

WEBSERVER-BEVEILIGING IN NEDERLAND

ir. W.H.B. Belgers

Madison Gurkha, Postbus 2216, 5600 CE Eindhoven

walter@madison-gurkha.com

<http://www.madison-gurkha.com/>

14 april 2004

SAMENVATTING

Begin 2004 is door Madison Gurkha een onderzoek uitgevoerd naar de gebruikte webserver software op het Nederlandse deel van het internet. Gekeken is naar de versie-nummers van de software. Ondanks het feit dat deze gefingeerd kunnen worden is het aantal kwetsbare webserver zo hoog dat we, als we corrigeren, toch nog kunnen stellen dat het met de beveiliging van de Nederlandse webserver slecht gesteld is. Ongeveer een derde van alle servers is voorzien van de laatste versie, de overige servers draaien veelal (sterk) verouderde software die kwetsbaar is voor aanvallen waarbij een aanvaller willekeurige code op de server kan laten uitvoeren.

INLEIDING

Voortdurend worden nieuwe beveiligingslekken ontdekt in besturingssystemen en applicatiesoftware. Gelukkig worden die lekken vroeger of later ook gedicht. Dat wil zeggen: er wordt een nieuwe versie van de software aangeboden of een patch. In de meeste gevallen zal de beheerder de beschikbaarstelling van updates of patches in de gaten moeten houden en deze installeren. Beveiligen is dus iets wat je niet eenmalig doet; het is een doorlopend proces.

In februari en maart 2004 heeft Madison Gurkha een onderzoek uitgevoerd om te zien in hoeverre beheerders dit beveiligingsproces beheersen, door te kijken naar het beveiligingsniveau van webserver software op het internet.

METHODIEK

Op 11 februari 2004 waren meer dan 23 miljoen IP-nummers gereserveerd in Nederland. Steekproefsgewijs zijn hieruit 1.331.226 IP-nummers

genomen ter onderzoek. Het onderzoek beperkte zich tot het bepalen van het versienummer van de gebruikte webserver software, waarbij de webserver zich op de standaard poort (poort 80) moest bevinden. De reden om het onderzoek te beperken tot webserver is dat deze veel voorkomen en het versienummer op eenvoudige, legale wijze op te vragen is.

Met behulp van het tool *nmap*[3] is onderzocht welke systemen reageerden op verzoeken op poort 80. Naar alle systemen die reageerden is een HTTP 'OPTIONS' verzoek gestuurd. Er waren uiteindelijk 25.191 systemen die het verzoek beantwoordden.

De respons van een webserver op een verzoek (zoals het 'OPTIONS' verzoek) bevat normaliter een 'Server:' veld, zoals bijvoorbeeld 'Server: Apache/1.3.26 (Unix) Debian GNU/Linux PHP/4.1.2'. Dit veld bevat versie-informatie omtrent de gebruikte software. Deze waarden zijn in het onderzoek gebruikt. Daarbij dienen wel enige kanttekeningen te worden geplaatst.

Zo is het mogelijk de software zodanig

aan te passen dat deze een ander `Server:` veld terugstuurt. Zo meldde een van de onderzochte webservers zich als `'Server: Anders-dan-je-denkt'`. Ook kwam het voor dat een bestaande servernaam met een (nog) niet bestaand versienummer werd gebruikt zoals `'Apache/6.6.6'`.

Het wijzigen van de meldtekst gebeurt over het algemeen niet veel. Wat wel vaak voorkomt is het weglaten van het versienummer. In de populaire webserver Apache is dit door het toevoegen van een enkele regel aan het configuratiebestand eenvoudig mogelijk. De reden om het versienummer niet prijs te geven is om het inbrekers moeilijker te maken. Het idee is dat een inbreker eerst onderzoekt welke versie van de software draait, waarna een aanval kan worden opgezet voor die specifieke versie. In de praktijk gebeuren de meeste aanvallen tegenwoordig door automatische scripts, danwel uitgevoerd door de aanvaller zelf, danwel via een virus of worm. Bij zo'n aanval wordt gepoogd een specifieke *bug* te misbruiken, waarbij niet gekeken wordt naar de gebruikte versie. Het niet meegeven van het versienummer heeft dus tegenwoordig weinig toegevoegde waarde (maar als het met weinig moeite te implementeren is, is dit toch aan te raden).

Behalve dat beheerders meldteksten aanpassen komt het ook voor dat leveranciers gepatchte versies uitbrengen met hetzelfde versienummer als de ongepatchte software. Zo houdt Red Hat Linux zich bijvoorbeeld niet aan de gewoonte om patchlevels in de versiestring op te nemen.

