Linux Security HOWTO

Kevin Fenzi, kevin@tummy.com & Dave Wreski, dave@linuxsecurity.com
Vertaald door Nine Matthijssen, smurfin@nl.linux.org v1.1.1, 17 maart 2000

Dit document is een algemeen overzicht van beveiligingskwesties waar een beheerder van Linux systemen mee geconfronteerd wordt. Het behandelt een algemene beveiligingsfilosofie en een aantal specifieke voorbeelden van hoe je je Linux-systeem beter tegen indringers kunt beveiligen. Ook zijn er verwijzingen naar materiaal en programma's die betrekking hebben op beveiliging. Verbeteringen, constructieve kritiek, toevoegingen en correcties worden dankbaar geaccepteerd. Stuur je reactie alsjeblieft naar beide auteurs, met "Security HOWTO" als onderwerp.

Inhoudsopgave

1 Inleiding

Dit document behandelt enkele van de belangrijkste kwesties die betrekking hebben op beveiliging in Linux. Algemene filosofie en via het netwerk ontstane middelen worden besproken.

In een aantal andere HOWTO documenten worden ook beveiligingskwesties behandeld en naar deze documenten wordt verwezen als dat nodig is.

Dit document kan *niet* zo bijgewerkt zijn dat alle nieuwe beveiligingslekken erin genoemd worden, aangezien er voortdurend nieuwe beveiligingslekken worden ontdekt. Dit document zal je vertellen waar je moet zoeken naar dergelijke up-to-date informatie en zal je enkele algemene methoden geven om te voorkomen dat zulke beveiligingslekken plaats hebben.

1.1 Nieuwe versies van dit document

Nieuwe versies van dit document zullen periodiek worden gestuurd naar *comp.os.linux.answers*. Ze zullen ook worden toegevoegd aan de diverse sites die zulke informatie archiveren, waaronder:

http://www.linuxdoc.org/

Bovendien zul je normaal gesproken dit document ook moeten kunnen vinden op de Linux World Wide Web home page via:

http://metalab.unc.edu/mdw/linux.html

Tot slot zou de meest recente versie van dit document ook in verschillende formaten beschikbaar moeten zijn op:

http://scrye.com/˜kevin/lsh/

of

http://www.linuxsecurity.com/Security-HOWTO

of

http://www.tummy.com/security-howto

1.2 Reacties

Alle opmerkingen, foutmeldingen, aanvullende informatie en alle mogelijke vormen van kritiek kunnen worden gestuurd naar:

kevin@tummy.com

en

dave@linuxsecurity.com

Let op: Stuur je reactie alsjeblieft naar *beide* auteurs. Ook moet je voor de zekerheid "Linux", "securityöf "HOWTOïn het onderwerp zetten om Kevin's spamfilter te ontwijken.

1.3 Disclaimer

Voor de inhoud van dit document kan geen aansprakelijkheid worden geaccepteerd. Gebruik de begrippen, voorbeelden en andere inhoud op eigen risico. Bovendien is dit een concept, mogelijk met veel onnauwkeurigheden of fouten.

Voor een aantal voorbeelden en beschrijvingen wordt gebruik gemaakt van de RedHat(tm) pakket-layout en systeemsetup. De weg die jij moet bewandelen om zover te komen kan anders zijn.

Voor zover we weten worden er alleen programma's beschreven die onder bepaalde voorwaarden mogen worden gebruikt of geëvalueerd voor persoonlijke doeleinden. De meeste programma's zijn beschikbaar, compleet met broncode, onder *GNU* `<http://www.gnu.org/copyleft/gpl.html>` voorwaarden.

1.4 Copyright informatie

2 Overzicht

Dit document zal enkele procedures en veelgebruikte software om je te helpen je Linux systeem veiliger te maken proberen uit te leggen. Het is belangrijk om, voordat we beginnen, eerst enkele basisbegrippen te bespreken en een basisbeveiliging te creëren.

2.1 Wat is het nut van beveiliging?

In de altijd veranderende wereld van globale datacommunicatie, goedkope Internetverbindingen en het hoge tempo van software ontwikkeling, wordt beveiliging een steeds belangrijker onderwerp. Beveiliging is nu een basisvereiste, omdat globale informatica inherent onveilig is. Als je gegevens bijvoorbeeld van punt A naar B op het Internet gaan, zou het onderweg via diverse andere punten kunnen gaan, hetgeen andere gebruikers de gelegenheid geeft het te onderscheppen en het zelfs te wijzigen. Zelfs andere gebruikers op je systeem kunnen opzettelijk jouw gegevens veranderen in iets dat je niet bedoelde. Onbevoegde toegang tot je systeem kan verkregen worden door indringers, ook bekend als "crackers", die dan geavanceerde kennis gebruiken om zich als jou voor te doen, informatie van je te stelen of je zelfs de toegang tot je eigen middelen te ontzeggen. Als je je afvraagt wat het verschil is tussen een "Hackerën een "Cracker", bekijk dan Eric Raymond's document "How to Become A Hacker", beschikbaar op *http://www.netaxs.com/˜esr/faqs/hacker-howto.html.*

2.2 Hoe veilig is veilig?

Houd allereerst in gedachten dat geen enkel computersysteem ooit volledig veilig kan zijn. Je kunt het alleen maar steeds moeilijker voor iemand maken om je systeem in gevaar te brengen. Voor de gemiddelde Linux thuisgebruiker is er niet veel nodig om de terloopse cracker buiten de deur te houden. Voor professionele Linux gebruikers (banken, telecommunicatie-bedrijven enz.) is echter veel meer werk vereist.

Een andere factor om rekening mee te houden is dat hoe veiliger je systeem is, hoe indringeriger je beveiliging wordt. Je moet een balans zien te vinden zodat je systeem nog steeds bruikbaar is en toch veilig voor jouw doeleinden. Je kunt bijvoorbeeld verlangen dat iedereen die op je systeem inbelt een 'call-back modem' gebruikt om ze terug te kunnen bellen op hun nummer thuis. Dit is veiliger, maar als iemand niet thuis is, wordt het moeilijk voor hen om in te loggen. Je kunt je Linux systeem ook instellen zonder netwerk of verbinding met het Internet, maar dit beperkt zijn bruikbaarheid.

Als het een gemiddeld tot grote site betreft, zul je een beveiligingsbeleid moeten vaststellen, waarin staat hoeveel beveiliging voor jouw site vereist is en op welke wijze dit gecontroleerd wordt. Je kunt een voorbeeld van een welbekend beveiligingsbeleid vinden op *http://www.faqs.org/rfcs/rfc2196.html*. Het is recent bijgewerkt en bevat een goede opzet om een beveiligingsbeleid voor jouw bedrijf vast te kunnen stellen.

2.3 Wat probeer je te beschermen?

Voordat je probeert je systeem te beveiligen, moet je vaststellen tegen welk niveau van bedreiging je je moet beschermen, welke risico's je wel of niet moet nemen en hoe kwetsbaar je systeem als gevolg hiervan is. Je moet je systeem analiseren om te weten wat je beschermt, waarom je het beschermt, welke waarde het heeft en wie de verantwoording voor je data en andere bezittingen heeft.

- Een *risico* is de mogelijkheid dat een indringer succesvol kan zijn in zijn pogingen om toegang tot je computer te krijgen. Kan een indringer bestanden lezen of schrijven of programma's uitvoeren die schade kunnen veroorzaken? Kan hij kritieke data verwijderen? Kan hij voorkomen dat jij of jouw bedrijf belangrijk werk gedaan krijgt? Vergeet niet: iemand die zich toegang verschaft tot jouw account of jouw systeem kan zich ook als jou voordoen. Bovendien kan het hebben van één onveilig account erin resulteren dat je hele netwerk in gevaar komt. Als je een enkele gebruiker toestaat om in te loggen middels een `.rhosts` file of door gebruik te maken van een onveilige service als `tftp`, riskeer je dat een indringer 'zijn voet tussen de deur krijgt'. Als de indringer eenmaal een gebruikersaccount op jouw of iemand anders z'n systeem heeft, kan het gebruikt worden om toegang tot een ander systeem of account te verkrijgen.
- Een *bedreiging* vormt iemand die gemotiveerd is om onbevoegd toegang tot je netwerk of computer te krijgen. Je moet vaststellen wie je vertrouwt om toegang tot je systeem te hebben en welke bedreiging ze kunnen vormen.
- Er zijn verschillende typen indringers en het is handig om hun verschillende karakteristieken in gedachten te houden als je je systemen gaat beveiligen.
  - **De Nieuwsgierige** - Dit type indringer is voornamelijk geïnteresseerd in het uitvinden wat voor type systeem en gegevens je hebt.
  - **De Kwaadwillige** - Dit type indringer is erop uit om je systemen ten val te brengen, je webpagina te ontsieren of je op een andere manier te dwingen om geld en tijd te spenderen aan het herstellen van de schade die hij heeft aangebracht.
  - **De Geavanceerde** - Dit type indringer probeert je systeem te gebruiken om populair en berucht te worden. Hij kan je systeem gebruiken om aandacht te vragen voor zijn bekwaamheden.
  - **De Concurrentie** - Dit type indringer is geïnteresseerd in wat voor gegevens je op je systeem hebt. Het kan iemand zijn die denkt dat je iets hebt waarvan hij kan profiteren, financieel of anderszins.
  - **De Lener** - Dit type indringer is geïnteresseerd in het 'winkelen' op je systeem om de daarbij verkregen middelen voor eigen doeleinden te gebruiken. Het typeert hem om chat of irq servers te draaien, porno archiefsites of zelfs DNS servers.
  - **Haasje-over** - Dit type indringer is alleen geïnteresseerd in je systeem om het te gebruiken om andere systemen binnen te dringen. Als je systeem voorzien is van veel verbindingen of een poort is naar een aantal interne hosts, zou je dit type tegen kunnen komen om te proberen je systeem te beschadigen.

- *Kwetsbaarheid* beschrijft hoe goed beveiligd je computer is vanaf een ander netwerk en de mogelijkheid die iemand heeft om onbevoegd toegang te verkrijgen. Wat staat er op het spel als iemand inbreekt in je systeem? Natuurlijk zullen de zorgen van een dynamische PPP thuisgebruiker verschillen van die van een bedrijf die zijn machine met het Internet of een ander groot netwerk heeft verbonden. Hoeveel tijd zal het in beslag nemen om alle gegevens die verloren zijn gegaan te herstellen/opnieuw aan te maken? Nu wat tijd investeren kan tien keer zoveel tijd later besparen als je de gegevens die verloren zijn gegaan opnieuw aan moet maken. Heb je je back-up strategie gecontroleerd en je gegevens recentelijk geverifieerd?

2.4 Een beveiligingsbeleid ontwikkelen

Creëer een eenvoudig, algemeen beleid voor je systeem dat je gebruikers gemakkelijk kunnen begrijpen en volgen. Het moet zowel de gegevens als de privacy van de gebruikers beschermen. Enkele dingen die je kunt overwegen om toe te voegen zijn: wie heeft toegang tot het systeem (Kan mijn vriend mijn account gebruiken?), wie is er bevoegd om software op het systeem te installeren, wie bezit welke gegevens, herstel na een ramp en passend gebruik van het systeem.

Een algemeen geaccepteerd beveiligingsbeleid begint met de zin:

Dat wat niet toegestaan is, is verboden.

Dit betekent dat, tenzij je een gebruiker toegang tot een dienst toestaat, die gebruiker geen gebruik mag maken van die dienst totdat je toegang toestaat. Verzeker jezelf ervan dat het beleid werkt op je reguliere gebruikersaccount. Zeggen "Ach, ik word geen wijs uit dat permissie-probleem, ik doe het wel als root", kan leiden tot beveiligingslekken die erg voor de hand liggend zijn, zelfs degenen die nog niet misbruikt zijn.

rfc1244 is een document dat beschrijft hoe je je eigen netwerk beveiligingbeleid moet creëren.

rfc1281 is een document dat een voorbeeld van een beveiligingsbeleid laat zien met een gedetailleerde beschrijving van elke stap.

Tot slot zou je het COAST-beleid archief op *ftp://coast.cs.purdue.edu/pub/doc/policy* kunnen bekijken om na te gaan hoe een beveiligingsbeleid er in werkelijkheid uitziet.

2.5 Manieren om je site te beveiligen

Dit document zal verschillende manieren bespreken waarop je de dingen waar je hard voor hebt gewerkt kunt beveiligen: je lokale machine, je gegevens, je gebruikers, je netwerk en zelfs je reputatie. Wat zou er gebeuren met je reputatie als een indringer enkele gegevens van je gebruikers zou wissen? Of je website zou ontsieren? Of het collectieve project plan van je bedrijf voor het komend kwartaal zou publiceren? Als je een netwerkinstallatie overweegt, zijn er vele factoren waar je rekening mee moet houden alvorens een enkele machine aan je netwerk toe te voegen.

Zelfs als je een enkel dialup PPP account hebt of slechts een kleine site, houdt dit niet in dat indringers niet in jouw systemen geïnteresseerd zijn. Grote geavanceerde sites zijn niet de enige doelen – veel indringers willen simpelweg zoveel mogelijk sites binnendringen, ongeacht hun grootte. Bovendien kunnen ze een beveiligingslek in jouw site gebruiken om toegang te verkrijgen tot andere sites waarmee je bent verbonden.

Indringers hebben heel veel tijd en kunnen het gokken hoe je je systeem verduisterd hebt voorkomen door gewoon alle mogelijkheden te proberen. Er zijn ook een aantal redenen waarom een indringer in jouw systeem geïnteresseerd zou kunnen zijn, welke we later zullen bespreken.

2.5.1 Beveiliging van de host

Het gebied van beveiliging waar beheerders zich het meest op concentreren is wellicht de host-gebaseerde beveiliging. Dit houdt kenmerkend in het ervoor zorgen dat je eigen systeem veilig is en het hopen dat iedereen op je netwerk hetzelfde doet. Goede wachtwoorden kiezen, de lokale netwerkdiensten van je host beveiligen, de account-bestanden goed bijhouden en programma's met bekende beveiligingslekken verbeteren, zijn onder andere de dingen die onder de verantwoordelijkheid vallen van de lokale beveiligingsbeheerder. Hoewel dit absoluut noodzakelijk is, kan het een ontmoedigende taak worden als je systeem groter wordt dan een paar machines.

2.5.2 Beveiliging van het lokale netwerk

Beveiliging van het netwerk is net zo belangrijk als beveiliging van de lokale host. Met honderden, duizenden of meer computers op hetzelfde netwerk kun je er niet op vertrouwen dat al deze systemen veilig zijn. Je ervan verzekeren dat alleen geautoriseerde gebruikers je netwerk kunnen gebruiken, firewalls bouwen, een hoge mate van versleuteling gebruiken en zeker weten dat er geen "louche"(dus onveilige) machines met je netwerk verbonden zijn, maakt allemaal deel uit van de taken van de beveiligingsbeheerder van een netwerk.

Dit document behandelt enkele van de technieken die worden gebruikt om je site te beveiligen en zal je hopelijk enkele manieren laten zien om te voorkomen dat een indringer toegang krijgt tot wat je probeert te beschermen.

2.5.3 Beveiliging door onduidelijkheid

Een soort beveiliging die besproken moet worden is "beveiliging door onduidelijkheid". Dit betekent bijvoorbeeld het verplaatsen van een dienst die bekende beveiligingskwetsbaarheden heeft naar een niet-standaard poort in de hoop dat aanvallers het niet in de gaten hebben en het dus niet misbruiken. Wees gerust dat ze kunnen vaststellen dat het er is en dat ze het zullen misbruiken. Beveiliging door onduidelijkheid is helemaal geen beveiliging. Simpelweg omdat het feit dat je een kleine site hebt of niet te veel opvalt niet inhoudt dat een indringer niet geïnteresseerd zal zijn in wat je hebt. We zullen bespreken wat je beschermt in de volgende paragrafen.

2.6 Organisatie van dit document

Dit document is verdeeld in een aantal paragrafen. Ze behandelen verscheidene algemene beveiligingskwesties. De eerste, 3 (Fysieke beveiliging), behandelt hoe je je fysieke machine moet beschermen tegen geknoei. De tweede, 4 (Lokale beveiliging), beschrijft hoe je je systeem moet beschermen tegen geknoei van lokale gebruikers. De derde, 5 (Beveiliging van bestanden en bestandssystemen), laat zien hoe je bestandssystemen en permissies op bestanden moet instellen. De volgende, 6 (Wachtwoordbeveiliging en -versleuteling), bespreekt hoe je versleuteling kunt gebruiken om je machine en netwerk beter te beveiligen. 7 (Beveiliging van de kernel) bespreekt welke kernelopties je moet instellen of je bewust van moet zijn voor een veiliger systeem. 8 (Beveiliging van het netwerk) beschrijft hoe je je Linux systeem beter kunt beveiligen tegen netwerkaanvallen. 9 (Beveiligingsvoorbereidingen) bespreekt hoe je je machine(s) moet voorbereiden voor je ze on-line brengt. De volgende, 10 (Wat te doen tijdens en na een inbraak), bespreekt wat te doen als je een aanval op je systeem constateert of ontdekt dat dit recentelijk is gebeurd. In 11 (Bronnen) worden enkele primaire bronnen opgesomd waar je meer over beveiliging kunt vinden. In de V en A paragraaf 13 (Veel gestelde vragen) worden enkele veel gestelde vragen beantwoord en tot slot volgt een conclusie in 14 (Conclusie).

De twee belangrijkste punten die je je moet realiseren als je dit document leest, zijn:

- Wees je bewust van je systeem. Controleer systeemlogs zoals `/var/log/messages`, houd een oogje op je systeem en
- Houd je systeem up-to-date door je ervan te verzekeren dat je de meest recente software versies hebt geïnstalleerd en verbeteringen hebt aangebracht bij beveiligingswaarschuwingen. Dit gewoonweg doen zal helpen om je systeem merkbaar veiliger te maken.

3 Fysieke beveiliging

De eerste laag van beveiliging waar je rekening mee moet houden is de fysieke beveiliging van je computersystemen. Wie heeft directe fysieke toegang tot je machine? Zouden ze dat ook moeten hebben? Kun je je machine beschermen tegen hun geknoei? Zou je dat ook moeten doen?

Hoeveel fysieke beveiliging je nodig hebt op je systeem is erg afhankelijk van je situatie en/of budget.

Als je een thuisgebruiker bent, zul je waarschijnlijk niet veel nodig hebben (alhoewel je misschien je machine wilt beschermen tegen het geknoei van kinderen of hinderlijke familieleden). Als je in een laboratorium bent, zul je aanzienlijk meer nodig hebben, maar gebruikers moeten wel hun werk kunnen doen op hun machines. Veel van de volgende paragrafen bieden uitkomst. Als je in een kantoor bent, kun je wel of niet je machine beveiligen na kantoortijd of als je weg bent. Bij sommige bedrijven leidt het onbeveiligd achterlaten van je computer tot ontslag.

