangemaakte account beschikt over het DBA-privilege. Zodra over voldoende privileges wordt beschikt wordt ook gebruik gemaakt van database-links.

> ***De Oracle Voyager-worm lijkt een eenvoudige worm, die nauwelijks een poging doet om zichzelf te verbergen***

Commando interface

Voor de meeste doeleinden, waarvoor iemand een worm zou willen maken, is het noodzakelijk om op een later tijdstip dan de verspreiding nieuwe commando's te kunnen geven. In de

Voyager-worm is een poging gedaan om een dergelijke interface te kunnen implementeren, door het creëren van een database-trigger AA waarvan de source via Google opgehaald zou worden. Google is hiervan op de hoogte gesteld en heeft het gebruikte request geblokkeerd.

Communicatie

In zijn algemeenheid is er naast het geven van commando's aan de worm, ook op andere vlakken behoefte aan communicatie. Te denken valt hierbij met name aan communicatie tussen geïnfecteerde systemen. De Voyager-worm lijkt op dit gebied over niets anders te beschikken dan de hiervoor genoemde database-trigger. Wel maakt de worm gebruik van e-mail om gegevens over de geïnfecteerde databases naar buiten te brengen. In de huidige vorm is dit alleen van alle password-hashes van een aantal geïnfecteerde databases naar larry@oracle.com.

Er is vooralsnog geen mogelijkheid ingebouwd om de database blijvend kwaad te doen of om gegevens te stelen

Gegevens verzamelen

Om gebruik te kunnen maken van het hele wormsysteem (alle door de worm geïnfecteerde systemen) is het nodig om de informatie welke systemen geïnfecteerd zijn op de één of andere manier aan een centrale plaats kenbaar te maken, of zodanig onderling te communiceren, dat (vrijwel) alle geïnfecteerde systemen snel kunnen worden bereikt.

Potentiële aanvalmogelijkheden

De meeste wormen hebben naast de gebruikte aanvalsmethoden nog een aantal potentiële aanvalsmethoden (rudimentair) geïmplementeerd. Dit laat de mogelijkheid open om de manier van verspreiding te wijzigen. Wormen worden veelal opgespoord door hun karakteristieke verspreidingswijze. Deze ongebruikte aanvalsmethoden maken het mogelijk om dit te wijzigen. Bij de Voyager-worm zou je het verzamelen van password-hashes als een onderdeel van een potentiële nieuwe aanvalsmethode kunnen beschouwen.
Een andere karakteristiek van wormen (evenals virussen) is hun payload. Hoewel de letterlijke vertaling van deze Engelse term 'betaalde vracht' is, wordt hier meestal onder verstaan, in hoeverre en hoe de vorm (of het virus) vervelende dingen doet in de geïnfecteerde systemen. Voor de Voyager-worm is dit:

- Het granten van de DBA-rol aan public.
- Het eventueel weggooien en opnieuw aanmaken van de after database logon trigger
- Het genereren van veel netwerkverkeer
  - Scannen van nieuwe potentiële doelwitten
  - Sturen van e-mail met password-hashes aan larry@oracle.com
  - Creëren van database-links
  - Een denial of service door het stoppen van de listener

Karakteristieken

Bij het analyseren van het gevaar van de Oracle Voyager worm, springen een aantal karakteristieken in het oog. Allereerst lijkt het een vooralsnog eenvoudige worm. Hij doet nauwelijks een poging om zichzelf te verbergen. Hij maakt gebruik van vulnerability's, die als eerste aangepakt worden bij het beveiligen van een Oracle-database. Er is vooralsnog geen mogelijkheid ingebouwd om de database blijvend kwaad te doen (wel onbereikbaar te maken via de listener) of om gegevens te stelen. Tevens is het verspreidingmechanisme dusdanig dat een snelle infectie van een groot aantal servers onmogelijk is. Omdat dit laatste uiteindelijk afhankelijk is van het aantal database-servers dat rechtstreeks luistert naar sqlnet requests aan het internet, is het zelfs de vraag of dit ooit tot een exponentiële verspreiding kan leiden. Het lamleggen van een deel van het internetverkeer, dat we normaal associëren met een uitbraak van een virus of worm wordt veroorzaakt door deze exponentiele verspreiding. Wat is dan wel het gevaar van een Oracle-worm? Dat is de mogelijkheid dat een dergelijk wapen wordt ingezet tegen specifieke sites, bijvoorbeeld voor bedrijfsspionage.

Gerard Uiterwaal is Oracle- en security expert en werkzaam bij Motiv. 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.