Het is dus niet altijd mogelijk aan het versienummer te zien of het om kwetsbare (ongepatchte) of veilige (gepatchte) software gaat. Dit dient in het achterhoofd te worden gehouden bij het interpreteren van de resultaten uit dit onderzoek. Ondanks deze slag om de arm zijn de resultaten verontrustend.

WEBSERVER SOFTWARE

Als we de versienummers buiten beschouwing laten en alleen kijken naar het soort webserver dat draait, dan zien we dat Apache een marktaan-

Software	Servers	Percentage
Apache	18.858	74,9%
Microsoft-IIS	4.273	17,0%
Lotus-Domino	287	1,1%
Anders	1.773	7,0%
Totaal	25.191	100,0%

Tabel 1: Webserver software

deel heeft van 74,9% in onze steekproef, gevolgd door Microsoft-IIS met 17,0%. Lotus-Domino is de enige andere server met meer dan een procent marktaandeel (1,1%).

Deze cijfers komen ongeveer overeen met de Netcraft Web Server Survey[2] van april 2004. Daarin wordt gemeld dat Apache wereldwijd een marktaandeel van 67,20% heeft tegenover 21,02% voor Microsoft-IIS. Apache lijkt in Nederlands dus iets populairder te zijn.

Waar het in dit onderzoek om te doen is, is de veiligheid van de gebruikte versie van de software. Allereerst kijken we naar de gebruikte versies van Apache. Apache is er in versie 1.x en 2.x. Hoewel alleen aan versie 2.x actief ontwikkeld wordt, worden voor versie 1.x nog steeds patches uitgebracht. Ten tijde van het onderzoek waren versies 1.3.29 en 2.0.48 de nieuwste versies (inmiddels is dit niet meer het geval).

Van de 18.858 servers die zich melden als Apache servers zijn er verrassend veel die verder geen versienummer prijsgeven, namelijk 7.087 (37,6%). De oorzaak is vrijwel zeker het feit dat veel installaties dit tegenwoordig standaard in de configuratie opnemen.

Van de overgebleven 11.771 systemen zeggen er 1.531 (13,0%) versie 2.0.x en 10.044 (85,3%) versie 1.3.x te draaien. De percentages uitgesplitst naar versienummer zijn te vinden in tabel 2 en tabel 3.

Er wordt nog veel gebruikt gemaakt van de oudere versie 1.3.x. Een oorzaak hiervoor is onder ander het feit dat deze versie nog wordt meegeleverd als embedded webserver voor verschillende pakketten en besturingssystemen. Ook zullen sommige beheerders die al een versie 1.3.x

Versie	Servers	Percentage
2.0.48	968	8,2%
2.0.47	162	1,4%
2.0.41 ^t / _m 2.0.46	49	0,4%
2.0.40	328	2,8%
Anders	24	0,2%
Totaal	1.531	13,0%

Tabel 2: Apache versies (2.0.x)

Versie	Servers	Percentage
1.3.29	3.228	27,4%
1.3.27	2.315	19,7%
1.3.26	1.807	15,4%
1.3.20	1.038	8,8%
Anders	1.656	14,1%
Totaal	10.044	85,3%

Tabel 3: Apache versies (1.3.x)

van Apache hebben draaien liever niet upgraden naar versie 2.x vanwege het extra werk dat deze upgrade met zich meebrengt.

De sites met versie 2.0.x van Apache draaien meestal de nieuwste versie, het gebruik van versie 2.0.48 komt bijna drie maal zo veel voor als versie 2.0.40, na 2.0.48 de populairste 2.0.x versie. Bij versie 1.3.x zien we een ander beeld, daar is ook de nieuwste versie het populairst, maar zien we ook erg veel oude(re) versies terug. Er zijn 1.923 servers gevonden die melden dat ze een Apache versie uit de reeks 1.3.0 - 1.3.25 draaien (10,2% van alle Apache servers; 7,6% van alle webserver). Al deze versies bevatten de zogenaamde *'chunked encoding vulnerability'*. Dit medio 2002 ontdekte beveiligingsprobleem, dat in de categorie *'buffer overflows'* valt, wordt al lange tijd misbruikt door aanvallers. Programmatuur om dit probleem te misbruiken, op een scala van besturingssystemen, is al lange tijd beschikbaar. Tevens zijn er wormen in omloop die dit lek misbruiken om in te breken. Het is bij deze aanval mogelijk om op het systeem zelf willekeurige code uit te voeren met de rechten waar-

onder de webserver-software draait. Afhankelijk van de configuratie en extra veiligheidsmaatregelen kan dat resulteren in volledige controle over de machine, de mogelijkheid om de inhoud van de site aan te passen, of een Denial of Service aanval waarbij de webserver 'uit de lucht' gaat.