Voor de hand liggende fysieke beveiligingsmethoden als sloten op deuren, kabels, afgesloten kasten en videobewaking zijn allemaal goede ideeën, maar vallen buiten de strekking van dit document. :)

3.1 Computersloten

Veel moderne PC-kasten bieden een voorziening om ze öp slot te doen". Gewoonlijk zal dit een socket aan de voorkant van de kast zijn, waarmee je met een bijgeleverd sleuteltje de computer op slot kunt zetten of van het slot kunt halen. Kastsloten kunnen helpen voorkomen dat iemand je PC steelt of de kast opent en je hardware rechtstreeks manipuleert/steelt. Soms kunnen ze ook voorkomen dat iemand je computer opnieuw opstart vanaf z'n eigen floppy of andere hardware.

Deze kastsloten doen verschillende dingen al naar gelang de ondersteuning in het moederbord en de wijze waarop de kast is gemaakt. Op veel PC's is het zo gemaakt dat je de kast moet openbreken om hem open te krijgen. Op enkele andere laten ze je geen nieuwe toetsenborden of muizen aansluiten. Raadpleeg de voorschriften van je moederbord of kast voor meer informatie. Dit kan soms een erg bruikbare voorziening zijn, ondanks dat de sloten gewoonlijk van erg lage kwaliteit zijn en makkelijk kunnen worden gesloopt door aanvallers met slotenmakersgereedschap.

Sommige machines (voornamelijk SPARC's en Mac's) hebben een oog aan de achterkant waar je een kabel door kunt halen, zodat aanvallers de kabel moeten doorknippen of de kast moeten slopen om erin te kunnen komen. Een hangslot of een combinatieslot erdoor is een afschrikwekkend middel voor iemand die je machine wil stelen.

3.2 Beveiliging van de BIOS

De BIOS is het laagste niveau van software dat je x86-gebaseerde hardware configureert of manipuleert. LILO en andere Linux bootmethodes benaderen de BIOS om vast te stellen hoe je Linux machine opgestart moet worden. Andere hardware waar Linux op draait heeft vergelijkbare software (OpenFirmware op Mac's en nieuwe Sun's, Sun boot PROM, enz...). Je kunt je BIOS gebruiken om te voorkomen dat aanvallers je machine opnieuw opstarten en je Linux systeem manipuleren.

In de BIOS van veel PC's kun je een boot wachtwoord instellen. Dit verschaft niet zo heel veel beveiliging (de BIOS kan worden gereset of verwijderd als iemand in de kast kan komen), maar het kan een goed afschrikwekkend middel zijn (d.w.z. het kost tijd en laat sporen van geknoei na). Op dezelfde wijze kan op S/Linux (Linux voor SPARC(tm) processor machines) je EEPROM worden ingesteld om een boot-wachtwoord te vereisen. Dit kan vertragend werken voor aanvallers.

Veel x86 BIOS'sen staan je ook toe om diverse andere goede beveiligingsinstellingen te specificeren. Raadpleeg je BIOS handleiding of bekijk het de volgende keer als je opstart. Sommige BIOS'sen staan bijvoorbeeld het opstarten vanaf floppy drives niet toe en sommigen vereisen wachtwoorden om toegang te krijgen tot enkele BIOS voorzieningen.

Let op: Als je een server machine hebt en je stelt een boot wachtwoord in, zal je machine niet onbeheerd opstarten. Houd in gedachten dat je in geval van een stroomstoring moet komen opdagen om het wachtwoord in te toetsen. ;(

3.3 Beveiliging van de Boot Loader

In de verschillende Linux boot loaders kan ook een wachtwoord ingesteld worden. LILO heeft bijvoorbeeld `password` en `restricted` instellingen; `password` vereist een wachtwoord bij het opstarten, terwijl `restricted` alleen een wachtwoord bij het opstarten vereist als je opties specificeert (zoals `single`) bij de `LILO` prompt.

Uit de lilo.conf man pagina:

```
password=password
            The per-image option `password=...' (see below) applies to all images.
restricted
            The per-image option `restricted' (see below) applies to all images.

      password=password
            Protect the image by a password.

      restricted
            A password is only required to boot the image if
            parameters are specified  on  the  command  line
            (e.g. single).
```

Houd in gedachten dat wanneer je al deze wachtwoorden instelt, je ze ook moet onthouden. :) Onthoud ook dat deze wachtwoorden de vastbesloten aanvaller louter zullen vertragen. Ze kunnen niet voorkomen dat iemand opstart vanaf een floppy en je root partitie mount. Als je beveiliging samen met een boot loader gebruikt, kun je net zo goed het opstarten vanaf een floppy uitschakelen in de BIOS van je computer en de BIOS met een wachtwoord beschermen.

Als iemand beveiligingsgerelateerde informatie van een andere boot loader heeft, zouden we dat graag willen horen. (`grub`, `silo`, `milo`, `linload`, enz.)

Let op: Als je een server machine hebt en je stelt een boot wachtwoord in, zal je machine *niet* onbeheerd opstarten. Houd in gedachten dat je in geval van een stroomstoring moet komen opdagen om het wachtwoord in te toetsen. ;(

3.4 xlock en vlock

Als je af en toe van je machine afdwaalt, is het wel aardig dat je de mogelijkheid hebt om je console te "vergrendelen"zodat niemand je werk kan verknoeien of bekijken. Twee programma's die dat doen zijn `xlock` en `vlock`.

`xlock` is een X beeldschermvergrendeling. Het zou bij elke Linux distributie moeten zitten die X ondersteunt. Bekijk hiervoor de man pagina voor meer opties, maar in het algemeen kun je `xlock` draaien vanaf elke xterm op je console en het zal het beeldscherm vergrendelen en om een wachtwoord vragen om te ontgrendelen.

`vlock` is een simpel klein programma waarmee je enkele of alle virtuele consoles op je Linux box kunt vergrendelen. Je kunt alleen degene waarop je aan het werk bent vergrendelen, of allemaal. Als je er maar één afsluit, kunnen anderen binnenkomen en de console gebruiken; ze kunnen alleen niet jouw virtuele console gebruiken totdat je hem ontgrendelt. `vlock` wordt geleverd bij Redhat Linux, maar de weg die jij moet bewandelen om zover te komen kan anders zijn.

Natuurlijk zal het vergrendelen van je console iemand ervan weerhouden om met je werk te knoeien, maar het zal ze niet beletten om je machine opnieuw op te starten of anderszins je werk te verstoren. Het weerhoudt ze ook niet om je machine te benaderen vanaf een andere machine op het netwerk en problemen te veroorzaken.

Maar belangrijker, het weerhoudt iemand niet om het X Window systeem geheel te verlaten en naar een normale virtuele login prompt te gaan of naar de virtuele console waarvan X11 is opgestart en het op non-actief te zetten om zodoende jouw privileges te verkrijgen. Om deze reden zou je moeten overwegen om het alleen in combinatie met xdm te gebruiken.

3.5 Fysieke beveiligingsgevaren opsporen

Het eerste waar je altijd op moet letten is of je machine opnieuw is opgestart. Omdat Linux een robuust en stabiel besturingssysteem is, zijn de enige keren dat het opnieuw opgestart moet worden, de keren dat *jij* het buiten gebruik stelt voor upgrades van het besturingssysteem, wisseling van hardware of iets dergelijks. Als je machine opnieuw is opgestart buiten jouw medeweten, kan dat een teken zijn dat een indringer je systeem in gevaar brengt. Veel van de manieren waarop je machine in gevaar kan worden gebracht, vereisen dat de indringer je machine opnieuw opstart of hem uitschakelt.

Controleer op tekenen van geknoei met de kast of de omgeving van de computer. Hoewel veel indringers hun sporen in de logbestanden verwijderen, is het een goed idee om ze te controleren en enige discrepantie op te merken.

Het is ook een goed idee om loggegevens op een veilige plaats op te slaan, bijvoorbeeld op een toegewijde log server op je goed beschermde netwerk. Als een machine eenmaal te maken heeft gehad met een aanval, zijn loggegevens van weinig nut meer, omdat ze hoogstwaarschijnlijk ook zijn aangepast door de indringer.

De syslog daemon kan zo worden ingesteld dat hij loggegevens automatisch naar een centrale syslog server stuurt, maar dit wordt kenmerkend ongecodeerd gedaan, zodat een indringer de mogelijkheid heeft om de gegevens te bekijken terwijl ze worden verzonden. Dit kan informatie over je netwerk blootgeven, waarvan het niet de bedoeling is dat het openbaar wordt. Er zijn syslog daemons beschikbaar die de gegevens coderen terwijl ze worden verzonden.

Wees je ook bewust dat het namaken van syslogberichten gemakkelijk is – met een speciaal programma dat reeds gepubliceerd is. Syslog accepteert zelfs net log entries die het doen voorkomen alsof ze van de lokale host afkomstig zijn, zonder enige aanwijzing over hun ware herkomst.

Enkele dingen die je moet controleren in je logs:

- korte of onvolledige logs
- logs die vreemde tijdstippen bevatten

- logs met verkeerde permissies of eigendomsrecht
- registraties van het opnieuw opstarten van het systeem of diensten
- ontbrekende logs
- `su` entries of logins vanaf vreemde plaatsen

We zullen systeemloggegevens 9.4 (later) in deze HOWTO bespreken.

4 Lokale beveiliging

Het volgende waar we naar gaan kijken is de beveiliging van je systeem tegen aanvallen van lokale gebruikers. Zeiden we zojuist *lokale* gebruikers? Ja!

Toegang verkrijgen tot een lokaal gebruikersaccount is een van de eerste dingen die indringers op een systeem proberen op hun weg naar het misbruiken van het root account. Met een lakse lokale beveiliging kunnen ze hun normale gebruikerstoegang üpgraden"naar een roottoegang door gebruik te maken van een verscheidenheid aan bugs en minnetjes ingestelde lokale diensten. Als je ervoor zorgt dat je lokale beveiliging waterdicht is, zal de indringer nog een hindernis moeten nemen.

Lokale gebruikers kunnen ook een hoop schade aanrichten op je systeem, zelfs (juist) als ze inderdaad diegene zijn die ze zeggen dat ze zijn. Het verstrekken van accounts aan mensen die je niet kent of van wie je geen achtergrondinformatie hebt, is een erg slecht idee.

4.1 Nieuwe accounts aanmaken

Je moet ervan overtuigd zijn dat je gebruikersaccounts verschaft met slechts de minimale vereisten voor de taak die ze moeten doen. Als je je zoon (10 jaar) een account verschaft, zul je wellicht willen dat hij alleen toegang heeft tot een tekstverwerker of tekenprogramma, maar geen gegevens kan verwijderen die niet van hem zijn.

Enkele goede vuistregels als je andere mensen rechtmatige toegang tot je Linux-machine toestaat:

- Geef ze het minimale aantal privileges dat ze nodig hebben.
- Weet wanneer/waar vandaan ze inloggen of waar vandaan ze zouden moeten inloggen.
- Verwijder niet gebruikte accounts.
- Het gebruik van hetzelfde gebruikers ID op alle computers en netwerken is aan te raden om het onderhoud van accounts te vereenvoudigen. Ook staat het een eenvoudigere analyse van loggegevens toe.
- Het aanmaken van groeps-gebruiker ID's zou absoluut verboden moeten zijn. Gebruikersaccounts zorgen ook voor verantwoordelijkheid en dit is bij groepsaccounts niet mogelijk.

Veel lokale gebruikersaccounts die gebruikt worden bij een aanval zijn in geen maanden of jaren meer gebruikt. Omdat niemand ze gebruikt, zorgen ze voor een ideaal aanvalsvoertuig.

4.2 Root beveiliging

Het meest gewilde account op je machine is het root (superuser) account. Dit account heeft zeggenschap over de gehele machine, wat ook zeggenschap kan inhouden over andere machines op het netwerk. Onthoud dat je het root account alleen moet gebruiken voor hele korte specifieke taken en dat het meestal uitgevoerd moet worden als een normale gebruiker. Zelfs kleine foutjes als je ingelogd bent als root kunnen problemen veroorzaken. Hoe korter je ingelogd bent met root privileges, hoe veiliger het is.

Enkele trucks om te voorkomen dat je je computer overhoop haalt als root:

- Als je bezig bent met een complex commando, probeer het dan eerst op een niet-destructieve manier ... vooral commando's die wildcards gebruiken: als je bijvoorbeeld `"rm foo*.bar"`

  wilt doen, doe dan eerst `"ls foo*.bar"`

  om zeker te weten dat je de bestanden verwijdert die je wilde verwijderen. Het gebruik van `echo` in plaats van destructieve commando's werkt soms ook.
- Voorzie je gebruikers van een standaard alias voor het rm commando om een bevestiging te vragen voordat ze bestanden verwijderen.
- Wordt alleen root om enkele specifieke taken uit te voeren. Als je jezelf erop betrapt dat je probeert uit te vinden hoe iets werkt, ga dan eerst terug naar de gewone gebruiker-shell, totdat je `zeker` weet wat er als root gedaan moet worden.
- Het command path voor de root gebruiker is erg belangrijk. Het command path (dat wil zeggen de `PATH` omgevingsvariabele) specificeert de directory's waarin de shell zoekt naar programma's. Probeer het command path voor de root gebruiker zoveel mogelijk te beperken en zet *nooit* een `.` (hetgeen betekent "de huidige directory") in je PATH. Zorg er bovendien voor dat je nooit directory's met schrijfpermissie in je zoekpad hebt, omdat dit aanvallers toestaat om bestaande binary's aan te passen of nieuwe binary's aan je zoekpad toe te voegen, hetgeen hen toestaat als root te opereren de volgende keer dat je dat commando uitvoert.
- Gebruik nooit de rlogin/rsh/rexec tools (genaamd de r-utility's) als root. Ze zijn onderhevig aan vele soorten aanvallen en zijn ronduit gevaarlijk als je ze uitvoert als root. Maak nooit een `.rhosts` bestand aan voor root.
- Het `/etc/securetty` bestand bevat een lijst met terminals waarop root in kan loggen. Standaard (onder Red Hat Linux) is dit ingesteld op alleen de lokale virtuele consoles (vty's). Wees erg voorzichtig met het toevoegen van iets anders aan dit bestand. Je zou indirect op je normale gebruikersaccount in moeten kunnen loggen en vervolgens `su` als dat nodig is (hopelijk via `6.4 (ssh)` of een ander versleuteld kanaal), zodat er geen reden is waarom je direct als root in zou moeten loggen.
- Wees altijd langzaam en weloverwogen als je bezig bent als root. Je acties kunnen een heleboel dingen beïnvloeden. Denk na voordat je typt!

Als het absoluut noodzakelijk is om iemand (hopelijk erg vertrouwd) roottoegang tot je machine toe te staan, zijn er een aantal tools die kunnen helpen. `sudo` staat gebruikers toe hun wachtwoord te gebruiken om toegang te krijgen tot een beperkte set commando's als root. Zodoende kun je, bijvoorbeeld, een gebruiker in staat stellen om verwijderbare media uit te werpen en te mounten op je Linux box, maar verder geen andere root privileges te hebben. `sudo` houdt ook een log bij van alle geslaagde en mislukte sudo pogingen, zodat je uit kunt zoeken wie welk commando gebruikte om wat te doen. Om deze reden werkt `sudo` zelfs goed op plaatsen waar een aantal mensen root toegang hebben, omdat het je helpt om aangebrachte wijzigingen bij te houden.

Hoewel `sudo` gebruikt kan worden om bepaalde gebruikers bepaalde privileges voor bepaalde taken te geven, heeft het een aantal tekortkomingen. Het moet alleen gebruikt worden voor een beperkt takenpakket, zoals een server opnieuw opstarten of nieuwe gebruikers toevoegen. Elk programma waarbij het uitwijken naar een shell mogelijk is, zal root toegang verschaffen aan een gebruiker die het aanroept via `sudo`. Dit omvat de meeste tekstverwerkers bijvoorbeeld. Ook een programma zo onschadelijk als `/bin/cat` kan worden gebruikt om bestanden te overschrijven, waarmee het mogelijk zou kunnen worden om root te misbruiken. Beschouw `sudo` als een middel voor het toekennen van verantwoordelijkheden en verwacht niet dat het de root gebruiker kan vervangen en tevens veilig is.

5 Beveiliging van bestanden en bestandssystemen

Een paar minuten van voorbereiding en planning vooraf, voordat je je systemen online zet, kan helpen ze (en de gegevens die erop opgeslagen zijn) te beschermen.

- Er zou nooit een reden mogen zijn om toe te staan dat SUID/SGID programma's uitgevoerd mogen worden vanuit de home directory's van gebruikers. Gebruik de `nosuid` optie in `/etc/fstab` voor partities die beschrijfbaar zijn door anderen dan root. Je wilt misschien ook `nodev` en `noexec` toepassen op de home partities van gebruikers, evenals op `/var`, dus het uitvoeren van programma's is verboden, net als het creëren van character of block devices, wat trouwens toch nooit noodzakelijk zou moeten zijn.
- Als je bestandssystemen exporteert middels NFS, let er dan op dat je `/etc/exports` instelt met de meest beperkte toegang mogelijk. Dit houdt in dat je geen gebruik moet maken van wildcards, geen root schrijfpermissie toestaat en waar mogelijk read-only exporteert.
- Stel de bestandsaanmaak `umask` van je gebruikers zo beperkt mogelijk in. Zie 5.1 (umask instellingen).
- Als je gebruik maakt van een netwerk bestandssysteem zoals NFS om bestandssystemen te mounten, let er dan op dat je `/etc/exports` instelt met geschikte beperkingen. Kenmerkend is het gebruik van 'nodev', 'nosuid' en misschien 'noexec' wenselijk.
- Stel limieten in voor het bestandssysteem in plaats van het toestaan van `unlimited`, wat standaard is. Je kunt de limieten per gebruiker beheren door gebruik te maken van de resource-limits PAM module en `/etc/pam.d/limits.conf`. De limieten voor de groep `users` kunnen er bijvoorbeeld zo uit zien:

  ```
  @users     hard  core    0
  @users     hard  nproc   50
  @users     hard  rss     5000
  ```

  Dit zegt: verbied het creëren van core bestanden, beperk het aantal processen tot 50 en beperk het geheugengebruik tot 5M per gebruiker.
- De `/var/log/wtmp` en `/var/run/utmp` bestanden bevatten de login records van alle gebruikers op je systeem. Hun zuiverheid moet behouden blijven, omdat ze gebruikt kunnen worden om te bepalen wanneer en waar vandaan een gebruiker (of een mogelijke indringer) je systeem is binnengekomen. Deze bestanden moeten ook `644` permissies hebben, zonder dat het invloed heeft op het normale systeemgebruik.
- Het onveranderlijke bit kan gebruikt worden om te voorkomen dat een bestand dat beschermd moet worden per ongeluk verwijderd of overschreven wordt. Het voorkomt ook dat iemand een symbolische link aanmaakt naar het bestand (zulke symbolische links zijn de bron geweest van aanvallen die betrekking hadden op het verwijderen van `/etc/passwd` of `/etc/shadow`). Zie de `chattr`(1) man pagina voor informatie over het onveranderlijke bit.

