Voorwoord/colofon

Beste lezer,

Tijd voor vakantie, dus waar haal je dan de inspiratie vandaan om een voorwoord te schrijven? Geen idee, het is nu even mooi weer en wat doe je dan? Even mijn browser starten en mijn startpagina verschijnt, een rss-aggregator, waarmee ik mijn RSS-jes kan volgen. En dat levert dan meteen inspiratie op. Want waar haal ik normaal alle kennis vandaan? Van internet, blogjes en zo. Er zijn zoveel sites die interessant en relevant zijn, die kun je niet allemaal actief volgen. Vandaar dat ik gebruik maak van een rss-reader. Ik gebruik zelf netvibes.com. Igoogle is ook leuk, maar ja, Google weet al zo veel van mij, ik wil niet alle eieren in één mandje bewaren. In ieder geval is zo'n aggregator erg handig. Zo kan ik van tientallen sites het nieuws volgen. En kan ik op basis van een enkele regel tekst besluiten om de site te bezoeken. Dat gaat veel sneller dan al die sites altijd te bezoeken. Ja, ja, zo hou ik alles in de greep...

Een paar van de sites die ik via rss'jes volg:

- http://www.itsprivacy.nl/, een Nederlandse site over privacy. Geen idee wie erachter zit, maar zo af en toe volg ik een berichtje.
- Natuurlijk ook http://www.security.nl/. Heel veel info, meningen en peilingen. Een aantal vakbroeders neemt actief deel. En de site levert natuurlijk altijd screendumps voor weer een powerpoint over informatiebeveiliging.
- http://cloudsecurity.org/ heb ik in een eerder nummer al opgevoerd, maar verdient weer een nominatie.
- We hebben ook een digitale concurrent: http://www.net-security.org publiceert ook een gratis vakblad. Leuk om eens te vergelijken!
- Je bent als professional natuurlijk geen knip voor je neus waard als je Bruce Schneier op zijn blog http://www.schneier.com/blog/ niet volgt. Nuttig ook als je inktvis op vrijdag leuk vindt.

Mijn 'identity' hobby kan ik botvieren op de volgende sites:

- Paul Madsen: http://connectid.blogspot.com/
- Ash (Ashraf Motiwala) http://identityman.blogspot.com/
- Kim Cameron natuurlijk: http://www.identityblog.com/
- Mike Jones: http://self-issued.info/
- Eve Maler: http://www.xmlgrrl.com/blog
- Jeff Bohren: http://idlogger.wordpress.com/
- Ian Yip: http://blog.ianyip.com/
- En goede kennis Marcus vind je hier: http://www.bloglines.com/blog/Marcus-Lasance

Er zijn nog veel meer interessante sites, maar ja, ik heb niet zoveel ruimte. En ik wil toch nog even mijn blogje melden: http://id-use.blogspot.com/, waar ik wat nadenk over identity 2.0 dingetjes.

Behoefte aan nog meer information overflow: start Twitter en volg gerust @securitystuff, @prabath, @websecuritynews, @ryanaraine, @gcluley (jawel van sophos), @cloudbook, @metadaddy, @remcobakker, @robstwitts, @meneer (dat ben ik).

Veel leesplezier,

André Koot
Hoofdredacteur

Informatiebeveiliging is het huisorgaan van het Platform voor InformatieBeveiliging (PvIB) en bevat ontwikkelingen en achtergronden over onderwerpen op het gebied van informatiebeveiliging.

Redactie
André Koot (hoofdredactie, werkzaam bij Univé-VGZ-IZA-Trias),
e-mail: A.Koot@Unive.nl
Peter Westerling/Monique van Diessen (eindredactie, TOPpers Media bv, Berlicum)

Redactieraad
Tom Bakker (Delta Lloyd)
Mario de Boer (Logica)
Lex Borger (Domus technica)
Lex Dunn (Capgemini)
Rob Greuter (Secode Nederland)
Aart Jochem (GOVCERT.NL)
Renato Kuiper (HP)
Henk Meeuwisse (Sogeti)
Gerrit Post (G & I Beheer BV)

Advertentieacquisitie
e-mail: adverteren@pvib.nl

Vormgeving
Van Velzen Grafisch Ontwerp, 's-Hertogenbosch

Uitgever
Platform voor InformatieBeveiliging (PvIB)
Postbus 1058
3860 BB NIJKERK
T (033) 247 34 92
F (033) 246 04 70
E-mail: secretariaat@pvib.nl
Website: www.pvib.nl

Abonnementen
De abonnementsprijs bedraagt 115 euro per jaar (exclusief BTW), prijswijzigingen voorbehouden.

