Securizando uma rede heterogénia com utilitários de software livres

by Georges Tarbouriech
<georges.t(at)linuxfocus.org>

About the author:

O George já é um utilizador do Unix de há longa data. Ele agradece à comunidade de software livre por nos fornecer imensos utilitários de segurança.

Translated to English by:
Georges Tarbouriech
<georges.t(at)linuxfocus.org>

Abstract:

Este artigo foi primeiro publicado numa revista francesa de Linux, numa edição especial dedicada à segurança. O editor, os autores, os tradutores, gentilmente permitiram à LinuxFocus publicar cada um dos artigos desta edição especial. Assim, a LinuxFocus trá−los−á até si logo que estejam traduzidos para Inglês. O meu agradecimento a todas as pessoas envolvidas neste trabalho. Este resumo será reproduzido para cada artigo tendo a mesma origem.

Preâmbulo

A segurança das redes de computadores é, provavelmente, um dos maiores desafios da tecnologia do século XXI.
Contudo, como noutros campos preocupantes, toda a gente fala acerca de, mas quem se devia sentir mais afectado não parece ter detectadado a dimensão de um potencial desastre. Os "mais afectados" são os principais software's ou os designers de sistemas. O melhor exemplo, mais uma vez, vem da Redmond, onde a segurança parece ser uma palavra imperativa, pelo menos sobre "menos controle" que o marketing, por exemplo.
Felizmente, as duas últimas décadas do século XX viram o nascimento do Software livre e a filosofia adjacente ao mesmo. Se "deseja" melhorar a segurança das suas máquinas, dos seus sistemas, das suas redes... é aqui que tem de procurar. A comunidade de software livre fez muito mais pela segurança que todas as grandes companhias de software juntas.
Tendo isto dito, os utilitários não fazem tudo, e securizar uma rede, por exemplo, é um trabalho permanente:

novas oportunidades a cada momento!
Isto quer dizer que nunca conseguirá dizer que a sua rede está 100% segura. Pode, somente, reduzir os riscos. O que mostramos aqui, é uma pequena parte do que pode fazer para limitar os riscos. Depois de ler esta edição especial (Nota do Autor: lembre−se que este artigo foi parte de uma edição especial dedicada à segurança de uma revista de Linux francesa) saberá um pouco mais acerca de segurança, mas de nenhum modo poderá dizer que a sus rede está segura. Foi avisado!
Por último mas não o menos importante: tal artigo não pode ser exaustivo. Existe muita literatura nesta da matéria e está longe de retractar todo a problemática. Assim, não espere que este artigo mencione tudo, SOs, utilitários, configurações...
Para terminar este preâmbulo, adicione−se que algumas partes deste artigo são "roubadas" da LinuxFocus, mas não se preocupe com a concordância do autor: acontece que é só um e a mesma pessoa !

Apresentação

Primeiro, falaremos acerca de uma estrutura de uma rede muito heterogénea, contendo sistemas mais ou menos difundidos. Quanto mais SOs, maior é a complexidade visto que nem todos são iguais perante o adversário. Para além disto, as máquinas usadas como servidores deviam ter funções diferentes numa rede: utilizaremos uma rede diversificada.
De seguida, passaremos por um conjunto de utilitários essenciais para melhorar a segurança. A escolha será arbitária: são muito numerosos para os mencionar a todos. Obviamente que explicaremos como securizar as suas máquinas e redes com estes utilitários. O próximo capítulo reverá algumas características de sistemas diferentes durante a etapa de segurança.
A conclusão tentará explicar a "relatividade" dos processos de securização, para lhe mostrar porque é um caminhos longo, sem "entrar" em futurismos.

Exemplo de uma rede heterogénea

Como primeira vantagem, o protocolo TCP/IP é "falado" por todos os SOs na terra. Com ele, muitos sistemas diferentes são capazes de comunicar uns com os outros. Assim, na rede utilizaremos como exemplo o TCP/IP estará sempre presente. Por outras palavras não mencionaremos protocolos proprietários, os menos difundidos, nem sequer os desactualizados. Nem sequer falaremos da estrutura física, que é o tipo de ligação, de categoria, etc.
Assim, nesta rede, meteremos um pouco de tudo. Claro que encontraremos o Unix, proprietário ou livre: por exemplo, uma amostra de Solaris 2.6, ou SunOS 5.6, um pouco de QNX ou NeXTSTEP, ou NetBSD ou OpenBSD. No lado "convencional" incluiremos o único e Não Terminado 4.0 (não nenhum outro, são piores). Aqui também poderíamos ter adicionado o OS2 que é menos mau. Po último adicionaremos uma amostra de "inconvencional", digamos BeOS e AmigaOS (sim, existe... bem mas não tão real!)
Claro que alguns de vós se estão a queixar: o quê não há AIX ou HP−UX? Não! Se gostássemos de mencionar todo o Unix, seria artigo de dez volumes. Contudo as regras básicas de segurança são aplicáveis a todos os sistemas.