- SUID en SGID bestanden op je systeem vormen een mogelijk beveiligingsrisico en zullen nauwgezet in de gaten gehouden moeten worden. Omdat deze programma's speciale privileges toekennen aan de gebruiker die ze uitvoert, is het noodzakelijk om je ervan te verzekeren dat er geen onveilige programma's geïnstalleerd zijn. Een favoriet trucje van crackers is om SUID-root programma's te misbruiken en dan een SUID programma achter te laten als een achterdeur om de volgende keer binnen te komen, zelfs als het oorspronkelijke gat is gedicht.

  Zoek alle SUID/SGID programma's op je systeem en houd bij wat ze zijn, zodat je je bewust bent van enige veranderingen die kunnen duiden op een mogelijke indringer. Gebruik het volgende commando om te zoeken naar alle SUID/SGID programma's op je systeem:

  ```
  root#  find / -type f \( -perm -04000 -o -perm -02000 \)
  ```

  De Debian distributie voert elke nacht een taak uit om te bepalen welke SUID programma's er bestaan. Het vergelijkt het dan met de uitvoer van de vorige nacht. Je kunt kijken in `/var/log/setuid*` voor deze log.

  Je kunt de SUID of SGID permissies op een verdacht programma verwijderen met `chmod` en ze dan terugzetten als je denkt dat dat absoluut nodig is.

- Bestanden met schrijfpermissie voor iedereen, in het bijzonder systeembestanden, kunnen een beveiligingslek zijn als een cracker toegang krijgt tot je systeem en ze aanpast. Directory's met schrijftoegang voor iedereen zijn bovendien gevaarlijk, omdat ze een cracker toestaan om naar wens bestanden toe te voegen of te verwijderen. Om alle bestanden met schrijfpermissie voor iedereen op je systeem te vinden, gebruik je het volgende commando:

  ```
  root# find / -perm -2 ! -type l -ls
  ```

  en verzeker je ervan dat je weet waarom deze bestanden schrijfpermissie hebben. Bij normaal gebruik zullen verscheidene bestanden schrijfpermissie voor iedereen hebben, inclusief enkele van `/dev` en symbolische links, dus middels `!  -type l` worden deze niet meegenomen door het voorgaande `find` commando.

- Bestanden die niemand toebehoren kunnen ook een indicatie zijn dat een indringer je systeem is binnengedrongen. Je kunt bestanden op je systeem die geen eigenaar hebben of tot geen enkele groep behoren, vinden met het commando:

  ```
  root# find / -nouser -o -nogroup -print
  ```

- Het zoeken naar `.rhosts` bestanden zou onderdeel uit moeten maken van je vaste systeembeheertaken, omdat deze bestanden niet toegestaan zouden mogen zijn op je systeem. Onthoud, een cracker heeft slechts één onveilig account nodig om mogelijk toegang tot je gehele netwerk te verkrijgen. Je kunt alle `.rhosts` bestanden op je systeem vinden met het volgende commando:

  ```
  root# find /home -name .rhosts -print
  ```

- Tot slot, voordat je de permissies op welke systeembestanden dan ook gaat wijzigen, moet je je ervan verzekeren dat je begrijpt wat je aan het doen bent. Verander nooit de permissies van een bestand, omdat het de makkelijkste manier lijkt om dingen werkend te krijgen. Bepaal altijd waarom het bestand die permissie heeft alvorens het te veranderen.

5.1 Umask instellingen

Het `umask` commando kan gebruikt worden om de standaard manier waarop bestanden op je systeem aangemaakt worden te bepalen. Het is de achtste aanvulling van de gewenste bestandsmodus. Als bestanden worden aangemaakt zonder te letten op hun permissie-instellingen, kan de gebruiker onbewust lees- of schrijfpermissie geven aan iemand die deze permissie niet mag hebben. Kenmerkende `umask` instellingen bevatten `022`, `027` en `077` (welke de meest beperkte is). Normaal gesproken wordt umask ingesteld in `/etc/profile`, zodat het van toepassing is op alle gebruikers op het systeem. Het bestandsaanmaak-mask kan worden berekend door de gewenste waarde af te trekken van 777. Met andere woorden, een umask van 777 heeft tot gevolg dat nieuw aangemaakte bestanden voor niemand lees-, schrijf- of uitvoerpermissies bevatten. Een mask van 666 heeft tot gevolg dat nieuw aangemaakte bestanden een mask van 111 hebben. Je kunt bijvoorbeeld een regel hebben die er zo uitziet:

```
# Set the user's default umask
  umask 033
```

Let erop dat je de umask van root `077` maakt, wat lees-, schrijf en uitvoerpermissie voor andere gebruikers uitschakelt, tenzij het expliciet is gewijzigd met `chmod`. In dit geval zullen nieuw aangemaakte directory's 744 permissies hebben, verkregen door 033 af te trekken van 777. Nieuw aangemaakte bestanden die gebruik maken van de 033 umask, zullen permissies van 644 hebben.

Als je Red Hat gebruikt en je houdt aan hun gebruiker- en groep-ID aanmaakschema (User Private Groups), is het alleen noodzakelijk om `002` voor een `umask` te gebruiken. Dit komt door het feit dat de standaard instelling één gebruiker per groep is.

5.2 Bestandspermissies

Het is belangrijk om je ervan te verzekeren dat je systeembestanden niet open staan voor aanpassingen door gebruikers en groepen die zulk systeemonderhoud niet zouden moeten doen.

Unix scheidt toegangsbeheer op bestanden en directory's volgens drie kenmerken: eigenaar, groep en anderen. Er is altijd exact één eigenaar, een willekeurig aantal leden van de groep en alle anderen.

Een korte uitleg van Unix permissies:

Eigendom - Welke gebruiker(s) en groep(en) heeft/hebben het beheer over de permissie-instellingen van de node en parent van de node.

Permissies - Bits die kunnen worden ingesteld of opnieuw ingesteld kunnen worden om bepaalde soorten toegang tot ze te kunnen verlenen. Permissies voor directory's kunnen een andere betekenis hebben dan dezelfde set permissies voor bestanden.

Lezen:

- De mogelijkheid hebben om de inhoud van een bestand te kunnen lezen
- De mogelijkheid hebben om een directory te kunnen lezen

Schrijven:

- De mogelijkheid hebben om een bestand te kunnen toevoegen of wijzigen
- De mogelijkheid hebben om bestanden in een directory te kunnen verwijderen of verplaatsen

Uitvoeren:

- De mogelijkheid hebben om een binair programma of shell script uit te kunnen voeren
- De mogelijkheid hebben om in een directory te kunnen zoeken, gecombineerd met leestoegang

Save Text attribuut: (Voor directory's)

Het "sticky bit"heeft ook een andere betekenis als het toegepast wordt op directory's dan wanneer het toegepast wordt op bestanden. Als het "sticky bit"wordt ingesteld op een directory, mag een gebruiker alleen bestanden verwijderen die zijn eigendom zijn of waarvoor hem expliciet schrijfpermissie is verleend, zelfs wanneer hij schrijfpermissie heeft voor de directory. Dit is ontworpen voor directory's als `/tmp`, die schrijfpermissie voor iedereen hebben, maar waar het misschien niet wenselijk is dat elke gebruiker naar wens bestanden kan verwijderen. Het "sticky bitïs te zien als een `t` in een lange directory opsomming.

SUID attribuut: (Voor bestanden)

Dit beschrijft set-uder-id permissies op het bestand. Als de set-user-ID access mode is ingesteld in de eigendomsrechten en het bestand is executable, zullen processen die het uitvoeren toegang verkrijgen tot systeembronnen, gebaseerd op de gebruiker die eigenaar van het bestand is, in tegenstelling tot de gebruiker die het proces heeft aangemaakt. Dit is de oorzaak van de vele "buffer overflow"misbruiken.

SGID attribuut: (Voor bestanden)

Indien ingesteld in de groeppermissies, beheert deze bit de "set-group-ID"status van een bestand. Dit gaat op dezelfde manier als SUID, behalve dat het nu op de groep betrekking heeft. Het bestand moet executable zijn voordat dit enig effect kan hebben.

SGID attribuut: (Voor directory's)

Als je het SGID bit op een directory instelt (met `chmod g+s` *`directory`*), hebben bestanden die in die directory aangemaakt zijn hun groep ingesteld op de groep van de directory.

Jij - De eigenaar van het bestand

Groep - De groep waar je toe behoort

Iedereen - Iedereen op het systeem die niet de eigenaar is en geen deel uitmaakt van de groep

Bestandsvoorbeeld:

```
-rw-r--r--  1 kevin  users                114 Aug 28  1997 .zlogin
1e bit - directory?                     (nee)
 2e bit - lezen door eigenaar?           (ja, door kevin)
  3e bit - schrijven door eigenaar?       (ja, door kevin)
   4e bit - uitvoeren door eigenaar?       (nee)
    5e bit - lezen door groep?              (ja, door users)
     6e bit - schrijven door groep?          (nee)
      7e bit - uitvoeren door groep?          (nee)
       8e bit - lezen door iedereen?           (ja, door iedereen)
        9e bit - schrijven door iedereen?       (nee)
         10e bit - uitvoeren door iedereen?      (nee)
```

De volgende regels zijn voorbeelden van de minimale set van permissies die vereist zijn voor de beschreven toegang. Misschien wil je meer permissies geven dan hetgeen hier opgesomd is, maar dit zou moeten beschrijven wat deze minimale permissies op bestanden doen:

```
-r-------- Staat leestoegang op het bestand toe aan de eigenaar.
--w------- Staat het de eigenaar toe om het bestand aan te passen of te
           verwijderen.(Merk op dat iedereen met schrijfpermissie op de
           directory waar het bestand zich in bevindt, het kan overschrijven
           en dus verwijderen.)
---x------ De eigenaar kan dit programma uitvoeren, maar geen shell scripts
           waarvoor ook nog leestoegang nodig is.
---s------ Uitvoeren is mogelijk met een effectieve User-ID = naar eigenaar.
--------s- Uitvoeren is mogelijk met een effectieve Group-ID = naar groep.
-rw------T Geen update van "last modified time". Wordt meestal gebruikt voor
           swap bestanden.
---t------ Geen effect.(voorheen sticky bit)
```

Directoryvoorbeeld:

```
drwxr-xr-x  3 kevin  users             512 Sep 19 13:47 .public_html/
1e bit - directory?                   (ja, het bevat veel bestanden)
 2e bit - lezen door eigenaar?         (ja, door kevin)
  3e bit - schrijven door eigenaar?     (ja, door kevin)
   4e bit - uitvoeren door eigenaar?     (ja, door kevin)
    5e bit - lezen door groep?            (ja, door users)
     6e bit - schrijven door groep?        (nee)
      7e bit - uitvoeren door groep?        (ja, door users)
       8e bit - lezen door iedereen?         (ja, door iedereen)
        9e bit - schrijven door iedereen?     (nee)
         10e bit - uitvoeren door iedereen?    (ja, door iedereeen)
```

De volgende regels zijn voorbeelden van de minimale set van permissies die vereist zijn voor de beschreven toegang. Misschien wil je meer permissies geven dat hetgeen hier opgesomd is, maar dit zou moeten beschrijven wat deze minimale permissies op directory's doen:

```
dr-------- De inhoud kan opgesomd worden, maar bestandsattributen kunnen niet
           gelezen worden.
d--x------ De directory is toegankelijk en kan in opdrachten worden verwerkt
           waarin het directorypad wordt gebruikt.
dr-x------ Bestandsattributen kunnen gelezen worden door de eigenaar
d-wx------ Bestanden kunnen worden aangemaakt/verwijderd, zelfs als de
           directory niet de huidige is.
d------x-t Voorkomt dat bestanden worden verwijderd door anderen met
           schrijftoegang. Wordt gebruikt bij /tmp.
d---s--s-- Geen effect.
```

Systeemconfiguratie bestanden (meestal in `/etc`) zijn meestal modus `640` (`-rw-r---`) en eigendom van root. Afhankelijk van de beveiligingsvereisten van je site kun je dit aanpassen. Laat nooit enige systeembestanden beschrijfbaar zijn voor een groep of iedereen. Sommige configuratiebestanden, waaronder `/etc/shadow`, zouden alleen leesbaar voor root moeten zijn en directory's in `/etc` zouden op z'n minst niet toegankelijk voor anderen moeten zijn.

SUID shell scripts

SUID shell scripts vormen een serieus beveiligingsrisico en om deze reden zal de kernel ze niet toejuichen. Ongeacht hoe veilig je denkt dat een shell script is, het kan worden misbruikt om een cracker een root shell te geven.

5.3 Controle op integriteit

Een andere zeer goede manier om lokale (en ook netwerk) aanvallen op je systeem op te sporen is het uitvoeren van een ïntegrity checker"zoals `Tripwire`, `Aide` of `Osiris`. Deze "Integrity checkers"voeren een aantal controles uit op al je belangrijke binary's en configuratiebestanden en vergelijkt ze met een database van eerdere, goed-bewezen waarden als een naslagwerk. Kortom, alle veranderingen in de bestanden zullen genoteerd worden.

Het is een goed idee om dit soort programma's op een floppy te installeren en dan het floppy tegen schrijven te beveiligen. Op deze manier kunnen indringers niet knoeien met de "Integrity checkeröf de database veranderen. Als je zoiets eenmaal hebt ingesteld, is het een goed idee om het uit te voeren als een onderdeel van je normale beveiligingsbeheertaken om te zien of er iets is veranderd.

Je kunt zelfs een `crontab` entry toevoegen om het controleprogramma vanaf je floppy elke nacht uit te voeren en de resultaten in de ochtend per e-mail te krijgen. Iets als:

```
# set mailto
  MAILTO=kevin
# run Tripwire
  15 05 * * * root /usr/local/adm/tcheck/tripwire
```

zal je elke morgen om 5.15 uur een verslag mailen.

"Integrety checkers"kunnen een uitkomst zijn om indringers op te sporen voordat je ze op een andere manier in de gaten krijgt. Omdat er op een doorsnee systeem veel bestanden wijzigen, moet je voorzichtig zijn in het bepalen van wat het werk van een cracker is en wat je zelf gedaan hebt.

Je kunt de open source versie van `Tripwire` vinden op *http://www.tripwire.org*, zonder kosten. Handleidingen en ondersteuning kunnen aangeschaft worden.

`Aide` vind je op *http://www.cs.tut.fi/˜rammer/aide.html*.

`Osiris` vind je op *http://www.shmoo.com/osiris/*.

5.4 Trojan Horses

"Trojan Horses"zijn genoemd naar de fabelachtige tactische zet in Homer's "Iliad". Het idee is dat een cracker een programma of binary dat er goed uitziet verspreidt en andere mensen aanmoedigt het te downloaden en het uit te voeren als root. Het programma kan vervolgens schade aanrichten op hun systeem als ze niet opletten. Terwijl ze denken dat de binary die ze zojuist hebben verkregen iets doet (en dat is wellicht ook zo), schaadt het ook hun beveiliging.

Je moet voorzichtig zijn welke programma's je installeert op je computer. Red Hat levert MD5 checksums en PGP signatures op zijn RPM bestanden, zodat je kunt verifiëren dat je het origineel installeert. Andere distributies hebben soortgelijke methoden. Je moet nooit enige onbekende binary, waarvan je de source niet hebt, als root uitvoeren! Er zijn maar weinig aanvallers bereid om source code vrij te geven, zodat het in het openbaar aan een nauwkeurig onderzoek kan worden onderworpen.

Hoewel het veelomvattend kan zijn, vergewis je er dan van dat je de source van een programma van de originele distributie site afhaalt. Als het programa uitgevoerd gaat worden als root, zorg dan dat of jij of iemand die je vertrouwt de source heeft bekeken en geverifieerd heeft.

6 Wachtwoordbeveiliging en -versleuteling

Wachtwoorden zijn één van de belangrijkste beveiligingsmogelijkheden die vandaag de dag gebruikt worden. Het is belangrijk voor zowel jezelf als je gebruikers om veilige, niet te raden, wachtwoorden te hebben. Het merendeel van de meer recente Linux distributies levert `passwd` programma's die het je niet toestaan om een makkelijk te raden wachtwoord in te stellen. Zorg dat je `passwd` programma up to date is en deze mogelijkheden heeft.

Een diepgaande verhandeling over versleuteling valt buiten de strekking van dit document, maar een inleiding is op z'n plaats. Versleuteling is erg nuttig, waarschijnlijk zelfs noodzakelijk tegenwoordig. Er zijn veel verschillende methoden om gegevens te versleutelen, elk met zijn eigen kenmerken.

De meeste Unix systemen (en Linux is geen uitzondering) gebruiken primair een eenrichtingsverkeer versleutelingsalgoritme, genaamd DES (Data Encryption Standard) om wachtwoorden te versleutelen. Dit versleutelde wachtwoord wordt dan opgeslagen in (gebruikelijk) `/etc/passwd` (of minder gebruikelijk) `/etc/shadow`. Als je probeert in te loggen wordt het wachtwoord dat je intypt opnieuw versleuteld en vergeleken met de entry in het bestand waarin je wachtwoorden worden opgeslagen. Als ze overeenkomen moet het wel hetzelfde wachtwoord zijn en word je toegang verleend. Hoewel DES een tweerichtingsverkeer versleutelingsalgoritme is (je kunt een bericht coderen en decoderen, op voorwaarde dat je de juiste sleutels hebt), is de variant die de meeste Unix systemen gebruiken de öne-way". Dit betekent dat het niet mogelijk is om de versleuteling om te keren om zodoende het wachtwoord te verkrijgen vanuit de inhoud van `/etc/passwd` (of `/etc/shadow`).

Aanvallen met brute kracht, zoals "Crackö f "John the Ripper"(zie paragraaf 6.9 ()) kunnen vaak wachtwoorden raden, tenzij je wachtwoord afdoende willekeurig is. PAM modules (zie hieronder) staan je toe om een andere versleutelingsroutine voor je wachtwoorden te gebruiken. (MD5 of iets dergelijks). Je kunt tevens Crack in je voordeel gebruiken. Overweeg het periodiek uitvoeren van Crack op je wachtwoord database om onveilige wachtwoorden te vinden. Neem vervolgens contact op met de overtredende gebruiker en vertel hem dat hij zijn wachtwoord moet veranderen.

Kijk op *http://consult.cern.ch/writeup/security/security_3.html* voor informatie over het kiezen van een goed wachtwoord.

6.1 PGP en Public-Key versleuteling

Public-key versleuteling, zoals dat gebruikt wordt voor PGP, gebruikt een sleutel voor het coderen en een sleutel voor het decoderen. Traditionele versleuteling gebruikt echter dezelfde sleutel voor het coderen en decoderen; deze sleutel moet bekend zijn bij beide partijen en zal dus op de een af andere wijze veilig van de een naar de ander overgebracht moeten worden.

Om de noodzaak van het veilig overbrengen van de coderingssleutel te verlichten, gebruikt public-key versleuteling twee afzonderlijke sleutels: een publieke sleutel en een persoonlijke sleutel. Ieders publieke sleutel is beschikbaar voor iedereen om de codering uit te voeren, terwijl tegelijkertijd iedereen zijn of haar persoonlijke sleutel heeft om berichten, die gecodeerd zijn met de juiste publieke sleutel, te decoderen.