Versie	Servers	Percentage
5.0	3.339	78,1%
4.0	771	18,0%
5.1	147	3,4%
6.0	10	0,2%
3.0	5	0,1%
Anders	1	0,0%

Tabel 4: Microsoft IIS versies

Bij Microsoft's IIS zijn minder verschillende versies in gebruik. Van de 4.273 systemen draaiden er 3.339 (78,1%) IIS 5.0, 147 draaiden 5.1 (3,4%) en 771 (18,0%) hadden nog IIS 4.0 draaien.

Bij IIS 6.0 heeft Microsoft ervoor gezorgd dat veel onveilige protocollen standaard uit staan. Bij oudere versies staan deze standaard aan. Het gaat daarbij onder andere om ISAPI extensies waar in het verleden al een aantal problemen in is gevonden. IIS 6.0 wordt nog maar weinig gebruikt, omdat Windows 2003 Server nog erg nieuw was ten tijde van het onderzoek.

In oudere (4.0, 5.0, 5.1) versies van IIS zijn, net als in Apache, de nodige beveiligingsproblemen gevonden. Hiervoor zijn fixes uitgebracht in de vorm van security hotfixes, maar ook als Service Packs (het gaat hier om een bundel van hotfixes voor zowel het besturingssysteem als IIS). Deze hotfixes veranderen niets aan het versienummer dat wordt gemeld door de server en de aanwezigheid ervan kon in dit onderzoek dan ook niet worden vastgesteld. Uit praktijk blijkt dat het slechts een minderheid is die alle patches tijdig aanbrengt.

OVERIGE SOFTWARE

Behalve het versienummer van de gebruikte webserver software wordt vaak nog extra informatie meegegeven. Een voorbeeld van een server identificatie die werd teruggegeven is:

```
Apache/1.3.20 Sun Cobalt (Unix)
Chili!Soft-ASP/3.6.2 mod_ssl/2.8.4
OpenSSL/0.9.6b PHP/4.1.2
mod_auth_pam_external/0.1
FrontPage/4.0.4.3 mod_perl/1.25
```

We zien hier dat de server zich meldt als Apache/1.3.20. Daarnaast blijkt dit een Apache versie op een Sun Cobalt systeem te zijn met ASP extensies en enkele tijdens de compilatie ingebouwde modules zoals FrontPage, PHP en Perl. Ook hiervoor worden versienummers meegegeven. Niet elke webserver geeft zo'n uitgebreid overzicht. We zullen verder onderzoek plegen naar het gebruik van PHP en OpenSSL.

PHP

PHP[5] is een scripting taal die, hoewel deze niet beperkt is tot het World Wide Web, veel voor web-applicaties wordt gebruikt. Hoewel de meeste beveiligingsproblemen in PHP programmatuur optreden door onveilig programmeren, beperken we ons hier tot beveiligingsproblemen in de PHP interpreter zelf. Dit soort lekken leiden er in de regel toe dat een aanvallende willekeurige code kan uitvoeren op de server. Dit gebeurt dan met de rechten waaronder PHP programma's worden uitgevoerd (meestal de rechten waarmee de webserver software draait).

Hoewel PHP ook beschikbaar is voor gebruikers van IIS, is dit niet in het onderzoek meegenomen omdat IIS het PHP versienummer niet geeft. Dit komt door het feit dat PHP geen onderdeel van IIS maar een add-on is, terwijl PHP bij het gebruik van Apache normaliter wordt meegecompileerd.

PHP versie 4.3.4 (en 4.3.4-RC, een 'release candidate') was ten tijde van het onderzoek het

Versie	Servers	Percentage
4.3.4(RC)	3.284	37,0%
4.1.2	1.954	22,0%
4.3.3(RC)	787	8,9%
4.3.2(RC)	780	8,8%
4.0.x(RC)	636	7,2%
Anders	1.427	16,1%

Tabel 5: PHP versies

meest populair. Dit was de nieuwste versie (inmiddels is versie 4.3.5 al bijna beschikbaar). Opvallend is dat van de 8.868 systemen die een PHP versienummer prijsgeven, er 827 (9,3%) versie 4.1.1 of ouder draaiden. Meer dan twee jaar geleden is in een security advisory[6] een groot aantal problemen bekend gemaakt in deze versies, die kunnen leiden tot het uitvoeren van code door aanvallers. Dat betekent dat 9,3% van de servers wordt beheerd door iemand die structureel geen softwareupdates uitvoert, ook niet als er kritieke problemen bekend zijn.

De hoeveelheid servers die de laatste versie van de software gebruiken (37,0%) komt overeen met de hoeveelheid servers die de laatste versie van Apache gebruiken (35,6%).

OPENSSL

Een ander versienummer dat Apache vaak meegeeft is die van de ingebouwde OpenSSL[4] software. OpenSSL wordt gebruikt voor het opzetten van beveiligde verbindingen tussen een browser en een webserver. Het protocol dat hiervoor wordt gebruikt is SSL, OpenSSL is daar een implementatie van.