PvIB abonnementenadministratie
Platform voor InformatieBeveiliging (PvIB)
Postbus 1058
3860 BB NIJKERK
e-mail: secretariaat@pvib.nl

ISSN 1569-1063

creative commons

PvIB Platform voor InformatieBeveiliging

AutoNessus: herhaaldelijk scannen realiseert gemak

Auteur: Ing. Frank Breedijk CISSP > Frank Breedijk werkt al ruim drie jaar als Security Engineer voor Schuberg Philis en is bereikbaar via fbreedijk@schubergphilis.com.

Tijdens mijn werk als Security Engineer bij Schuberg Philis maak ik regelmatig gebruik van vulnerability scanners zoals Nessus en OpenVAS. Deze scanners zijn krachtig gereedschap, maar bij lange na niet perfect. Iedere scan levert een rapport op met een groot aantal bevindingen. Iedere bevinding moet (separaat) worden onderzocht om op de juiste waarde te kunnen schatten. Logischerwijs vergt dit een aanzienlijke tijdsinspanning. Vaak worden deze scanners slechts op ad hoc basis ingezet, bijvoorbeeld bij het opleveren van een nieuwe infrastructuur of bij grote veranderingen. De dynamiek van de IT, waar iedere infrastructuur die we beheren regelmatig verandert, maakt het wenselijk scans herhaaldelijk uit te voeren. Deze uitdaging heeft me ertoe aangezet het programma AutoNessus te schrijven, een programma dat het uitvoeren en het verwerken van de resultaten van deze scans vereenvoudigd.

Wat is een vulnerability scanner?

In de voorgaande paragraaf heb ik een aantal keren het woord vulnerability scanner gebruikt, maar... wat is een vulnerability scanner nu eigenlijk?

Een vulnerability scanner is een programma dat als doel heeft kwetsbaarheden in software of infrastructuren geautomatiseerd op te sporen. Nessus en OpenVAS zijn bekende netwerk vulnerability scanners. Deze programma's zijn er dus speciaal op gericht om via het netwerk kwetsbaarheden op te sporen, maar er zijn nog veel meer vulnerability scanners, ieder met zijn eigen doel en insteek (zie kader).

Nessus en OpenVAS werken aan de hand van de volgende procedure:

Stap 1: Stel vast of een IP adres actief is. Hiervoor worden technieken als een ping en een simpele portscan gebruikt;

Stap 2: Probeer vast te stellen welk diensten (services) een IP adres aanbiedt aan het netwerk en welk besturingssysteem aanwezig is;

Stap 3: Bepaal of bekende kwetsbaarheden (mogelijk) aanwezig zijn in de aangeboden services. Dit kan door simpel weg het versienummer van de applicatie te vergelijken met een lijst van bekende kwetsbaarheden, maar ook door eventueel stap 4;

Stap 4: Misbruik een kwetsbaarheid op het systeem om zijn aanwezigheid aan te tonen;

Stap 5: Rapporteer de bevindingen.

Helaas is het zo dat geautomatiseerd vulnerability scanners enkel die kwetsbaarheden vinden waarvoor reeds een test bestaat. Een goede penetration tester zal, geholpen door zijn of haar menselijke creativiteit, meer kwetsbaarheden vinden dan welk geautomatiseerd gereedschap dan ook. Een geautomatiseerde scanner kan en zal dus nooit een compleet beeld geven. Daarnaast is geen enkele test zonder risico. Geen enkele penentration tester of scanner fabrikant zal harde garanties geven dat een test geen enkele invloed zal hebben op de beschikbaarheid van de te testen applicatie of infrastructuur.

Waarom scannen?