Zowel public-key als private-key versleuteling hebben hun voordelen en je kunt over deze verschillen lezen in *The RSA Cryptography FAQ* `<http://www.rsa.com/rsalabs/newfaq/>`, genoemd aan het eind van deze paragraaf.

PGP (Pretty Good Privacy) wordt goed ondersteund onder Linux. De versies 2.6.2 en 5.0 staan er bekend om dat ze goed werken. Voor een goed eerste-beginselen-boekje over PGP en hoe het te gebruiken, kun je de PGP FAQ bekijken: *http://www.pgp.com/service/export/faq/55faq.cgi.*

Let erop dat je de versie gebruikt die geschikt is voor het land waar je woont. Als gevolg van exportbeperkingen door de regering van de VS, is het verboden om sterke versleuteling in elektronische vorm de landsgrenzen over te brengen.

Exportcontroles worden nu beheerd door EAR (Export Administration Regulations). Ze worden niet langer bepaald door ITAR.

Er is ook een stap-voor-stap gids voor het instellen van PGP onder Linux, beschikbaar op *http://mercury.chem.pitt.edu/~angel/LinuxFocus/English/November1997/article7.html*. Het is geschreven voor de internationale versie van PGP, maar het is makkelijk aan te passen aan de versie voor de Verenigde Staten. Je hebt mogelijk ook een patch nodig voor de recente versies van Linux; de patch is beschikbaar op *ftp://metalab.unc.edu/pub/Linux/apps/crypto*.

Er is een project waar gewerkt wordt aan een gratis re-implementatie van PGP met open source. GnuPG is een complete en gratis vervanging van PGP. Omdat het geen IDEA of RSA gebruikt, kan het zonder enige beperkingen gebruikt worden. GnuPG komt bijna overeen met *OpenPGP*. Zie de GNU Privacy Guard webpagina voor meer informatie: *http://www.gnupg.org/*.

Meer informatie over versleuteling kan gevonden worden in "The RSA cryptography FAQ", beschikbaar op *http://www.rsa.com/rsalabs/newfaq/*. Hier vind je informatie over begrippen als "Diffie-Hellman", "public-key cryptography", "digital certificates", enz.

6.2 SSL, S-HTTP, HTTPS en S/MIME

Gebruikers vragen vaak naar de verschillen tussen de diverse beveiligings- en versleutelingsprotocollen en hoe ze gebruikt moeten worden. Hoewel dit geen document over versleuteling is, is het een goed idee om kort uit te leggen wat elk protocol inhoudt en waar meer informatie te vinden is.

- **SSL:** - SSL, of Secure Sockets Layer, is een versleutelingsmethode die ontwikkeld is door Netscape om in beveiliging over het Internet te voorzien. Het ondersteunt diverse verschillende versleutelingsprotocollen en voorziet in client en server authenticatie. SSL opereert op de transportlaag, creëert een veilig versleuteld kanaal met gegevens en kan dus naadloos vele soorten gegevens versleutelen. Dit is in de meeste gevallen te zien wanneer een veilige site wordt benaderd om een veilig online document te bekijken met Communicator en dient als de basis voor veilige communicatie met Communicator, net als vele andere Netscape Communications gegevensversleuteling. Meer informatie kan worden gevonden op *http://www.consensus.com/security/ssl-talk-faq.html*. Informatie over Netscape's andere beveiligingsimplementaties en een goed startpunt voor deze protocollen is beschikbaar op *http://home.netscape.com/info/security-doc.html*.

- **S-HTTP:** - S-HTTP is een ander protocol dat voorziet in beveiligingsdiensten over het Internet. Het is ontworpen om te voorzien in vertrouwelijkheid, authenticatie, integriteit en niet afwijzend te zijn [kan niet aangezien worden voor iemand anders] terwijl het meerdere sleutelbeheertechnieken en cryptografische algoritmen ondersteunt via onderhandeling over de mogelijkheden tussen de betrokken partijen in iedere transactie. S-HTTP is beperkt tot de specifieke software die het uitvoert en versleutelt elk bericht individueel. [Uit de RSA Cryptography FAQ, pagina 138.]

- **S/MIME:** - S/MIME, of Secure Multipurpose Internet Mail Extension, is een versleutelingsstandaard die gebruikt wordt om e-mail en andere soorten berichten op het Internet te versleutelen. Het is een open standaard, ontwikkeld door RSA, dus het is waarschijnlijk dat we het binnenkort onder Linux tegenkomen. Meer informatie over S/MIME kan worden gevonden op *http://home.netscape.com/assist/security/smime/overview.html*.

6.3 Linux IPSEC uitvoeringen

Afgezien van CIPE en andere vormen van gegevensversleuteling, zijn er ook diverse andere uitvoeringen van IPSEC voor Linux. IPSEC is een poging van de IETF om cryptografisch-veilige communicaties op het IP netwerkniveau te creëren en om te voorzien in authenticatie, integriteit, toegangscontrole en vertrouwelijkheid. Informatie over IPSEC en ontwerpen voor Internet kan gevonden worden op *http://www.ietf.org/html.charters/ipsec-charter.html*. Je kunt er ook verwijzingen naar andere protocollen betreffende sleutelbeheer vinden en een IPSEC mailing list en archieven.

De x-kernel Linux uitvoering, die ontwikkeld wordt op de Universiteit van Arizona, gebruikt een objectgebaseerde opzet voor het uitvoeren van netwerkprotocollen genaamd x-kernel en kan worden gevonden op *http://www.cs.arizona.edu/xkernel/hpcc-blue/linux.html*. Simpel gezegd is de x-kernel een methode om berichten op het kernelniveau door te geven, wat zorgt voor een gemakkelijkere uitvoering.

Een andere vrij verkrijgbare IPSEC uitvoering is de Linux FreeS/WAN IPSEC. Hun webpagina verklaart:

> "Met deze diensten kun je veilige tunnels door niet vertrouwde netwerken bouwen. Alles wat door het niet vertrouwde net gaat wordt gecodeerd door de IPSEC gateway machine en gedecodeerd door de gateway aan de andere kant. Het resultaat is Virtual Private Network of VPN. Dit is een netwerk dat effectief privé is, ondanks dat het machines op verschillende sites omvat die zijn verbonden door het onveilige Internet."

Het is beschikbaar om te downloaden vanaf *http://www.xs4all.nl/~freeswan/* en heeft op het moment van dit schrijven juist versie 1.0 bereikt.

Net als bij andere vormen van versleuteling wordt het niet verspreid met de kernel vanwege exportbeperkingen.

6.4 `ssh` (Secure Shell) en `stelnet`

`ssh` en `stelnet` zijn programma's die het mogelijk maken om in te loggen op remote systemen middels een versleutelde verbinding.

`openssh` is een serie programma's die gebruikt wordt als een veilige vervanging voor `rlogin`, `rsh` en `rcp`. Het gebruikt public-key versleuteling zowel om communicatie tussen twee hosts te versleutelen als wel het authenticeren van gebruikers. Het kan gebruikt worden om veilig in te loggen op een remote host of voor het copiëren van gegevens tussen hosts, terwijl "man-in-het-midden aanvallen"(sessie-kaping) en DNS spoofing voorkomen worden. Het voert gegevenscompressie op je verbindingen uit en zorgt voor veilige X11 communicatie tussen hosts.

Er zijn tegenwoordig diverse ssh uitvoeringen. De originele commerciële uitvoering van Data Fellows kan gevonden worden op *http://www.datafellows.com*. De `ssh` homepage kan worden gevonden op *http://www.cs.hut.fi/ssh/*.

De uitstekende Openssh uitvoering is gebaseerd op een vroege versie van ssh van Data Fellows en is geheel bewerkt zodat het geen enkel patent of eigendomsdelen bevat. Het is gratis en onder een BSD licentie. Het kan worden gevonden op: *http://www.openssh.com*.

Er is ook een open source project om ssh van de grond af te re-implementeren, genaamd "psst...". Voor meer informatie zie: *http://www.net.lut.ac.uk/psst/*

Je kunt ook `ssh` gebruiken vanaf je Windows werkstation naar je Linux `ssh` server. Er zijn diverse gratis verkrijgbare Windows client uitvoeringen, inclusief degene op *http://guardian.htu.tuwien.ac.at/therapy/ssh/* als ook een commerciële uitvoering van Data Fellows *http://www.datafellows.com*.

SSLeay is een gratis uitvoering van Netscape's Secure Sockets Layer protocol, ontwikkeld door Eric Young. Het bevat diverse applicaties zoals een veilig telnet, een module voor Apache, diverse databases en ook diverse algoritmes inclusief DES, IDEA en Blowfish.

Door gebruik te maken van deze library werd een veilige telnet vervanging gecreëerd die versleutelt over een telnet-verbinding. In tegenstelling tot SSH gebruikt stelnet SSL, het Secure Sockets Layer protocol, ontwikkeld door Netscape. Je kunt Secure telnet en Secure FTP vinden door te beginnen met de SSLeay FAQ, beschikbaar op *http://www.psy.uq.oz.au/~ftp/Crypto/*.

SRP is een andere veilige telnet/ftp implementatie. Uit hun webpagina:

> "Het SRP project ontwikkelt veilige Internet software voor gratis wereldwijd gebruik. Beginnend met een volledig veilig Telnet en FTP distributie, hopen we de zwakke netwerk authenticatie systemen te verdringen met sterke vervangers die gebruikersvriendelijkheid niet opofferen voor beveiliging. Beveiliging moet standaard zijn, geen optie!"

Ga voor meer informatie naar *http://srp.stanford.edu/srp*.

6.5 PAM - Pluggable Authentication Modules

Recente versies van de Red Hat Linux distributie komen met een uniform authenticatie schema, genaamd "PAM". PAM maakt het mogelijk dat je je authenticatiemethoden en vereisten in één keer verandert en alle lokale authencatiemethoden kort samenvat zonder dat je enige binary's hoeft te recompileren. Configuratie van PAM ligt buiten de strekking van dit document, maar neem zeker eens een kijkje op de PAM website voor meer informatie. *http://www.kernel.org/pub/linux/libs/pam/index.html*.

Gewoon een paar dingen die je met PAM kunt doen:

- Gebruik versleuteling anders dan DES voor je wachtwoorden. (Het maakt het moeilijker ze met brute kracht te decoderen.)
- Stel resource limits voor al je gebruikers in, zodat ze geen denial-of-service aanvallen kunnen uitvoeren (hoeveelheid processen, hoeveelheid geheugen enz.)
- Schakel shadow passwords (zie hieronder) in één keer in.
- Zorg ervoor dat bepaalde gebruikers alleen kunnen inloggen op bepaalde tijden vanaf bepaalde plaatsen.

Binnen een paar uur, nodig voor het installeren en configureren van je systeem, kun je veel aanvallen op voorhand voorkomen. Gebruik bijvoorbeeld PAM om het systeemwijd gebruik van `.rhosts` bestanden in home directory's van gebruikers uit te schakelen door het toevoegen van deze regels aan `/etc/pam.d/rlogin`:

```
#
# Disable rsh/rlogin/rexec for users
#
login auth required pam_rhosts_auth.so no_rhosts
```

6.6 Cryptographic IP Encapsulation (CIPE)

Het primaire doel van deze software is te voorzien in een faciliteit voor een veilige (tegen het afluisteren of aftappen van berichten, inclusief het analyseren van het netwerkverkeer en faked message injection) subnetwerk interconnectie over een onveilig packet netwerk zoals het Internet.

CIPE versleutelt de gegevens op het netwerk niveau. Pakketten die reizen tussen hosts op het netwerk worden versleuteld. Het versleutelingsmechanisme is dichtbij het stuurprogramma geplaatst dat de pakketten verstuurt en ontvangt.

Dit werkt in tegenstelling tot SSH, dat de gegevens per verbinding versleutelt, op het socket niveau. Een logische verbinding tussen programma's die op verschillende hosts uitgevoerd worden wordt versleuteld.

CIPE kan gebruikt worden bij "tunnellingöm een Virtual Private Network aan te maken. Low-level versleuteling heeft het voordeel dat het zo gemaakt kan worden dat het transparant werkt tussen de twee netwerken verbonden in het VPN, zonder enige wijziging aan applicatie-software.

Samengevat uit de CIPE documentatie:

> De IPSEC normen definiëren een set protocollen die gebruikt kunnen worden (onder andere) om versleutelde VPN's te bouwen. IPSEC is echter een zware en gecompliceerde set van protocollen met een heleboel opties. Uitvoeringen van de volledige set protocollen worden nog zelden gebruikt en sommige geschilpunten (zoals sleutelbeheer) zijn nog niet volledig opgelost. CIPE gebruikt een eenvoudigere benadering, waarbij veel dingen die in parameters vastgelegd kunnen worden (zoals de keuze van het actuele versleutelingsalgoritme wat gebruikt wordt) een keuze is die tijdens de installatie gemaakt moet worden. Dit beperkt de flexibiliteit, maar zorgt voor een eenvoudige (en derhalve efficiënte, makkelijk te debuggen...) uitvoering.

Verdere informatie kan worden gevonden op *http://www.inka.de/~bigred/devel/cipe.html*

Net als met andere vormen van versleuteling wordt het niet standaard met de kernel meegeleverd wegens exportbeperkingen.

6.7 Kerberos

Kerberos is een authenticatiesysteem, ontwikkeld door het Athena Project op MIT. Als een gebruiker inlogt, authenticeert Kerberos deze gebruiker (gebruik makend van een wachtwoord) en voorziet de gebruiker in een manier om zijn identiteit te bewijzen aan andere servers en hosts verspreid over het netwerk.

Deze authenticatie wordt dan gebruikt door programma's zoals `rlogin` om het de gebruiker mogelijk te maken op andere hosts in te loggen zonder wachtwoord (in plaats van het `.rhosts` bestand). Deze authenticatiemethode kan ook gebruikt worden door het mailsysteem om te garanderen dat de mail is aangeleverd bij de juiste persoon, evenals het garanderen dat de afzender is wie hij beweert te zijn.

Kerberos en de andere programma's die erbij geleverd worden, voorkomen dat gebruikers het systeem "spoofenën zodoende laten geloven dat ze iemand anders zijn. Helaas is het installeren van Kerberos erg indringend, omdat het het aanpassen of vervangen van vele standaard programma's vereist.

Je kunt meer informatie over Kerberos vinden door te kijken op *the kerberos FAQ* en de code kan worden gevonden op *http://nii.isi.edu/info/kerberos/*.

[Van: Stein, Jennifer G., Clifford Neuman en Jeffrey L. Schiller. "Kerberos: Een Authenticatie Service voor Open Netwerk Systemen."USENIX Conference Proceedings, Dallas, Texas, Winter 1998.]

Kerberos zou niet je eerste stap moeten zijn in het verbeteren van de beveiliging van je host. Het is nogal ingewikkeld en wordt niet zo veel gebruikt als bijvoorbeeld SSH.

6.8 Shadow Passwords

Shadow passwords is een manier om je gecodeerde wachtwoordinformatie geheim te houden voor normale gebruikers. Recente versies van zowel Red Hat als Debian Linux maken standaard gebruik van shadow passwords, maar op andere systemen worden gecodeerde wachtwoorden opgeslagen in het `/etc/passwd` bestand, dat iedereen kan lezen. Iedereen kan hier vervolgens programma's op loslaten die wachtwoorden kunnen raden en zodoende proberen vast te stellen wat ze zijn. Shadow passwords daarentegen worden opgeslagen in `/etc/shadow`, dat alleen door bevoegde gebruikers kan worden gelezen. Om shadow passwords te kunnen gebruiken, moet je je ervan vergewissen dat al je voorzieningen die toegang nodig hebben tot wachtwoordinformatie opnieuw zijn gecompileerd om het te ondersteunen. PAM (hierboven) biedt je de mogelijkheid om simpelweg een shadow module te plaatsen; het vereist geen hercompilatie van executables. Je kunt een beroep doen op de Shadow-Password HOWTO voor meer informatie als dat nodig is. Het is beschikbaar op *http://metalab.unc.edu/LDP/HOWTO/Shadow-Password-HOWTO.html* Het is nu behoorlijk gedateerd en niet nodig voor distributies die PAM ondersteunen.

6.9 "Crackën "John the Ripper"

Als om wat voor reden dan ook je `passwd` programma geen moeilijk te raden wachtwoorden afdwingt, zul je wellicht een programma willen uitvoeren dat wachtwoorden kraakt om je zodoende ervan te kunnen vergewissen dat de wachtwoorden van je gebruikers veilig zijn.

Programma's die wachtwoorden kraken zijn gebaseerd op een simpel idee: ze proberen elk woord in het woordenboek en vervolgens variaties op deze woorden, coderen ze allemaal en vergelijken ze met je gecodeerde wachtwoord. Als dit tot een treffer leidt, weten ze wat je wachtwoord is.

Er zijn een aantal van deze programma's in omloop ... waarvan de twee meest opvallende "Crackën "John the Ripper"zijn (*http://www.false.com/security/john/index.html*). Ze nemen een hoop van je CPU tijd in beslag, maar je zou wel moeten kunnen vertellen of een aanvaller binnen zou kunnen komen middels deze programma's, door ze eerst zelf uit te voeren en gebruikers met zwakke wachtwoorden op de hoogte te stellen. Houd in de gaten dat een aanvaller eerst een ander lek moet gebruiken om je `/etc/passwd` bestand te kunnen lezen, maar zulke lekken komen vaker voor dan je denkt.

Omdat beveiliging slechts zo krachtig is als je meest onveilige host, is het de moeite waard te vermelden dat als je enige Windows machines op je netwerk hebt, je eens L0phtCrack, een Crack implementatie voor Windows, zou moeten proberen. Het is beschikbaar op *http://www.l0pht.com*

6.10 CFS - Cryptographic File System en TCFS - Transparent Cryptographic File System

CFS is een manier om hele directorystructuren te coderen en gebruikers de mogelijkheid te geven om gecodeerde bestanden hierin op te slaan. Het maakt gebruik van een NFS server die draait op de lokale machine. RPM's zijn beschikbaar op *http://www.zedz.net/redhat/* en meer informatie over hoe het werkt staat op *ftp://ftp.research.att.com/dist/mab/*.

TCFS overtreft CFS doordat er meer integratie met het bestandssysteem is toegevoegd, zodat het duidelijker voor de gebruikers is dat het bestandssysteem gecodeerd is. Meer informatie staat op *http://edu-gw.dia.unisa.it/tcfs/*.

Het hoeft ook niet gebruikt te worden op gehele bestandssystemen, het werkt ook op directorystructuren.

6.11 X11, SVGA en beeldschermbeveiliging

6.11.1 X11

Het is belangrijk dat je je grafische beeldscherm beveiligt om te voorkomen dat aanvallers je wachtwoord inpikken terwijl je het intypt, documenten of andere informatie die op je scherm staat lezen of zelfs een lek gebruiken om root toegang te verkrijgen. Het uitvoeren van remote X applicaties over het netwerk kan ook vol gevaar zijn, doordat het snuffelaars mogelijk wordt gemaakt om al je interactie met het remote systeem te zien.