Ten tijde van het onderzoek was OpenSSL 0.9.7c de laatste versie. Ook versie 0.9.6l was een versie zonder bekende problemen (inmiddels zijn er nieuwere versies beschikbaar).

Van de 5.233 servers die een OpenSSL versienummer prijsgeven draait 29,2% de nieuwste versie. Dit komt weer ongeveer overeen met de getallen die we zagen voor Apache en PHP.

Versie	Servers	Percentage
0.9.7c	1.526	29,2%
0.9.6b	1.271	24,3%
0.9.6	681	13,0%
0.9.6x	707	13,5%
0.9.7a/0.9.7b	425	8,1%
0.9.5a	290	5,5%
≤0.9.5	241	4,6%
0.9.7x en 0.9.8-dev	92	1,8%

Tabel 6: OpenSSL versies

De oudere versies hebben allen problemen. Soms zijn dat moeilijk te exploiteren problemen. Vaak is het maximaal haalbare resultaat een Denial of Service (het laten crashen van de webserver). Echter, in OpenSSL versies ouder dan 0.9.6e en OpenSSL 0.9.7-beta2 zitten een aantal problemen waardoor een aanvaller willekeurige code op de server kan uitvoeren. Dat betekent dat 2.512 servers een OpenSSL versie draaien die kwetsbaar is (van de 707 0.9.6x servers uit tabel 6 draaien er 29 0.9.6a). Dat is maar liefst 48,0% van alle servers die een OpenSSL versienummer gaven.

Zelfs na correctie van deze getallen vanwege de onzekerheid dat een gemelde softwareversie ook de werkelijk draaiende versie is, is dit een verontrustend hoog aantal: als we de cijfers extrapoleren naar de 23 miljoen Nederlandse IP-nummers, dan zouden ruim 40.000 webserver dit specifieke beveiligingslek (en wellicht nog meer lekken) hebben. Overigens tellen we dan dus ook de over het algemeen slecht beveiligde kabelmodemgebruikers mee.

CONCLUSIES

Concluderend kunnen we stellen dat ongeveer een derde van alle webserver verouderde software gebruikt. Hoewel dit niet in alle gevallen tot kwetsbaarheden leidt, toont het wel aan dat een doorlopend proces van patchen of vernieuwen van software ontbreekt. Dat geldt voor

servers die vanaf het internet benaderbaar zijn. Veelal zijn interne systemen afgeschermd van het internet door middel van een firewall. In de regel zijn interne systemen minder goed beveiligd dan systemen die via het internet bereikbaar zijn, en zullen de problemen intern dus nog groter zijn.

Mogelijk zijn er beheerders die op geregelde tijden controleren of er updates nodig zijn, resulterend in het in geringe mate achterlopen bij de nieuwste releases. Deze manier van werken vermindert het risico op een inbraak, maar niet voldoende. Zodra nieuwe beveiligingslekken bekend zijn duurt het niet lang voordat geautomatiseerde scripts en wormen gebruik maken van deze lekken om in te breken op willekeurige systemen. Zodra een security advisory wordt uitgegeven, door de leverancier of via een publieke mailinglijst zoals Bugtraq[1] dient de software zo snel mogelijk te worden gepatched of vernieuwd. Beveiliging is dus een doorlopend proces.

Daarnaast moet gezegd worden dat het installeren van de nieuwste versies van de gebruikte software nog niet noodzakelijk tot gevolg heeft dat alle beveiligingsproblemen zijn opgelost. Veel problemen ontstaan bij het configureren van software en in zelf ontwikkelde software of koppelingen (bijvoorbeeld tussen een web server en een database server).

Om tot een gedegen oordeel te komen over het beveiligingsniveau van bepaalde servers is dus een goede technische security audit nodig. De resultaten van dit eenvoudige onderzoek, dat alleen naar de standaard webserver software kijkt, toont echter al aan dat het met de beveiliging van webserver op het Nederlandse deel van het internet slecht gesteld is.

REFERENTIES

- [1] Bugtraq en NTBugtraq,
<http://www.securityfocus.com/archive/1>
en <http://www.ntbugtraq.com/>
- [2] Netcraft Web Server Survey,
[http://news.netcraft.com/archives/
web_server_survey.html](http://news.netcraft.com/archives/web_server_survey.html)
- [3] nmap, <http://www.insecure.org/nmap/>
- [4] OpenSSL, <http://www.openssl.org/>
- [5] PHP, <http://www.php.net/>
- [6] E-matters PHP advisory, [http://security.
e-matters.de/advisories/012002.txt](http://security.e-matters.de/advisories/012002.txt)