Het potentiële risico van een vulnerability scan wordt vaak als argument gebruikt om helemaal niet te scannen. Of er wordt bijvoorbeeld besloten vulnerability scans alleen uit te laten voeren door penetration testers. Hoewel ik de gedachte hierachter begrijp, wil ik iedereen toch aansporen om ook zelf vulnerability scans uit te voeren. En wel om twee redenen: vulnerability scanners zijn vrijelijk beschikbaar voor zowel 'de goeden' als 'de slechten' en het daadwerkelijke risico valt wel mee.

Veel vulnerability scanners zijn open source en/of gratis beschikbaar. Nessus, OpenVAS, NMAP, Nikto en vele anderen zijn voor iedereen te downloaden en dus door iedereen te gebruiken. De informatie die een dergelijke tool geeft over de aanwezige kwetsbaarheden kun je dus feitelijk als publiek beschikbare informatie beschouwen. Het is weliswaar illegaal om zonder toestemming een scan uit te voeren, maar iedereen die wel eens een firewall log gezien heeft, weet dat dit niet betekent dat het niet gebeurt. En als iedereen, goed of slecht, deze informatie kan genereren, kun je het maar beter zelf weten ook.

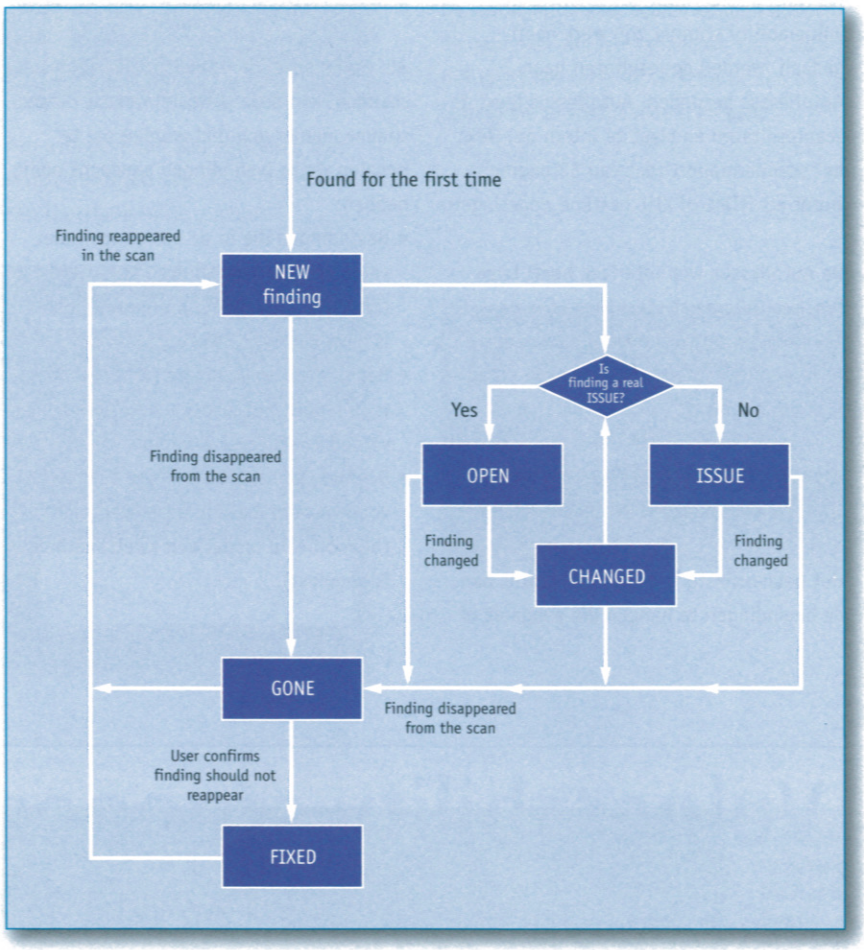
Bij het uitvoeren van een vulnerability scan ontstaat een spanningsveld. Immers, om de drie aspecten van informatiebeveiliging (beschikbaarheid, integriteit en vertrouwelijkheid) te garanderen, moet een actie worden uitgevoerd, waarbij één van deze aspecten, beschikbaarheid, niet honderd procent kan worden gegarandeerd. Dit

risico kan niet worden uitgesloten, maar het is wel te relativeren. Allereerst is de omgeving waarschijnlijk al zonder ons medeweten gescand. Alles wat met het internet verbonden is, wordt immers wel eens gescand. Ten tweede valt het daadwerkelijke risico wel mee. Ik heb in drie jaar redelijk intensief scannen slechts twee maal een infrastructuur tijdelijk verstoort. In beide gevallen was er geen permanente schade en kon de scan met een lagere intensiteit later gewoon worden voortgezet.