X heeft een aantal manieren voor toegangsbeheer. De eenvoudigste van deze is host-gebaseerd: je gebruikt `xhost` om alle hosts die toegang mogen hebben tot je beeldscherm te specificeren. Dit is absoluut niet veilig, want als iemand toegang heeft tot je machine, kan het commando `xhost +` *`hun machine`* gegeven worden en men komt gemakkelijk binnen. Bovendien, als je toegang vanaf een niet vertrouwde machine toe moet staan, kan iedereen daar je beeldscherm compromitteren.

Als je `xdm` (X Display Manager) gebruikt om in te loggen, krijg je een veel betere toegangsmethode: MIT-MAGIC-COOKIE-1. Een 128-bit "cookie"wordt gegenereerd en opgeslagen in je `.Xauthority` bestand. Als je een remote machine toegang moet verschaffen tot je beeldscherm, kun je het `xauth` commando en de informatie in je `.Xauthority` bestand gebruiken om toegang te verschaffen voor alleen die verbinding. Zie de Remote-X-Apps mini-HOWTO, beschikbaar op *http://metalab.unc.edu/LDP/HOWTO/mini/Remote-X-Apps.html*.

Je kunt ook `ssh` gebruiken (zie 6.4 (), hierboven) om veilige X verbindingen mogelijk te maken. Dit heeft als voordeel dat het ook duidelijk is voor de eindgebruiker en houdt in dat er geen ongecodeerde gegevens over het netwerk verzonden worden.

Bekijk de `Xsecurity` man pagina voor meer informatie over X beveiliging. De meest veilige gok is het gebruik van `xdm` om in te loggen op je console en vervolgens `ssh` te gebruiken om naar remote sites te gaan waarop je X programma's wilt uitvoeren.

6.11.2 SVGA

SVGAlib programma's zijn kenmerkend SUID-root, ten einde al de video hardware van je Linux machines te kunnen benaderen. Dit maakt ze erg gevaarijk. Als ze crashen, zul je kenmerkend je machine opnieuw moeten opstarten om een bruikbare console terug te krijgen. Let er op dat elk SVGA programma dat je uitvoert authentiek en op z'n minst enigszins vertrouwd is. Nog beter, voer ze helemaal niet uit.

6.11.3 GGI (Generic Graphics Interface project)

Het Linux GGI project probeert verschillende van de problemen met video interfaces onder Linux op te lossen. GGI zal een klein stukje van de video code naar de Linux kernel verplaatsen en vervolgens de toegang tot het videosysteem beheren. Dit betekent dat GGI in staat is om op elk moment je console in een bekende, goede staat te herstellen. Het voorziet ook in een beveiligings-waarschuwingssleutel, zodat je zeker weet dat er geen Trojan horse `login` programma op je console draait.*http://synergy.caltech.edu/˜ggi/*

7 Beveiliging van de kernel

Dit is een beschrijving van de opties voor het configureren van de kernel die met beveiliging te maken hebben, een beschrijving van wat ze doen en hoe je ze moet gebruiken.

Omdat de kernel het gebruik van je computer op het netwerk beheert, is het belangrijk dat deze erg veilig is en niet in gevaar gebracht kan worden. Om enkele van de meest recente netwerkaanvallen te voorkomen, moet je proberen om je kernelversie actueel te houden. Je kunt nieuwe kernels vinden op `<ftp://ftp.kernel.org>` of bij de leverancier van je distributie.

Er is ook een internationale groep die een afzonderlijke uniforme coderingspatch verschaft voor de conventionele Linux kernel. Deze patch verschaft ondersteuning voor een aantal cryptografische subsystemen en zaken die niet kunnen worden toegevoegd aan de conventionele kernel vanwege exportbeperkingen. Voor meer informatie kun je hun webpagina bezoeken op: *http://www.kerneli.org*

7.1 Opties om 2.0 kernels te compileren

De volgende opties zijn voor 2.0.x kernels van toepassing. Je zou deze opties moeten kunnen zien tijdens het configuratieproces van de kernel. Veel van de opmerkingen hier komen uit `./linux/Documentation/Configure.help`. Dit is hetzelfde document als waarnaar verwezen wordt wanneer de Help faciliteit aangeroepen wordt tijdens de `make config` fase van het compileren van de kernel.

- Netwerk Firewalls (CONFIG_FIREWALL)

  Deze optie moet ingeschakeld zijn als je van plan bent om enige firewalling of masquerading op je Linux machine uit te voeren. Als het slechts gaat om een gewone client machine, is het veilig deze optie niet in te schakelen.

- IP: forwarding/gatewaying (CONFIG_IP_FORWARD)

  Als je IP forwarding inschakelt, wordt je Linux box wezenlijk een router. Als je machine aangesloten is op een netwerk, kun je gegevens doorsturen van het ene netwerk naar het andere en wellicht een firewall ondermijnen die daar was geplaatst om dit te voorkomen. Gewone dial-up gebruikers zullen dit uit willen schakelen en andere gebruikers moeten zich concentreren op de beveiligingsimplicaties als ze dit doen. Firewall machines zullen dit ingeschakeld willen hebben en samen met firewall software gebruiken.

  Je kunt IP forwarding dynamisch inschakelen door gebruik te maken van het volgende commando:

  ```
  root#  echo 1 > /proc/sys/net/ipv4/ip_forward
  ```

  en het uitschakelen met het commando:

  ```
  root#  echo 0 >/proc/sys/net/ipv4/ip_forward
  ```

  Houd in gedachten dat de bestanden in /proc "virtuele"bestanden zijn en de getoonde omvang van het bestand wellicht niet overeenkomt met de gegevensuitvoer hiervan.

- IP: syn cookies (CONFIG_SYN_COOKIES)

  Een "SYN Attackïs een "denial of service"(DoS) aanval die alle hulpbronnen op je machine verbruikt en je dwingt om opnieuw op te starten. We kunnen geen reden bedenken waarom je dit niet gewoon inschakelt. In de 2.2.x kernel series staat deze configuratieoptie alleen syn cookies toe, maar schakelt ze niet in. Om ze in te schakelen, moet je het volgende commando geven:

  ```
  root# echo 1 >
  /proc/sys/net/ipv4/tcp_syncookies
  ```

- IP: Firewalling (CONFIG_IP_FIREWALL)

Deze optie is noodzakelijk als je je machine gaat configureren als een firewall, aan masquerading gaat doen of je dial-up werkstation wil beschermen tegen het binnendringen van iemand via je PPP dial-up interface.

- IP: firewall packet logging (CONFIG_IP_FIREWALL_VERBOSE)

  Deze optie geeft je informatie over de pakketten die je firewall ontvangt, zoals afzender, ontvanger, poort enzovoorts.

- IP: Drop source routed frames (CONFIG_IP_NOSR)

  Deze optie moet ingeschakeld worden. Source routed frames bevatten het gehele pad tot hun bestemming binnenin het pakket. Dit betekent dat de routers waar het pakket doorheen gaat het niet hoeven te inspecteren en het gewoon doorsturen. Dit kan ertoe leiden dat er gegevens je systeem binnenkomen die een potentieel beveilingslek kunnen zijn.

- IP: masquerading (CONFIG_IP_MASQUERADE)

  Als een van de computers op je lokale netwerk, waarvoor je Linux box als een firewall optreedt, iets wil versturen naar buiten, kan jouw box zich vermommen (masquerade) als die host. Dat wil zeggen, het stuurt het verkeer door naar de bedoelde bestemming, maar laat het eruitzien alsof het komt van de firewall box zelf. Zie *http://www.indyramp.com/masq* voor meer informatie.

- IP: ICMP masquerading (CONFIG_IP_MASQUERADE_ICMP)

  Deze optie voegt ICMP masquerading toe aan de vorige optie, waarbij alleen masquerading van TCP of UDP verkeer plaatvindt.

- IP: transparent proxy support (CONFIG_IP_TRANSPARENT_PROXY)

  Hiermee kan je Linux firewall elk netwerkverkeer dat van het lokale netwerk afkomstig is en bestemd is voor een remote host transparant omleiden naar een lokale server, genaamd een "transparent proxy server". Dit zorgt ervoor dat de lokale computers denken dat ze praten tegen het remote eind, terwijl ze in feite verbonden zijn met de lokale proxy. Zie de IP-Masquerading HOWTO en *http://www.indyramp.com/masq* voor meer informatie.

- IP: always defragment (CONFIG_IP_ALWAYS_DEFRAG)

Gewoonlijk is deze optie uitgeschakeld, maar als je een firewall of een masquerading host aan het bouwen bent, zul je dit in willen schakelen. Wanneer gegevens van de ene host naar de andere worden gestuurd, wordt het niet altijd verstuurd als een enkel gegevenspakket, maar eerder wordt het gefragmenteerd in verschillende stukken. Het probleem hierbij is dat de poortnummers alleen in het eerste fragment opgeslagen zijn. Dit betekent dat iemand informatie aan de overblijvende pakketten kan toevoegen die daar niet hoort te zijn. Het kan ook een "teardrop attack"voorkomen tegen een interne host die hier zelf nog niet tegen gepatched is.

- Packet Signatures (CONFIG_NCPFS_PACKET_SIGNING)

  Dit is een optie, beschikbaar in de 2.2.x kernel series, die NCP pakketten zal voorzien van een kenmerk voor een betere beveiliging. Normaal kun je het uit laten staan, maar het is er als je het toch nodig hebt.

- IP: Firewall packet netlink device (CONFIG_IP_FIREWALL_NETLINK)

  Dit is een erg aardige optie waarmee je de eerste 128 bytes van de pakketten van een user-space programma kan analyseren, om zodoende te bepalen of je, gebaseerd op zijn deugdelijkheid, het pakket wilt accepteren of afwijzen.

7.2 Opties om 2.2 kernels te compileren

Voor 2.2.x kernels zijn veel van de opties hetzelfde, maar er zijn ook een paar nieuwe ontwikkeld. Veel van de opmerkingen hier komen van `./linux/Documentation/Configure.help`, wat hetzelfde document is waarnaar verwezen wordt als je de Help faciliteit gebruikt tijdens de `make config` fase bij het compileren van de kernel. Alleen de nieuw toegevoegde opties worden hieronder opgesomd. Raadpleeg de 2.0 beschrijving voor een lijst met andere noodzakelijke opties. De meest opmerkelijke verandering in de 2.2 kernel series is de IP firewalling code. Het `ipchains` programma wordt nu gebruikt om IP firewalling te installeren, in plaats van het `ipfwadm` programma dat gebruikt werd in de 2.0 kernel.

- Socket Filtering (CONFIG_FILTER)

  Voor de meeste mensen is het veilig om nee te zeggen tegen deze optie. Met deze optie kun je een userspace filter verbinden met elke socket en vaststellen of pakketten moeten worden toegestaan of afgewezen. Zeg nee, tenzij je een erg specifieke noodzaak hebt en in staat bent om zo'n filter te programmeren. Houd ook in de gaten dat ten tijde van dit schrijven alle protocollen behalve TCP ondersteund werden.

- Port Forwarding

Port Forwarding is een aanvulling op Masquerading, dat enige doorzending van pakketten van buiten naar binnenin een firewall op vastgestelde poorten toestaat. Dit kan nuttig zijn als je, bijvoorbeeld, een webserver achter een firewall of masquerading host wil draaien en die webserver toegankelijk moet zijn vanaf de buitenwereld. Een externe client stuurt een verzoek naar poort 80 van de firewall, de firewall stuurt dit verzoek door naar de webserver, de webserver behandelt het verzoek en de resultaten worden via de firewall naar de originele client verstuurd. De client denkt dat de firewall machine zelf de webserver draait. Dit kan ook gebruikt worden voor het verdelen van de belasting als je een heleboel identieke webservers achter de firewall hebt. Informatie over deze voorziening is beschikbaar op http://www.monmouth.demon.co.uk/ipsubs/portforwarding.html (om op het WWW te surfen heb je toegang tot een machine op het Internet nodig, die een programma als lynx of Netscape heeft). Voor algemene informatie kun je kijken op ftp://ftp.compsoc.net/users/steve/ipportfw/linux21/

- Socket Filtering (CONFIG_FILTER)

  Bij gebruik van deze optie kunnen user-space programma's een filter aan elke socket verbinden en daardoor de kernel vertellen dat bepaalde soorten gegevens wel of niet door de socket mogen gaan. Linux socket filtering werkt voorlopig op alle soorten sockets behalve TCP. Zie het tekstbestand `./linux/Documentation/networking/filter.txt` voor meer informatie.

- IP: Masquerading

  De 2.2 kernel masquerading is verbeterd. Het verschaft aanvullende ondersteuning voor het vermommen (masquerading) van speciale protocollen etc. Lees de IP Chains HOWTO voor meer informatie.

7.3 Kernel Devices

Er zijn een paar block en character devices beschikbaar onder Linux die je ook behulpzaam kunnen zijn met de beveiliging.

De twee devices `/dev/random` en `/dev/urandom` worden verschaft door de kernel om op elk tijdstip te kunnen voorzien in willekeurige gegevens.

Zowel `/dev/random` als `/dev/urandom` zouden veilig genoeg moeten zijn om te gebruiken voor het genereren van PGP sleutels, het aanroepen van `ssh` en andere applicaties waar veilige, willekeurige nummers vereist zijn. Voor aanvallers moet het onmogelijk zijn het volgende nummer te voorspellen gezien elke aanvangsvolgorde van nummers van deze bronnen. Er is een hoop moeite voor gedaan om zeker te stellen dat de nummers die je van deze bronnen krijgt, willekeurig in elke betekenis van het woord zijn.

Het enige verschil tussen de twee devices, is dat `/dev/random` door zijn voorraad willekeurige bytes heen raakt en je laat wachten tot er weer een voorraad is aangemaakt. Merk op dat het op sommige systemen een blokkade voor lange tijd kan opwerpen doordat er gewacht wordt op het invoegen van nieuwe door de gebruiker gegenereerde entropie in het systeem. Je moet voorzichtigheid betrachten voordat je `/dev/random` gebruikt. (Wellicht is het beste wat je kunt doen, het te gebruiken wanneer je gevoelige sleutelinformatie aan het genereren bent en je de gebruiker vertelt herhaaldelijk op het toetsenbord te slaan totdat je de melding "OK, genoeg"geeft).

`/dev/random` is hoge kwaliteit entropy, gegenereerd door het meten van de interrupt tijden etc. Het blokkeert totdat er voldoende bits met willekeurige gegevens beschikbaar zijn.

/dev/urandom is vergelijkbaar, maar wanneer de opslag van entropie op een laag pitje staat, zal het een cryptografisch sterk mengelmoesje van wat er is terugsturen. Dit is niet zo veilig, maar het is voldoende voor de meeste applicaties.

Je kunt de devices raadplegen door iets dergelijks te gebruiken:

```
root#  head -c 6 /dev/urandom | mimencode
```

Dit zal acht regels willekeurige karakters op de console afdrukken, geschikt voor wachtwoordgeneratie. Je kunt `mimencode` in het `metamail` package vinden.

Zie /usr/src/linux/drivers/char/random.c voor een beschrijving van het algoritme.

Met dank aan Theodore Y. Ts'o, Jon Lewis en anderen van Linux-kernel die mij (Dave) hiermee geholpen hebben.

8 Beveiliging van het netwerk

Netwerkbeveiliging wordt steeds belangrijker omdat mensen steeds langer verbonden zijn met het netwerk. De beveiliging van het netwerk in gevaar brengen is vaak veel eenvoudiger dan het in gevaar brengen van de fysieke of lokale beveiliging en wordt steeds alledaagser.

Er zijn een aantal goede tools die hulp bieden bij netwerkbeveiliging en steeds meer daarvan worden geleverd bij Linux distributies.

8.1 Packet Sniffers

Een van de meest gebruikte manieren waarop indringers zich toegang tot meer systemen op je netwerk verschaffen is door het gebruik van een "packet snifferöp een reeds in gevaar gebrachte host. Deze "sniffer"luistert op de Ethernet poort slechts naar dingen als `passwd`, `login` en `su` in de pakkettenstroom en logt dan het verkeer dat volgt. Op deze manier verkrijgen aanvallers wachtwoorden voor systemen waarop ze niet eens probeerden in te breken. Wachtwoorden die uit platte tekst bestaan zijn erg kwetsbaar voor deze aanval.

Voorbeeld: Host A is gecompromitteerd. Aanvaller installeert een sniffer. Sniffer pikt een beheerder logging op naar host B, afkomstig van host C. Het verkrijgt het persoonlijke wachtwoord van de beheerder zodra er ingelogd wordt op B. Dan doet de beheerder een `su` om een probleem op te lossen. Ze hebben nu het root wachtwoord voor host B. Later laat de beheerder iemand `telnet` uitvoeren vanaf zijn account naar host Z op een andere site. Nu heeft de aanvaller een wachtwoord/login op host Z.

Tegenwoordig hoeft de aanvaller niet eens meer een systeem te compromitteren om dit te doen: ze kunnen ook een laptop of pc het gebouw binnenbrengen en het systeem aftappen.

Het gebruik van `ssh` of andere methoden om wachtwoorden te coderen dwarsboomt deze aanval. Zaken als APOP voor POP accounts kunnen deze aanval ook voorkomen. (Normale POP logins zijn hier erg kwetsbaar voor, zoals alles dat platte-tekst wachtwoorden over het netwerk verstuurt).

8.2 Systeemdiensten en tcp-wrappers

Het eerste waar je naar moet kijken, voordat je je Linux systeem op *ENIG* netwerk aansluit, is wat voor diensten je moet bieden. Diensten die je niet hoeft te bieden moeten uitgeschakeld worden, zodat je één ding minder hebt om je zorgen over te maken en aanvallers een plek minder hebben om te zoeken naar een lek.

Er zijn een aantal manieren om diensten onder Linux uit te schakelen. Je kunt kijken naar je `/etc/inetd.conf` bestand om te zien welke diensten worden aangeboden door je `inetd`. Schakel degenen die je niet nodig hebt uit door een `#` aan het begin van de regel te plaatsen en vervolgens je inetd proces een SIGHUP te sturen.

Je kunt ook diensten uit je `/etc/services` bestand verwijderen (of een `#` aan het begin van de regel plaatsen). Dit heeft tot gevolg dat lokale clients de dienst ook niet kunnen vinden (als je bijvoorbeeld `ftp` verwijdert en probeert te ftp-en naar een remote site vanaf die machine, zal dat mislukken en de boodschap ünknown service"zal getoond worden). Het is meestal de moeite niet waard om diensten te verwijderen uit `/etc/services`, omdat het geen aanvullende beveiliging verschaft. Als een lokale persoon `ftp` zou willen gebruiken ondanks het feit dat je een `#` aan het begin van de regel hebt geplaatst, maken ze hun eigen client aan die de gebruikelijke FTP poort gebruikt en nog prima werkt.

Enkele diensten die je wellicht ingeschakeld zou willen laten zijn:

- `ftp`
- `telnet` (of `ssh`)
- mail, zoals `pop-3` of `imap`
- `identd`

Als je weet dat je een bepaald pakket niet gaat gebruiken, kun je het ook helemaal verwijderen. `rpm -e` ***`naam van het pakket`*** onder de Red Hat distributie zal het gehele pakket verwijderen. Onder Debian doet `dpkg -remove` hetzelfde.

Bovendien moet je echt de rsh/rlogin/rcp utility's uitschakelen en tevens voorkomen dat login (gebruikt door `rlogin`), shell (gebruikt door `rcp`) en exec (gebruikt door `rsh`) worden gestart in `/etc/inetd.conf`. Deze protocollen zijn extreem onveilig en zijn in het verleden het doel geweest van misbruik.

Je moet `/etc/rc.d/rc[0-9].d` (op Red Hat; `/etc/rc[0-9].d` op Debian) controleren om te zien of er servers in deze directory's gestart worden die niet nodig zijn. De bestanden in deze directory's zijn eigenlijk symbolische links naar bestanden in de directory `/etc/rc.d/init.d` (op Red Hat; `/etc/init.d` op Debian). Het hernoemen van de bestanden in de `init.d` directory schakelt alle symbolische links uit die naar dat bestand verwijzen. Als je een dienst slechts voor een bepaald run level uit wilt schakelen, hernoem dan de desbetreffende symbolische link door de hoofdletter `S` te vervangen door een kleine letter `s`, zoals dit:

```
root#  cd /etc/rc6.d
root#  mv S45dhcpd s45dhcpd
```

Als je `rc` bestanden in BSD-stijl hebt, moet je `/etc/rc*` controleren op programma's die je niet nodig hebt.

Bij de meeste Linux distributies worden tcp_wrappers geleverd die al je TCP diensten "wrappen". Een tcp_wrapper (`tcpd`) wordt aangeroepen door `inetd` in plaats van de echte server. `tcpd` controleert dan de host die om de dienst verzoekt en start ofwel de echte server op of weigert toegang vanaf die host. Met `tcpd` kun je de toegang tot je TCP diensten beperken. Je moet een `/etc/hosts.allow` aanmaken en alleen toevoegen in de hosts die toegang tot de diensten van je machine nodig hebben.

Als je een dialup thuisgebruiker bent, stellen we voor dat je ze ALLEMAAL weigert. `tcpd` logt ook mislukte pogingen om toegang tot diensten te krijgen, dus dit kan je waarschuwen als je aangevallen wordt. Als je nieuwe diensten toevoegt, moet je je ervan overtuigen dat je ze zo configureert dat ze tcp-verbindingen gebruiken als ze zijn gebaseerd op TCP. Een normale dialup gebruiker kan bijvoorbeeld voorkomen dat buitenstaanders verbinding maken met zijn machine en toch de mogelijkheid hebben om post te ontvangen en netwerkverbindingen naar het Internet te maken. Om dit te doen, moet je het volgende aan je `/etc/hosts.allow` toevoegen:

ALL: 127.

En natuurlijk bevat /etc/hosts.deny:

ALL: ALL

wat externe verbindingen naar je machine voorkomt en je toch toestaat om van binnenuit een verbinding te maken met servers op het Internet.

Houd in gedachten dat tcp_wrappers alleen diensten die uitgevoerd worden door `inetd` beschermt en een bepaald aantal anderen. Het is heel goed mogelijk dat er ook andere diensten op je machine draaien. Je kunt `netstat -ta` gebruiken om een lijst te zoeken met alle diensten die je machine aanbiedt.

8.3 Verifieer je DNS informatie

Het up-to-date houden van DNS informatie van alle hosts op je netwerk kan helpen op de beveiliging te vergroten. Als een ongeautoriseerde host verbinding krijgt met je netwerk, kun je dat herkennen door het ontbreken van een DNS entry. Veel diensten kunnen zo worden ingesteld dat ze geen verbindingen toestaan van hosts die geen geldige DNS entry hebben.

8.4 `identd`

`identd` is een klein programma dat als kenmerk heeft dat het buiten je `inetd` server om draait. Het houdt bij welke gebruiker welke TCP dienst uitvoert en rapporteert dit vervolgens aan een ieder die hierom verzoekt.

Veel mensen onderschatten de bruikbaarheid van `identd` en schakelen het dus uit of blokkeren alle verzoeken van de site hiervoor. `identd` is er niet om remote sites van dienst te zijn. Er is geen enkele manier om erachter te komen of de gegevens die je ontvangt van de remote `identd` al dan niet correct zijn. Er vindt geen verificatie plaats van `identd` verzoeken.

Waarom zou je het dan willen draaien? Omdat het *jou* van nut is en het extra gegevens kan opleveren als je iemand moet traceren. Als je `identd` niet is gecompromitteerd, weet je dat het remote sites de gebruikersnaam of gebruikers-id van mensen die TCP diensten gebruiken vertelt. Als de beheerder op een remote site je aanspreekt en je vertelt dat gebruiker die-en-die heeft geprobeerd om hun site te "hacken", kun je gemakkelijk actie ondernemen tegen die gebruiker. Als je geen `identd` draait, zul je een heleboel logs moeten bekijken om uit te vinden wie er op dat moment online was en normaal gesproken veel meer tijd nodig hebben om die gebruiker op te sporen.

Aan de `identd` die geleverd wordt bij de meeste distributies is meer in te stellen dan de meeste mensen denken. Je kunt het voor bepaalde gebruikers uitschakelen (ze kunnen een `.noident` bestand aanmaken), je kunt alle `identd` verzoeken loggen (we raden dit aan), je kunt zelfs identd een uid in plaats van een gebruikersnaam of zelfs NO-USER laten terugsturen.

8.5 SATAN, ISS en andere netwerkscanners

Er zijn een aantal verschillende softwarepakketten die doen aan poort- en dienst-gebaseerd scannen van machines of netwerken. SATAN, ISS, SAINT en Nessus zijn enkele van de meer bekende. Deze software maakt verbinding met de doelmachine (of al de doelmachines op een netwerk) op alle mogelijke poorten en probeert uit te vinden welke dienst daar draait. Gebaseerd op deze informatie kun je zeggen of de machine kwetsbaar is voor een bepaald soort misbruik op die server.

SATAN (Security Administrator's Tool for Analyzing Networks) is een poortscanner met een webinterface. Het kan ingesteld worden om lichte, medium of zware controles op een machine of een computernetwerk uit te voeren. Het is een goed idee om SATAN te downloaden en je machine of netwerk te scannen en de problemen die het vindt op te lossen. Download SATAN vanaf *metalab* `<http://metalab.unc.edu/pub/packages/security/Satan-for-Linux/>` of vanaf een goed bekend staande FTP of website. Er is een trojaanse copie van SATAN verspreid over het net: *http://www.trouble.org/˜zen/satan/satan.html*. Merk op dat SATAN al geruime tijd niet meer is bijgewerkt en enkele van de andere tools hieronder wellicht beter werken.

ISS (Internet Security Scanner) is een andere poort-gebaseerde scanner. Het is sneller dan SATAN en dus wellicht beter voor grotere netwerken. Echter, SATAN heeft de eigenschap dat het meer informatie verschaft.

Abacus bestaat uit een set tools om te voorzien in host-gebaseerde beveiliging en inbraakdetectie. Neem een kijkje op de homepage op het web voor meer informatie. *http://www.psionic.com/abacus/*

SAINT is een bijgewerkte versie van SATAN. Het is web-gebaseerd en heeft veel meer up-to-date tests dan SATAN. Je kunt hier meer over te weten komen op: *http://www.wwdsi.com/˜saint*

Nessus is een gratis beveiligingsscanner. Het is gemakkelijk in gebruik dankzij een GTK grafische interface. Het is ook uitgerust met een erg aardige plugin setup voor nieuwe poort-scan testen. Kijk voor meer informatie op: *http://www.nessus.org*

8.5.1 Poortscans detecteren

Er zijn enkele tools ontworpen om je te waarschuwen voor poortscans door SATAN, ISS en andere scansoftware. Echter, als je tcp_wrappers ruimdenkend gebruikt en regelmatig je logbestanden nakijkt, zullen je deze poortscans wel opvallen. Zelfs met de meest minimale instelling laat SATAN sporen na in de logs op een Red Hat systeem.

Er zijn ook "stealth"poort scanners. Een pakket waarop het TCP ACK bit ingesteld is (zoals dat gedaan wordt wanneer de verbindingen tot stand zijn gebracht) zal waarschijnlijk wel door een firewall komen die pakketten filtert. Het RST pakket dat teruggestuurd wordt vanaf een poort die _*geen tot stand gebrachte sessie had*_ kan worden gezien als bewijs dat er leven is op die poort. Ik denk niet dat tcp_wrappers dit zal detecteren.

8.6 `sendmail`, `qmail` en MTA's

Een van de belangrijkste diensten die je aan kan bieden is een mailserver. Helaas is het ook de meest kwestbare voor aanvallen, wat te wijten is aan het aantal taken dat het uit moet voeren en de privileges die het kenmerkend nodig heeft.

Als je `sendmail` gebruikt, is het erg belangrijk dat je de meest recente versie gebruikt. `sendmail` heeft een heel lange geschiedenis van misbruik op het gebied van beveiliging. Let erop dat je altijd de meest recente versie vanaf *http://www.sendmail.org* gebruikt.

Houd in gedachten dat sendmail niet hoeft te draaien om mail te kunnen versturen. Als je een thuisgebruiker bent, kun je sendmail helemaal uitschakelen en gewoon je mail client gebruiken om mail te versturen. Je kunt er ook voor kiezen om de -bd"flag uit het sendmail startup bestand te verwijderen en daarbij inkomende verzoeken om mail uit te schakelen. Met andere woorden, je kunt sendmail opstarten vanuit je startup script door daarvoor in de plaats het volgende te gebruiken:

```
# /usr/lib/sendmail -q15m
```

Dit zorgt ervoor dat sendmail de berichtenlijst elke vijftien minuten nazoekt op berichten die bij de eerste poging niet met succes konden worden overgebracht.

Veel beheerders kiezen ervoor om sendmail niet te gebruiken en kiezen in plaats daarvan voor een van de andere mailtransport middelen. Je zou kunnen overwegen om over te stappen op `qmail`. `qmail` is vanaf het eerste begin ontworpen met beveiliging in gedachten. Het is snel, stabiel en veilig. Qmail kan worden gevonden op *http://www.qmail.org*

Een directe concurrent van qmail is "postfix", geschreven door Wietse Venema, de auteur van tcp_wrappers en andere beveiligingstools. Vroeger heette het vmailer, werd gesponsord door IBM en is ook een mail-transport middel dat vanaf het eerste begin geschreven is met beveiliging in het achterhoofd. Je kunt meer informatie over postfix vinden op *http://www.postfix.org*

8.7 Denial of Service aanvallen

Een "Denial of Service"(DoS) aanval is er een waar de aanvaller probeert om enkele bronnen zo overbelast te maken dat ze geen rechtmatige verzoeken meer kunnen beantwoorden of rechtmatige gebruikers de toegang tot hun machine ontzeggen.

Denial of service aanvallen zijn in de aflopen jaren sterk in aantal toegenomen. Enkele van de meer populaire en recente zijn hieronder opgesomd. Merk op dat er steeds weer nieuwe verschijnen, dus dit zijn slechts een paar voorbeelden. Lees de Linux security lists, de bugtraq list en archieven voor meer recente informatie.

- **SYN Flooding** - SYN flooding is een denial of service aanval op het netwerk. Het maakt misbruik van een "loopholeïn de manier waarop TCP verbindingen tot stand gekomen zijn. De nieuwere Linux kernels (2.0.30 en hoger) hebben diverse in te stellen opties om te voorkomen dat SYN flood aanvallen mensen de toegang tot hun machine of diensten ontzeggen. Zie 7 (Kernel beveiliging) voor opties om de kernel juist te beveiligen.
- **Pentium "F00F"Bug** - Het is recentelijk ontdekt dat een serie assembleercodes die gestuurd wordt naar een echte Intel Pentium processor de machine opnieuw op zal starten. Dit heeft betrekking op elke machine met een Pentium processor (geen klonen, geen Pentium Pro of PII), ongeacht op welk besturingssysteem het draait. Linux kernels 2.0.32 en hoger bevatten een handigheid om deze bug te omzeilen en zodoende te voorkomen dat het je machine öp slot zet". Kernel 2.0.33 heeft een verbeterde versie van deze kernel fix en wordt aangeraden vanaf versie 2.0.32. Als je draait op een Pentium, moet je deze upgrade nu uitvoeren!
- **Ping Flooding** - Ping flooding is een eenvoudige denial of service aanval met brute kracht. De aanvaller stuurt een stroom (flood) van ICMP pakketten naar je machine. Als ze dit doen vanaf een host met een grotere bandbreedte dan die van jou, zal je machine niet in staat zijn om ook maar iets over het netwerk te versturen. Een variatie op deze aanval, genaamd "smurfing", stuurt ICMP pakketten naar een host met het return IP van *jouw* machine's, hetgeen ze toestaat om je minder traceerbaar te "flooden". Je kunt meer informatie over de "smurfäanval vinden op *http://www.quadrunner.com/˜chuegen/smurf.txt*

  Als je ooit te maken krijgt met een ping flood aanval, gebruik dan een tool als `tcpdump` om te bepalen waar de pakketten vandaan komen (of vandaan schijnen te komen) en neem vervolgens contact op met je provider om deze informatie door te geven. Ping floods kunnen het eenvoudigst tot staan worden gebracht op het router niveau of door gebruik te maken van een firewall.
- **Ping o' Death** - De Ping o' Death aanval stuurt ICMP ECHO REQUEST pakketten die te groot zijn om te passen in de kernel gegevensstructuren die bedoeld zijn om ze op te slaan. Omdat het sturen van een enkel, groot (65,510 bytes) "ping"pakket naar veel systemen erin zal resulteren dat ze "hangenöf zelfs crashen, werd dit probleem al snel de "Ping o' Death"genoemd. Dit probleem is al lang geleden opgelost en is niet langer iets waar je je druk over hoeft te maken.

- **Teardrop / New Tear** - Een van de meest recente misbruiken heeft te maken met een bug die aanwezig is in de IP fragmentatie code op Linux en Windows platformen. Het is opgelost in kernel versie 2.0.33 en het is niet meer nodig om tijdens het compileren van de kernel een optie te selecteren om gebruik te maken van deze oplossing. Linux is blijkbaar niet kwetsbaar voor het "newtear"misbruik.

Je kunt de code van de meeste misbruiken en een meer diepgaande beschrijving van hoe ze werken, vinden op *http://www.rootshell.com* door gebruik te maken van hun zoekmachine.

8.8 NFS (Network File System) beveiliging

NFS is een veelgebruikt protocol om bestanden te delen. Het staat servers toe om `nfsd` en `mountd` te draaien om gehele bestandssystemen te ëxporteren"naar andere machines door gebruik te maken van de ondersteuning van het NFS bestandssysteem dat is ingebouwd in hun kernels (of een andere client ondersteuning als het geen Linux machines zijn). `mountd` houdt de gemounte bestandssystemen bij in `/etc/mtab` en kan ze tonen met `showmount`.

Veel sites gebruiken NFS om gebruikers te voorzien van een home directory, zodat ze, ongeacht vanaf welke machine in het cluster ze inloggen, allemaal hun eigen bestanden hebben.

Er is maar een kleine hoeveelheid beveiliging toegestaan bij het exporteren van bestandssystemen. Je kunt je `nfsd` de remote root gebruiker (uid=0) laten omzetten naar de `nobody` gebruiker, waardoor totale toegang tot de bestanden die worden geëxporteerd wordt ontzegt. Omdat individuele gebruikers echter toegang hebben tot hun eigen bestanden (of op z'n minst met hetzelfde uid), kan de remote root gebruiker inloggen (of `su` gebruiken om in te loggen) op hun account en totale toegang hebben tot hun bestanden. Dit is slechts een kleine hindernis voor een aanvaller die toegang heeft om je remote bestandssystemen te mounten.

Als je NFS moet gebruiken, wees er dan zeker van dat je alleen exporteert naar die machines waarnaar het echt nodig is. Exporteer nooit je gehele root directory; exporteer alleen directory's die je moet exporteren.

Bekijk de NFS HOWTO voor meer informatie over NFS, beschikbaar op *http://metalab.unc.edu/mdw/HOWTO/NFS-HOWTO.html*

8.9 NIS (Network Information Service) (voorheen YP)

Network Information Service (voorheen YP) is een manier waarop informatie verspreid wordt naar een groep machines. De NIS beheerder beheert de informatietabellen en converteert ze naar NIS map bestanden. Deze mappen worden dan op het netwerk gezet, waar ze NIS client machines toestaan om login, wachtwoord, home directory en shell informatie te verkrijgen (alle informatie in een standaard `/etc/passwd` bestand). Dit staat gebruikers toe om eenmalig hun wachtwoord te veranderen, waarna dit zijn uitwerking heeft op alle machines in het NIS domein.

NIS is helemaal niet veilig. Dit was ook nooit de bedoeling. Het was bedoeld om handig en nuttig te zijn. Iedereen die de naam van je NIS domein kan raden (waar dan ook op het net), kan in het bezit komen van een kopie van je passwd bestand en gebruik maken van "crackën "John the Ripperöm de wachtwoorden van je gebruikers te kraken. Ook is het mogelijk om NIS te "spoofenën allerlei soorten nare trucks uit te halen.Als je NIS moet gebruiken, wees je dan bewust van de gevaren.

Er is een veel veiligere vervanging voor NIS, genaamd NIS+. Bekijk de NIS HOWTO voor meer informatie: *http://metalab.unc.edu/mdw/HOWTO/NIS-HOWTO.html*

8.10 Firewalls

Firewalls zijn bedoeld om te controleren welke informatie je lokale netwerk binnenkomt en uitgaat. De firewall host is kenmerkend verbonden met het Internet en je lokale LAN en de enige toegang vanaf je LAN naar het Internet is via de firewall. Op deze manier kan de firewall in de gaten houden wat naar en vanaf het Internet en je lokale LAN wordt gestuurd.

Er zijn een aantal soorten firewalls en manieren om ze op te zetten. Op Linux machines kunnen erg goede firewalls gemaakt worden. Firewall code kan rechtstreeks in 2.0 en hogere kernels ingebouwd worden. De user-space tools `ipfwadm` voor 2.0 kernels en `ipchains` voor 2.2 kernels staan je toe om direct de soorten netwerkverkeer die je toestaat te veranderen. Je kunt ook bepaalde soorten netwerkverkeer loggen.