Vulnerability scans zijn niet de enige werkzaamheden die verstoringen kunnen veroorzaken, changes zorgen vaak ook voor geplande of ongeplande verstoringen. Door een scan net als een, regelmatig terugkerende, change in te plannen, kan vaak een voor iedereen werkbare situatie geschapen worden.

Dubbel scanner, dubbel werk?

Het controleren van een netwerk op kwetsbaarheden zou geen eenmalige activiteit moeten zijn. Vrijwel iedere IT infrastructuur is aan verandering onderhevig. Zelfs een volledig statische infrastructuur beweegt ten opzicht van zijn omgeving. Iedere dag worden nieuwe kwetsbaarheden gevonden. Een systeem waarin we vandaag geen kwetsbaarheden konden vaststellen, kan met de informatie die we morgen hebben plotseling kwetsbaar blijken te zijn.



Het uitvoeren van een vulnerability scan is relatief eenvoudig. Helaas is het analyseren van de resultaten arbeidsintensief. Om een voorbeeld te geven: ik scan iedere maand een omgeving met 130 IP-adressen. Geen van de IP-adressen biedt normaal gesproken voor de scanner benaderbare diensten aan. Als ik deze infrastructuur met Nessus scan, dan levert dit een rapport van 52 pagina's op met meer dan vierhonderd bevindingen. Het kost mij zeker twee uur om dit volledig door te lezen en zeker vier uur als er ook een formele rapportage nodig is.
Ik vind het erg inefficiënt om elke maand wederom twee uur te investeren om een relatief statische omgeving te scannen, daarom ben ik AutoNessus gaan schrijven.

Wat doet AutoNessus?

AutoNessus is een tool die het mogelijk maakt op gezette tijden automatisch Nessus en OpenVAS scans uit te voeren. Daarnaast reduceert AutoNessus de tijd die nodig is voor het analyseren van twee opeenvolgende scans van dezelfde infrastructuur, door de resultaten van de huidige scan met de eerder beoordeelde resultaten van de voorgaande scan te vergelijken.

Over de auteur

Ing. Frank Breedijk CISSP werkt al ruim drie jaar als Security Engineer voor Schuberg Philis, de leverancier van bedrijfskritische outsourcings diensten met een honderd procent functionele beschikbaarheid. Zijn taken omvatten onder andere het geven security awareness trainingen, vulnerability management, interne security consultancy en het uitvoeren van technische audits. Voor Schuberg Philis werkte hij als security consultant by INS (nu BT), als ICT Security Officer bij InterXion en was hij drie en een half jaar langer manager van het EMEA Security Operation Center voor Unisys Managed Security Services. Naast zijn werk is Frank de auteur van AutoNessus. Frank is bereikbaar via het email adres fbreedijk@schubergphilis.com.

Stel dat een infrastructuur voor het eerst via AutoNessus wordt gescand. Als de vulnerability scanner zijn werk heeft gedaan, worden de resultaten naar AutoNessus gezonden. AutoNessus leest de scanresultaten en slaat ze intern op. Ook het scannerrapport (dat van 52 pagina's) wordt als HTML of XML bestand opgeslagen.

De AutoNessus web interface heeft twee functies. Allereerst staat hij je toe de bevindingen op verschillende manieren te filteren, waardoor er minder tijd nodig is voor de analyse.

Het belangrijkste is echter een status aan de bevindingen te hangen die aangeeft of een bevinding al dan niet een daadwerkelijke kwetsbaarheid is.