Firewalls zijn een erg nuttige en belangrijke techniek om je netwerk te beveiligen. Denk echter nooit dat omdat je een firewall hebt, je je machines die erachter hangen niet hoeft te beveiligen. Dit is een fatale fout. Bekijk de erg goede `Firewall-HOWTO` op je meest recente metalab archief voor meer informatie over firewalls en Linux. *http://metalab.unc.edu/mdw/HOWTO/Firewall-HOWTO.html*

Meer informatie kan ook gevonden worden in de IP-Masquerade mini-howto: *http://metalab.unc.edu/mdw/HOWTO/mini/IP-Masquerade.html*

Meer informatie over `ipfwadm` (de tool waarmee je de instellingen voor je firewall kan veranderen) kan worden gevonden op: *http://www.xos.nl/linux/ipfwadm/*

Als je geen ervaring met firewalls hebt en van plan bent er een op te zetten voor meer dan slechts een simpel beveiligingsbeleid, is het "Firewalls book"door O'Reilly and Associates of een ander online firewall document verplicht leesvoer. Bekijk *http://www.ora.com* voor meer informatie. Het National Institute of Standards and Technology heeft een uitstekend document over firewalls samengesteld. Hoewel het stamt uit 1995, is het nog steeds behoorlijk goed. Je kunt het vinden op *http://csrc.nist.gov/nistpubs/800-10/main.html*. Ook interessant:

- The Freefire Project – een lijst van vrij verkrijgbare firewall tools, beschikbaar op *http://sites.inka.de/sites/lina/freefire-l/index_ en.html*
- SunWorld Firewall Design – geschreven door de auteurs van het O'Reilly book. Dit verschaft een globale introductie tot de verschillende soorten firewalls. Het is beschikbaar op *http://www.sunworld.com/swol-01-1996/swol-01-firewall.html*
- Mason - de geautomatiseerde firewall bouwer voor Linux. Dit is een firewall script dat je al doende de dingen leert die je moet doen op je netwerk. Meer informatie op: *http://www.pobox.com/˜wstearns/mason/*

8.11 IP Chains - Linux Kernel 2.2.x Firewalling

Linux IP Firewalling Chains is een update naar de 2.0 Linux firewalling code voor de 2.2 kernel. Het heeft veel meer voorzieningen dan voorgaande uitvoeringen, inclusief:

- Meer flexibele pakketmanipulaties
- Complexere accounting
- Eenvoudige beleidswijzigingen zijn mogelijk op detail-niveau
- Fragmenten kunnen expliciet worden geblokkeerd, geweigerd etc.
- Logt verdachte pakketten

- Kan omgaan met protocollen anders dan ICMP/TCP/UDP.

Als je nu `ipfwadm` op je 2.0 kernel gebruikt, zijn er scripts beschikbaar om het `ipfwadm` commando formaat te converteren naar het formaat dat `ipchains` gebruikt.

Lees de IP Chains HOWTO voor verdere informatie. Het is beschikbaar op *http://www.rustcorp.com/linux/ipchains/HOWTO.html*

8.12 VPN's - Virtual Private Networks

VPN's zijn een manier om een "virtueel"netwerk tot stand te brengen bovenop een reeds bestaand netwerk. Dit virtuele netwerk is vaak gecodeerd en stuurt het verkeer alleen naar en van enkele bekende entiteiten die deel uitmaken van het netwerk. VPN's worden vaak gebruikt om iemand die thuis werkt via het publieke Internet te verbinden met het interne bedrijfsnetwerk.

Als je een Linux masquerading firewall draait en je moet MS PPTP (Microsoft's VPN point-to-point product) pakketten omzeilen, is er een Linux kernel patch uitgekomen om juist dat te doen. Zie: *ip-masq-vpn*.

Er zijn diverse Linux VPN oplossingen beschikbaar:

- vpnd. Zie de *http://sunsite.auc.dk/vpnd/*.
- Free S/Wan, beschikbaar op *http://www.xs4all.nl/~freeswan/*
- ssh kan worden gebruikt om een VPN te construeren. Zie de VPN mini-howto voor meer informatie.
- vps (virtual private server) op *http://www.strongcrypto.com*.

Zie ook de paragraaf over IPSEC voor aanwijzingen en meer informatie.

9 Beveiligingsvoorbereidingen (voordat je on-line gaat)

Ok, dus je hebt je systeem gecontroleerd en bevonden dat het zo veilig mogelijk is en je bent klaar om het on-line te zetten. Er zijn een aantal dingen die je nu moet doen om je voor te bereiden op een aanval, zodat je de indringer snel kan uitschakelen, de zaak herstelt en weer draait.

9.1 Maak een volledige backup van je machine

Een discussie over backupmethodes en opslag valt buiten de strekking van dit document, maar hier zijn een paar woorden over backups en beveiliging:

Als je minder dan 650 mb aan gegevens op een partitie op te slaan hebt, is een kopie van je gegevens op een CD-R een goede manier (omdat het moeilijk is om er later mee te knoeien en het een hele tijd meegaat mits het juist wordt bewaard). Tapes en andere herschrijfbare media moeten gelijk tegen schrijven worden beveiligd zodra je backup klaar is en vervolgens worden geverifieerd om geknoei te voorkomen. Let erop dat je je backups bewaart op een veilige off-line lokatie. Een goede backup verzekert je ervan dat je je systeem kunt herstellen vanaf een bekend goed punt.

9.2 Het kiezen van een goed backupschema

Een cyclus van zes tapes is makkelijk te onderhouden. Dit houdt in: vier tapes voor door de week, een tape voor de even vrijdagen en een tape voor de oneven vrijdagen. Voer elke dag een backup uit van alleen de gegevens die er op die dag bijgekomen zijn en zet op de desbetreffende vrijdagtape een volledige backup. Als je bepaalde belangrijke wijzigingen aanbrengt of enkele belangrijke gegevens aan je systeem toevoegt, zal een volledige backup op zijn plaats zijn.

9.3 Maak een backup van je RPM of Debian File Database

In het geval dat je systeem binnengedrongen wordt, kun je je RPM database gebruiken zoals je `tripwire` zou gebruiken, maar alleen als je er zeker van kan zijn dat het niet ook is aangepast. Je moet de RPM database kopiëren naar een diskette en deze kopie ten alle tijden off-line bewaren. De Debian distributie heeft waarschijnlijk iets dergelijks.

De bestanden `/var/lib/rpm/fileindex.rpm` en `/var/lib/rpm/packages.rpm` zullen waarschijnlijk niet op een enkele diskette passen. Maar gecomprimeerd zal elk wel op een enkele diskette passen.

Nu, als je systeem in gevaar is gebracht, kun je het volgende commando gebruiken:

```
root#  rpm -Va
```

om elk bestand op je systeem te verifiëren. Zie de `rpm` man pagina, want er zijn een paar andere opties die kunnen worden meegegeven om het minder verbose (uitgebreid) te maken. Denk eraan dat je er ook zeker van moet zijn dat je RPM binary niet in gevaar is gebracht.

Dit betekent dat elke keer dat een RPM wordt toegevoegd aan het systeem, de RPM database opnieuw moet worden gearchiveerd. Je moet de voordelen afwegen tegen de nadelen.

9.4 Houd je systeemlog gegevens bij

Het is erg belangrijk dat de informatie die afkomstig is van `syslog` niet gecompromitteerd is. De bestanden in `/var/log` lees- en schrijfbaar maken voor slechts een beperkt aantal gebruikers is een goed begin.

Houd een oogje op wat er daar weggeschreven wordt, speciaal onder de `auth` faciliteit. Veelvuldig mislukte logins bijvoorbeeld, kunnen een indicatie zijn voor een poging tot inbraak.

Waar je je logbestand moet zoeken hangt af van je distributie. Onder een Linux systeem dat in overeenstemming is met de "Linux Filesystem Standard", zoals Red Hat, moet je kijken in `/var/log` en `messages`, `mail.log` en anderen controleren.

Je kunt uitzoeken waar je distributie de logs wegschrijft door te kijken naar je `/etc/syslog.conf` bestand. Dit is een bestand dat `syslogd` (de systeem logging daemon) vertelt waar de diverse berichten moeten worden gelogd.

Misschien wil je ook je log-rotating script of daemon zo instellen dat ze de logs langer bewaren, zodat je de tijd hebt om ze te onderzoeken. Bekijk het `logrotate` pakket op recente Red Hat distributies. Andere distributies hebben waarschijnlijk een soortgelijk proces.

Als er met je logbestanden is geknoeid, kijk dan of je kan bepalen wanneer het geknoei is begonnen en met wat voor soort dingen geknoeid is. Zijn er grote periodes van tijd die niet gelogd zijn? Het zoeken op je backup tapes (als je die hebt) naar logbestanden waar niet mee geknoeid is, is een goed idee.

Indringers staan er bekend om dat ze logbestanden aanpassen om hun sporen uit te wissen, maar ze moeten toch worden gecontroleerd op vreemde gebeurtenissen. Je kunt de indringer in de gaten krijgen als hij probeert toegang te verkrijgen of een programma misbruikt om het root account te pakken te krijgen. Misschien zie je wel log entries voordat de indringer tijd heeft om ze aan te passen.

Je moet ook de `auth` faciliteit scheiden van andere loggegevens, evenals pogingen om van gebruiker te wisselen door gebruik te maken van `su`, login pogingen en andere loginformatie van gebruikers.

Stel, indien mogelijk, `syslog` zo in dat het een kopie van de belangrijkste gegevens stuurt naar een veilig systeem. Dit voorkomt dat een indringer zijn sporen uit kan wissen door het verwijderen van zijn login/su/ftp/etc pogingen. Zie de `syslog.conf` man pagina en ga naar de `@` optie.

Er zijn diverse meer geavanceerde `syslogd` programma's beschikbaar. Neem een kijkje op *http://www.core-sdi.com/ssyslog/* voor Secure Syslog. Met Secure Syslog kun je je syslog entries versleutelen om zeker te weten dat er niemand mee heeft geknoeid.

Een andere `syslogd` met meer mogelijkheden is *syslog-ng*. Hiermee heb je meer flexibiliteit in je logging en het kan ook voorkomen dat er met je remote syslog stromen wordt geknoeid.

Tot slot, logbestanden zijn veel minder bruikbaar als niemand ze leest. Maak zo af en toe eens wat tijd vrij om je logbestanden te bekijken om een indruk te krijgen hoe ze er op een gewone dag uitzien. Dit kan helpen om alles wat ongebruikelijk is te onderscheiden.

9.5 Maak gebruik van alle nieuwe systeem updates

De meeste Linux gebruikers installeren vanaf een CD-ROM. Doordat het tempo waarop beveiligingsfixes uitkomen hoog is, worden er altijd nieuwe (gecorrigeerde) programma's uitgegeven. Voordat je je machine verbindt met het netwerk, is het een goed idee om op de ftp site van je distributie te kijken en alle bijgewerkte pakketten, vanaf het moment dat je de CD-ROM van je distributie hebt ontvangen, te downloaden. Vaak bevatten deze pakketten belangrijke beveiligingsfixes, dus het is een goed idee om ze te installeren.

10 Wat te doen tijdens en na een inbraak

Dus je hebt enkele van de adviezen hier (of ergens anders) opgevolgd en een inbraak geconstateerd? Het eerste dat je moet doen is kalm blijven. Overhaaste acties kunnen meer schade aanrichten dan de aanvaller zou hebben gedaan.

10.1 Een aanval op de beveiliging is aan de gang

Het in de gaten krijgen van een aanval op de beveiliging die aan de gang is, kan een gespannen onderneming zijn. De manier waarop je reageert kan grote gevolgen hebben.

Als de aanval die je ziet een fysieke is, bestaat de kans dat je iemand hebt opgemerkt die heeft ingebroken in je huis, kantoor of laboratorium. Je zou de plaatselijke autoriteiten in moeten lichten. In een laboratorium kun je misschien iemand opgemerkt hebben die probeerde een kast te openen of een machine opnieuw op te starten. Afhankelijk van je autoriteit en procedures kun je hem vragen daarmee te stoppen of contact opnemen met lokale beveiligingsmensen.

Als je hebt geconstateerd dat een lokale gebruiker je beveiliging in gevaar tracht te brengen, is het eerste dat je moet doen je ervan vergewissen dat het inderdaad de persoon is die je denkt dat het is. Controleer de site waar vanaf hij inlogt. Is het de site waar vanaf hij normaal gesproken inlogt? Nee? Gebruik dan een niet-elektronische manier om contact te maken. Bel hem bijvoorbeeld op of loop naar zijn kantoor/huis en praat met hem. Als hij bevestigt dat hij verbinding heeft, kun je hem vragen om uit te leggen wat hij aan het doen was of hem vertellen dat hij ermee op moet houden. Als hij geen verbinding heeft en ook geen idee heeft waar je het over hebt, bestaat de kans dat dit incident verder uitgezocht moet worden. Bestudeer zulke incidenten en verzamel genoeg informatie voordat je enige beschuldiging uit.

Als je een aanval via het netwerk hebt geconstateerd, is het eerste dat je moet doen (als je daartoe de mogelijkheid hebt) het verbreken van de verbinding met het netwerk. Als ze verbonden zijn met een modem, haal de stekker van het modem er dan uit; als je verbonden zijn via Ethernet, haal dan de Ethernet kabel los. Dit voorkomt dat ze nog meer schade aanrichten. Ze zullen het waarschijnlijk als een netwerkprobleem zien en niet als een signaal dat ze opgemerkt zijn.

Als je de verbinding met het netwerk niet kunt verbreken (als je een drukke site hebt of je hebt geen fysieke controle over je machines), is de volgende stap om iets als `tcp_wrappers` of `ipfwadm` te gebruiken om toegang vanaf de site van de indringer te weigeren.

Als je niet alle mensen vanaf dezelfde site als de indringer de toegang kunt weigeren, zal afsluiten van het gebruikersaccount de oplossing zijn. Houd er rekening mee dat het afsluiten van een account niet gemakkelijk is. Denk aan de `.rhosts` bestanden, FTP toegang en een host met mogelijke achterdeuren.

Als je een van de bovenstaande dingen hebt gedaan (het netwerk afgesloten, toegang vanaf hun site geweigerd en/of hun account uitgeschakeld), moet je al hun gebruikersprocessen afsluiten en ze uitloggen.

Je moet je site de komende paar minuten goed in de gaten houden, want de aanvaller probeert om weer binnen te komen. Misschien door gebruik te maken van een ander account en/of vanaf een ander netwerkadres.

10.2 Een aanval heeft reeds plaatsgevonden

Dus je hebt ofwel een aanval opgemerkt die reeds heeft plaatsgevonden of je hebt het opgemerkt en (hopelijk) de overtredende aanvaller buiten je systeem gesloten. Wat nu?

10.2.1 Het gat dichten

Als het gelukt is om vast te stellen op welke manier de aanvaller je systeem is binnengedrongen, moet je proberen dat gat te dichten. Misschien zie je bijvoorbeeld diverse FTP entries net voordat de gebruiker inlogte. Schakel de FTP service uit en kijk of er een bijgewerkte versie van is of dat een van de mailing lists iets weet over een fix.

Controleer al je logbestanden en breng een bezoek aan je beveiligings lists en pagina's om te kijken of er een fix is voor nieuwe algemene misbruiken. Je kunt beveiligingsfixes voor Caldera vinden op *http://www.caldera.com/tech-ref/security/*. Red Hat heeft zijn beveiligingsfixes nog niet gescheiden van zijn bug fixes, maar hun distributie errata is beschikbaar op *http://www.redhat.com/errata*.

Debian heeft nu een mailing list over beveiliging en een webpagina. Zie: *http://www.debian.org/security/* voor meer informatie.

Het is erg waarschijnlijk dat als de ene distributeur een beveiligingsupdate heeft uitgegeven, de meeste andere Linux distributeurs dit ook zullen doen.

Er is nu een project dat de beveiliging onder Linux doorlicht. Ze gaan systematisch door alle user-space voorzieningen en kijken naar mogelijke beveiligingslekken en overflows. Uit hun aankondiging:

> "We proberen de Linux bronnen systematisch door te lichten teneinde net zo veilig te zijn als OpenBSD. We hebben reeds enkele problemen ontdekt (en opgelost), maar meer hulp is welkom. De lijst staat open voor iedereen en is tevens een bruikbaar hulpmiddel voor algemene discussies over beveiliging. Het adres van de lijst is: security-audit@ferret.lmh.ox.ac.uk. Stuur, om je in te schrijven, een e-mail naar: security-audit-subscribe@ferret.lmh.ox.ac.uk"

Als je de aanvaller niet buitensluit, komt hij waarschijnlijk terug. Niet alleen terug op je machine, maar terug ergens op je netwerk. Als hij een packet sniffer draaide, is de kans groot dat hij toegang heeft tot andere lokale machines.

10.2.2 De schade opnemen

Het eerste dat je moet doen is de schade opnemen. Waar is mee geknoeid? Als je een integrity checker zoals `Tripwire` draait, kun je die gebruiken om een integriteitscontrole uit te voeren; het helpt je met het bepalen waarmee is geknoeid. Zo niet, dan zul je al je belangrijke gegevens moeten nakijken.

Omdat Linux systemen steeds eenvoudiger te installeren zijn, kun je overwegen om je configuratiebestanden op te slaan, je disk(s) schoon te vegen, opnieuw te installeren en je gebruikersbestanden en configuratiebestanden vanaf backups terug te zetten. Zo ben je ervan verzekerd dat je een nieuw, schoon systeem hebt. Als je bestanden moet terugplaatsen vanaf een gecompromitteerd systeem, wees dan vooral voorzichtig met enige binary's die je terug plaatst, omdat het Trojan horses kunnen zijn die daar neergezet zijn door de indringer.

Als een indringer root toegang heeft verkregen, moet je opnieuw installeren. Bovendien wil je alle bewijs dat er is graag bewaren, dus het hebben van een reserve disk in de kluis is zinvol.

Vervolgens moet je je druk maken over hoe lang geleden het gebeurd is en of de backups beschadigd werk bevatten. Meer over backups volgt later.

10.2.3 Backups, backups, backups!

Het hebben van regelmatig gemaakte backups is een uitkomst voor beveiligingsaangelegenheden. Als je systeem gecompromitteerd is, kun je de gegevens die je nodig hebt herstellen vanaf de backups. Natuurlijk zijn sommige gegevens ook voor de aanvaller waardevol. Ze zullen ze niet alleen vernietigen, ze zullen ze stelen en er kopieën voor henzelf van maken; maar je hebt in ieder geval de gegevens nog.

Je moet diverse eerdere backups controleren voordat je een bestand herstelt waarmee geknoeid is. De indringer kan je bestanden al lang geleden hebben gecompromitteerd en je kunt veel succesvolle backups gemaakt hebben van het gecompromitteerde bestand.

Natuurlijk kleven er ook een aantal beveiligingsbezwaren aan backups. Zorg ervoor dat je ze op een veilige plaats bewaart. Weet wie er toegang toe heeft. (Als een indringer je backups te pakken kan krijgen, heeft hij toegang tot al je gegevens zonder dat je het ooit te weten komt.)

10.2.4 De indringer traceren

Ok, je hebt de indringer buitengesloten en je systeem hersteld, maar je bent nog niet helemaal klaar. Hoewel het onwaarschijnlijk is dat de meeste indringers ooit worden opgepakt, moet je aangifte doen van de aanval.

Je moet de aanval rapporteren aan de beheerder van de site vanwaar de aanvaller je systeem heeft aangevallen. Je kunt deze beheerder opzoeken met `whois` of de Internic database. Je zou hem een e-mail kunnen sturen met alle van toepassing zijnde log entries, datums en tijden. Als je iets anders opmerkerkelijks over je indringer is opgevallen, moet je dat ook melden. Na de e-mail verstuurd te hebben, zou je dit (mocht je daartoe geneigd zijn) moeten laten volgen door een telefoontje. Als die beheerder op zijn beurt je aanvaller in de gaten krijgt, kan contact worden opgenomen met de beheerder van de site waar de aanvaller vandaan komt enzovoort.

Goede crackers gebruiken vaak veel bemiddelende systemen. Sommige (of veel) daarvan weten wellicht niet eens dat ze zijn gecompromitteerd. Proberen om het spoor van een cracker terug te volgen naar zijn eigen systeem kan moeilijk zijn. Wees beleefd tegen de beheerders waar je mee praat, ze kunnen je een heel eind op weg helpen.

Je moet ook de beveiligingsorganisaties waar je deel van uitmaakt op de hoogte stellen (*CERT* `<http://www.cert.org/>` of soortgelijk), evenals de verkoper van je Linux systeem.

11 Bronnen

Er zijn VEEL goede sites over de beveiliging van Unix in het algemeen en over de beveiliging van Linux in het bijzonder. Het is erg belangrijk om je te abonneren op een (of meer) van de beveiligings mailing lists en bij te blijven op het gebied van beveiligingsfixes. De meeste van deze lists zijn klein van omvang en erg informatief.

11.1 FTP Sites

CERT is het Computer Emergency Response Team. Ze versturen vaak waarschuwingen voor recente aanvallen en fixes. Zie *ftp://ftp.cert.org* voor meer informatie.

ZEDZ (voorheen Replay) (*http://www.zedz.net*) heeft archieven van vele beveiligingsprogramma's. Omdat ze zich buiten de VS bevinden, hoeven ze zich niet te houden aan de coderingsbeperkingen van de VS.

Matt Blaze is de auteur van CFS en een goede beveiligingsadvocaat. Matt's archief is beschikbaar op *ftp://ftp.research.att.com/pub/mab* `<ftp://ftp.research.att.com/pub/mab>`

`tue.nl` is een goede Nederlandse FTP site over beveiliging. *ftp.win.tue.nl*

11.2 Websites

- The Hacker FAQ is een FAQ over hackers: *The Hacker FAQ*
- Het COAST archief heeft een groot aantal beveiligingsprogramma's en informatie voor Unix: *COAST*
- SuSe Security Page: *http://www.suse.de/security/*
- Rootshell.com is een goede site om te zien welke soorten misbruik er tegenwoordig worden gepleegd door crackers: *http://www.rootshell.com/*
- BUGTRAQ geeft advies over beveiligingsonderwerpen: *BUGTRAQ archives*
- CERT, het Computer Emergency Response Team, geeft advies over algemene aanvallen op Unix platvormen:*CERT home*
- Dan Farmer is de auteur van SATAN en vele andere beveiligingstools. Zijn eigen site bevat een handig overzicht met beveiligingsinformatie, evenals beveiligingstools: *http://www.trouble.org*
- De Linux security WWW is een goede site voor infomatie over de beveiliging van Linux: *Linux Security WWW*
- Infilsec heeft een kwetsbaarheidsengine die je kan vertellen welke kwetsbaarheden betrekking hebben op een bepaald platform: *http://www.infilsec.com/vulnerabilities/*
- CIAC verstuurt periodieke beveiligingsbulletins over algemeen misbruik: *http://ciac.llnl.gov/cgi-bin/index/bulletins*
- Een goed startpunt voor Linux Pluggable Authentication modules kan worden gevonden op *http://www.kernel.org/pub/linux/libs/pam/*.
- Het Debian project heeft een webpagina voor hun beveiligingsfixes en informatie. Het staat op *http://www.debian.com/security/*.
- WWW Security FAQ, geschreven door Lincoln Stein, is een goed naslagwerk over web beveiliging. Je kunt het vinden op *http://www.w3.org/Security/Faq/www-security-faq.html*

11.3 Mailing Lists

Bugtraq: Voor een abonnement op bugtraq stuur je een e-mail naar listserv@netspace.org, waarbij in de inhoud van het bericht "subscribe bugtraq"staat. (Zie de verwijzingen hierboven voor archieven).

CIAC: Stuur een e-mail naar majordomo@tholia.llnl.gov. Zet in de INHOUD (niet onderwerp) van het bericht "subscribe ciac-bulletin".

Red Hat heeft een aantal mailing lists, waarvan de belangrijkste de redhat-announce list is. Je kunt er lezen over beveiligings (en andere) fixes zodra ze uitkomen. Stuur een e-mail naar redhat-announce-list-request@redhat.com met als onderwerp "Subscribe". Zie *http://www.redhat.com/mailing-lists/redhat-announce-list/* voor meer informatie en archieven.

Het Debian project heeft een beveiligings mailing list die hun beveiligingsfixes behandelt. Zie *http://www.debian.com/security/* voor meer informatie.

11.4 Boeken - Gedrukt materiaal

Er zijn een aantal goede boeken over beveiliging in omloop. Deze paragraaf somt een klein aantal hiervan op. In aanvulling op de boeken die specifiek over beveiliging gaan, wordt beveiliging behandeld in een aantal andere boeken over systeembeheer.

Building Internet Firewalls door D. Brent Chapman & Elizabeth D. Zwicky
1e druk september 1995
ISBN: 1-56592-124-0

Practical UNIX & Internet Security, 2e druk door Simson Garfinkel & Gene Spafford
2e druk april 1996
ISBN: 1-56592-148-8

Computer Security Basics By Deborah Russell & G.T. Gangemi, Sr.
1e druk juli 1991
ISBN: 0-937175-71-4

Linux Network Administrator's Guide door Olaf Kirch
1e druk januari 1995
ISBN: 1-56592-087-2

PGP: Pretty Good Privacy door Simson Garfinkel
1e druk december 1994
ISBN: 1-56592-098-8

Computer Crime A Crimefighter's Handbook door David Icove, Karl Seger & William VonStorch (Consulting Editor Eugene H. Spafford)
1e druk augustus 1995
ISBN: 1-56592-086-4

Linux Security door John S. Flowers
New Riders
ISBN: 0735700354
maart 1999

Maximum Linux Security : A Hacker's Guide to Protecting Your Linux Server and Network
Anoniem
Paperback - 829 pages
Sams

ISBN: 0672313413
juli 1999

Intrusion Detection door Terry Escamilla
Paperback - 416 pagina's (September 1998)
John Wiley and Sons
ISBN: 0471290009

Fighting Computer Crime
Donn Parker
Paperback - 526 pagina's (September 1998)
John Wiley and Sons
ISBN: 0471163783

12 Verklarende woordenlijst

- **authenticatie:** Het te weten komen of de ontvangen gegevens hetzelfde zijn als de verzonden gegevens en of de beweerde afzender inderdaad de werkelijke afzender is.

- **bastion host:** Een computersysteem dat zwaar beveiligd moet zijn, omdat het kwetsbaar is voor aanvallen, meestal omdat het is blootgesteld aan het Internet en een belangrijk contactpunt is voor gebruikers van internationale netwerken. Het dankt zijn naam aan de geavanceerde verdedigingsprojecten op de buitenmuren van middeleeuwse kastelen. Bastions overzien kritieke verdedigingsgebieden, hebben meestal sterke muren, ruimte voor extra mankracht en het soms nuttige vat met kokend hete olie om aanvallers te ontmoedigen.

- **buffer overloop:** De algemene coderingsstijl is om nooit buffers die groot genoeg zijn toe te wijzen en niet te controleren op overlopen. Als zulke buffers overlopen, kan het uitvoerende programma (daemon of set-uid programma) worden verlokt tot het doen van andere dingen. Gewoonlijk gebeurt dit door het overschrijven van het retouradres van een functie om naar een andere locatie te verwijzen.

- **denial of service:** Een aanval die de hulpbronnen op je computer gebruikt voor dingen die het niet zou moeten doen en zodoende het normale gebruik van je netwerkbronnen voor legitieme doeleinden verhindert.

- **dual-homed host:** Een computersysteem voor algemene doeleinden dat op z'n minst twee netwerk interfaces heeft.

- **firewall:** Een component of set van componenten dat de toegang beperkt tussen een beveiligd netwerk en het Internet of tussen andere netwerken.

- **host:** Een computersysteem dat aangesloten is op een netwerk.

- **IP spoofing:** IP Spoofing is een complexe technische aanval die bestaat uit diverse onderdelen. Het is een beveiligingslek dat werkt door computers in een vertrouwensrelatie te laten denken dat je iemand bent die je in werkelijkheid niet bent. Er is een uitgebreid stuk geschreven over deamon9, route en infinity in Volume Zeven, uitgave 48 van Phrack Magazine.

- **non-repudiation:** De mogelijkheid die een ontvanger heeft om te kunnen bewijzen dat de afzender van bepaalde gegevens inderdaad de gegevens verstuurd heeft, zelfs wanneer de afzender later ontkent dat hij het ooit verstuurd heeft.

- **pakket:** Het basisonderdeel van communicatie op het Internet.

- **pakket filtering:** De actie die een apparaat onderneemt om selectief de gegevensstroom naar en vanaf een netwerk te beheren. Pakketfilters staan pakketten toe of blokkeren ze, gewoonlijk terwijl ze ze routen van het ene netwerk naar het andere (veelal vanaf het Internet naar een intern netwerk en vice versa). Om pakketfiltering tot stand te brengen, stel je regels op die bepalen welke soorten pakketten (degenen naar of vanaf een bepaald IP adres of poort) worden toegestaan en welke soorten geblokkeerd worden.

- **perimeter netwerk:** Een netwerk dat toegevoegd is tussen een beschermd netwerk en een extern netwerk, bedoeld om te voorzien in een aanvullende beveiligingslaag. Een perimeter netwerk wordt soms een DMZ genoemd.

- **proxy server:** Een programma dat de externe servers beheert ten behoeve van interne clients. Proxy clients communiceren met proxy servers, wiens relais clientverzoeken aan echte servers goedkeurt waarna het relais antwoord terug geeft aan clients.

- **superuser:** Een informele naam voor `root`.

13 Veel gestelde vragen

1. Is het veiliger om ondersteuning van stuurprogramma's direct in de kernel te compileren, in plaats van het een module te maken?

   Antwoord: Sommige mensen denken dat het beter is om de mogelijkheid tot het laden van stuurprogramma's voor apparaten middels modules uit te schakelen, omdat een indringer een Trojan module of een module die invloed kan hebben op de beveiliging van het systeem kan laden.

   Maar om modules te kunnen laden moet je root zijn. De module object bestanden zijn ook alleen beschrijfbaar door root. Dit betekent dat een indringer roottoegang nodig heeft om een module te plaatsen. Als de indringer roottoegang verkrijgt, zijn er meer serieuze zaken om je zorgen over te maken dan of hij al of niet een module kan laden.

Modules zijn bedoeld voor het dynamisch laden van ondersteuning voor een bepaald apparaat dat zelden gebruikt wordt. Op server machines of firewalls bijvoorbeeld, is het erg onwaarschijnlijk dat dit gebeurt. Om deze reden heeft het meer zin om ondersteuning voor machines die opereren als een server direct in de kernel te compileren.

2. Waarom mislukt het inloggen als root vanaf een remote machine altijd?

   Antwoord: Zie 4.2 (Root beveiliging). Dit is bewust gedaan om te voorkomen dat gebruikers via `telnet` een verbinding als root tot stand proberen te brengen, hetgeen een ernstige beveiligingskwetsbaarheid is, omdat dan het root wachtwoord, in leesbare tekst, verzonden zou worden over het netwerk. Vergeet niet: mogelijke indringers hebben de tijd en kunnen programma's uitvoeren die automatisch naar je wachtwoord zoeken.

3. Hoe schakel ik "shadow passwordsöp mijn Red Hat 4.2 of 5.x Linux box uit?

   Antwoord: Om "shadow passwordsüit te schakelen, voer je `pwconv` uit als root. Nu zou `/etc/shadow` moeten bestaan en worden gebruikt door applicaties. Als je Red Hat 4.2 of hoger gebruikt, zullen de PAM modules zich automatisch aanpassen aan de verandering van het gebruik van het normale `/etc/passwd` naar "shadow passwords"zonder enige andere wijziging.

   Een stukje achtergrondinformatie: "shadow passwordsïs een techniek om je wachtwoord in een bestand, anders dan het normale `/etc/passwd` bestand, op te slaan. Dit heeft verscheidene voordelen. Het eerste is dat het schaduw bestand, `/etc/shadow`, alleen leesbaar is voor root, in tegenstelling tot `/etc/passwd`, wat leesbaar moet blijven voor iedereen. Het andere voordeel is dat je als beheerder accounts kan vrijgeven af afsluiten, zonder dat iedereen de status van andere gebruikersaccounts weet.

   Het `/etc/passwd` bestand wordt dan gebruikt om gebruiker- en groepsnamen in op te slaan, die worden gebruikt door programma's als `/bin/ls` om het gebruikers ID naar de juiste gebruikersnaam om te zetten in een directoryweergave.

   Het `/etc/shadow` bestand bevat dan alleen de gebruikersnaam en zijn/haar wachtwoord en misschien informatie over het account, zoals wanneer het account vervalt e.d.

   Om "shadow passwordsïn te schakelen, voer je `pwconv` uit als root. Nu zou `/etc/shadow` moeten bestaan en worden gebruikt door applicaties. Omdat je Red Hat 4.2 of hoger gebruikt, zullen de PAM modules zich automatisch aanpassen aan de verandering van het gebruik van het normale `/etc/passwd` naar "shadow passwords"zonder enige andere wijziging.

Omdat je geïnteresseerd bent in het beveiligen van je wachtwoorden, zul je wellicht ook geïnteresseerd zijn in de totstandkoming van goede wachtwoorden op zich. Hiervoor kun je de `pam_cracklib` module gebruiken, die onderdeel uitmaakt van PAM. Het kijkt of je wachtwoord voortkomt in de "Crack libraries", om je te helpen met de beslissing of het te gemakkelijk te raden is door programma's die wachtwoorden kunnen kraken.

4. Hoe kan ik de Apache SSL extensies inschakelen?

   Antwoord:

   (a) Haal SSLeay 0.8.0 of hoger op vanaf `<ftp://ftp.psy.uq.oz.au/pub/Crypto/SSL>`.
   (b) Bouw, test en installeer het!
   (c) Haal de Apache 1.2.5 source op.
   (d) Haal de Apache SSLeay extensies op vanaf *here* `<ftp://ftp.ox.ac.uk/pub/crypto/SSL/apache_1.2.5+ssl_1.13.tar.gz>`.
   (e) Pak het uit in de apache-1.2.5 source directory en patch Apache zoals beschreven in README.
   (f) Configureer en bouw het.

   Je kunt ook *ZEDZ net* proberen, dat veel kant en klare pakketten heeft en zich buiten de Verenigde Staten bevindt.

5. Hoe kan ik gebruikersaccounts bewerken en toch de beveiliging behouden?

   Antwoord: De Red Hat distributie, speciaal Red Hat 5.0, bevat een groot aantal tools om de eigenschappen van gebruikersaccounts te veranderen.

   - De `pwconv` en `unpwconv` programma's kunnen gebruikt worden om te wisselen tussen "shadowën "non-shadowed"wachtwoorden.
   - De `pwck` en `grpck` programma's kunnen gebruikt worden om te verifiëren of de `passwd` en `group` bestanden juist ingedeeld zijn.
   - De `useradd`, `usermod` en `userdel` programma's kunnen gebruikt worden om gebruikersaccounts toe te voegen, te verwijderen en aan te passen. De `groupadd`, `groupmod` en `groupdel` programma's doen hetzelfde voor groepen.
   - Wachtwoorden voor groepen kunnen met behulp van `gpasswd` aangemaakt worden.

   Al deze programma's zijn "shadow-aware-- dat houdt in dat als je "shadowïnschakelt, ze `/etc/shadow` zullen gebruiken voor wachtwoordinformatie, anders doen ze dat niet. Zie de respectieve man pagina's voor aanvullende informatie.

6. Hoe kan ik met behulp van Apache bepaalde HTML documenten met een wachtwoord beveiligen?

   Ik wed dat je niet wist van het bestaan van *http://www.apacheweek.org* of wel?

Je kunt informatie over het authenticeren van gebruikers vinden op *http://www.apacheweek.com/features/userauth* evenals andere web server beveiligingstips van *http://www.apache.org/docs/misc/security_tips.html*

14 Conclusie

Door je in te schrijven op de mailing lists voor beveiligingswaarschuwingen en bij te blijven, kun je een hoop doen met het oog op beveiliging van je machine. Als je je logbestanden in de gaten houdt en iets als `tripwire` regelmatig uitvoert, kun je zelfs nog meer doen.

Een verstandig niveau van computerbeveiliging is niet moeilijk te onderhouden op een machine voor thuisgebruik. Meer moeite is vereist bij zakelijke machines, maar Linux kan zeker een veilig platform zijn. Dankzij het karakter van de ontwikkeling van Linux, komen beveiligingsoplossingen vaak veel sneller uit dan die voor commerciële besturingssystemen, wat Linux een ideaal platform maakt als beveiliging een vereiste is.

15 Dankbetuigingen

De informatie hier is verzameld uit vele bronnen. Dank aan de volgenden die zowel indirect als direct hebben bijgedragen:

Rob Riggs *rob@DevilsThumb.com*

S. Coffin *scoffin@netcom.com*

Viktor Przebinda *viktor@CRYSTAL.MATH.ou.edu*

Roelof Osinga *roelof@eboa.com*

Kyle Hasselbacher *kyle@carefree.quux.soltc.net*

David S. Jackson *dsj@dsj.net*

Todd G. Ruskell *ruskell@boulder.nist.gov*

Rogier Wolff *R.E.Wolff@BitWizard.nl*

Antonomasia *ant@notatla.demon.co.uk*

Nic Bellamy *sky@wibble.net*

Eric Hanchrow *offby1@blarg.net*

Robert J. Berger*rberger@ibd.com*

Ulrich Alpers *lurchi@cdrom.uni-stuttgart.de*

David Noha *dave@c-c-s.com*

Pavel Epifanov *epv@ibm.net*

Joe Germuska *joe@germuska.com*

Franklin S. Werren *fswerren@bagpipes.net*

Paul Rusty Russell <*Paul.Russell@rustcorp.com.au*>

Christine Gaunt <*cgaunt@umich.edu*>

lin *bhewitt@refmntutl01.afsc.noaa.gov*

A.Steinmetz *astmail@yahoo.com*

Jun Morimoto *morimoto@xantia.citroen.org*

Xiaotian Sun *sunx@newton.me.berkeley.edu*

Eric Hanchrow *offby1@blarg.net*

De volgende personen hebben deze HOWTO vertaald in verschillende andere talen!

Speciale dank aan hen allemaal voor hun hulp bij het verspreiden van het Linux woord ....

Pools: Ziemek Borowski *ziembor@FAQ-bot.ZiemBor.Waw.PL*

Japans: FUJIWARA Teruyoshi *fjwr@mtj.biglobe.ne.jp*

Indonesisch: Tedi Heriyanto *22941219@students.ukdw.ac.id*

Koreaans: Bume Chang *Boxcar0001@aol.com*

Spaans: Juan Carlos Fernandez *piwiman@visionnetware.com*

Nederlands: Nine Matthijssen *smurfin@nl.linux.org*