Als we nu dezelfde infrastructuur nogmaals scannen, kan deze statusinformatie op een slimme manier gebruikt worden om te bepalen welke bevindingen aandacht nodig hebben:

- Bevindingen die in de vorige scan niet voorkwamen, maar in deze scan wel (bijvoorbeeld omdat een nieuwe host is gevonden);
- Bevindingen waarvan de tekst veranderd is (bijvoorbeeld omdat het versienummer van een webserver is veranderd);
- Bevindingen die in de vorige scan wel voorkwamen, maar in deze scan niet (bijvoorbeeld omdat een kwetsbaarheid is opgelost).

Doordat de hoeveelheid te analyseren informatie afneemt, is er minder tijd nodig voor de analyse. Daarnaast gaat de accuratesse omhoog omdat de repetitiegraad van het werk omlaag gaat. Dat dit het resultaat ten goede komt, moge duidelijk zijn.

Praktijkvoorbeeld: Schuberg Philis

Mijn werkgever Schuberg Philis biedt outsourcing van bedrijfskritische applicaties met een honderd procent functionele beschikbaarheid. We gebruiken AutoNessus nu bijna twee jaar en scannen alle externe IP-adressen van al onze klanten (4000+ adressen) iedere maand. We scannen onder andere een flink aantal online banken, een grote webshop en de publieke website van een grote financiële instelling. Op dit moment staat een kleine

Vulnerability scanners

- Nessus – Network vulnerability scanner - www.nessus.org
- NMap – Network Mapper – www.insecure.org
  De network portscanner
- OpenVAS – Open Source network vulnerability scanner
  www.openvas.org
  Gebaseerd op de laatste opensource Nessus versie
- Nikto – Web server vulnerability scanner
  http://www.cirt.net/code/nikto.shtml
- Arirang – Open Source web server security scanner
  http://monkey.org/~pilot/arirang/
- Acunetix – Web application security scanner
  http://www.acunetix.com/
- GFI Languard – Network vulnerability scanner
  http://www.gfi.com/languard/
- Retine – Network vulnerability scanner
  http://www.eeye.com/html/Products/Retina/index.html
- SAINT – Network security scanner
  http://www.saintcorporation.com/index.html
- Qualys – Software as a Service network vulnerability scanner
  http://www.qualys.com/
- N-Stalker – Web application vulnerability scanner
  http://www.nstalker.com/
- Core Impact – Penetration testing tool
  http://www.coresecurity.com/
- IIS Internet Scanner – Network based application level scanner - http://www.iss.net/
- Microsoft Baseline Security Scanner – Microsoft Security Tool
  http://www.microsoft.com/technet/security/tools/mbsahome.mspx
- Hailstorm – Network vulnerability scanner
  http://www.cenzic.com/products_services/cenzic_hailstorm.php
- WebInspect – Web application vulnerability scanner
  http://www.spidynamics.com/products/webinspect/index.html
- NTO Spider – Web application vulnerability scanner
  http://www.ntobjectives.com/products/ntospider.php
- Grabber – Web application vulnerability scanner
  http://rgaucher.info/beta/grabber/
- Wapiti – Web application vulnerability scanner
  http://wapiti.sourceforge.net/

9000 bevindingen in het systeem. Voor de analyse van de resultaten is slechts ongeveer anderhalf à twee mandagen nodig. Ter illustratie: voor het analyseren van een eerste scan van 255 adressen is minimaal een halve dag nodig.

Conclusie

Wat mij betreft zou iedereen die een IT infrastructuur beheert deze regelmatig met een vulnerability scanner moeten controleren. Indien je verstandig met het geringe risico op downtime omgaat, is er altijd wel een moment te vinden waarop gescand kan worden. AutoNessus is een door mij geschreven open source tool, die de hoeveelheid werk gepaard gaand met het herhaaldelijk scannen van dezelfde infrastructuur kan reduceren en de nauwkeurigheid van de analyse kan verhogen.

Een overzicht van alle bevinding met status CHANGED van plugin 14260 voor een specifieke infrastructuur.

Een overzicht van een enkele bevinding. De status en remark velden kunnen door de gebruiker worden aangepast. Het veld diff geeft het textuele verschil tussen de laatste en de voorgaande scan aan.

Links

- www.autonessus.com
- www.schubergphilis.com
- http://twitter.com/autonessus
- http://www.linkedin.com/in/schanulleke
- www.cupfighter.net