Agora o que lhe vamos pedir ?
Por exemplo, digamos que o Solaris será o servidor de aplicações. O Irix administrará as cópias de segurança. O NT será outro servidor de aplicações. O Linux será um gateway. Outra máquina Linux será um servidor http ou base de dados. Todas as outras máquinas são clientes. Consideremos que esta rede tem cerca de 30 máquinas a utilizar ficheiros com palavras−passe para autenticação. Podíamos ter escolhido uma tecnologia mais sofisticada de autenticação: NIS (Yellows Pages) ou LDAP ou Kerberos... Façamos as coisas simples!

Nem utilizaremos o NFS. Mesmo que possa ser útil, quando a segurança é uma preocupação, é melhor esquecê−lo, apesar de alguns melhoramentos. Em França, pessoas mais idosas, costumavam dizer "não ponha todos os ovos no mesmo cesto". Assim os serviços ou protocolos "incertos" mais necessários só estarão presentes uma vez em máquinas que nada mais fazem. Por exemplo, só um servidor de ftp, um servidor de http, de preferência em máquinas Unix. Algumas máquinas Unix serão servidores SSH e outros clientes SSH. Mais acerca disto, posteriormente. Utilizaremos endereços estáticos: sem DHCP. Por outras palavras, ficaremos básicos! Claro que isto pode ser aplicado para uma rede com 50 máquinas: com muitas mais seria um pesadelo.

Utilitários e como utilizá−los

Como é habitual, existe mais do que um modo de o fazer (TIMTOWDI). O caso ideal seria começar desde o principio, com máquinas para instalar e rede por configurar. Mas isto só é verdades nos filmes! Assim, consideremos uma rede que cresce com o tempo, com máquinas a passarem de um lado para outro, com novas máquinas a chegar e, por aí adiante. Devido à "corrida" Mhz, por exemplo, as máquinas Intel não duram muito. Após cerca de 3 anos, torna−se um pouco difícil de encontrar material às peças. Por conseguinte ou recicla as máquinas para tarefas subsidiárias ou livra−se delas: triste mas verdade! Felizmente, algumas outras duram muito mais tempo e merecem ser melhoradas. Não acredito que isto é fora de tópico: um administrador deve trabalhar com alta disponibilidade em mente.

As bases

Ao primeiro passo do trabalho podia chamar "generalidades". Consiste em remover tudo o que é inútil em todas as máquinas: não é uma tarefa "leve"!. Cada SO, incluindo o Unix, instala um número inacreditável de serviços, protocolos que nunca irá precisar. A palavra mãe é: atire−os fora! No Unix, um modo simples ... e imperfeito é comentar tudo no /etc/inetd.conf. Com isto já são menos uns serviços. Claro que isto é um pouco exagerado, mas em muitas máquinas é perfeitamente aceitável. Depende das suas necessidades. Sob o Linux e alguns outros pode utilizar o comando *chkconfig* para desactivar alguns serviços.
Verifique os ficheiros SUID/SGID e não hesite em remover o bit "faulty" ou reconsidere a possibilidade de desactivar o programa. Um comando como: *find / −user root −a \(−perm −4000 −o −perm −2000 \) −print* will give you the list of those files. To remove the "s" bit, type *chmod a−s programname* (note: claro que perde alguma funcionalidade ao remover o bit "s". Ele tem um propósito ao fim de contas).
Remova programas "perigosos" ou outros conhecidos como "risky": por exemplo, os comandos remotos como rsh, rlogin, rcp... . O SSH substitui−los−à muito bem.
Verifique as permissões para os directórios tais como /etc, /var... Quanto mais restritivas melhor. Por exemplo, um comando como *chmod −R 700* no directório contendo os ficheiros de arranque (/etc/rc.d/init.d em muitos Unixes) nem é má ideia. A mesma regra aplica−se a todos os sistemas pertencentes à rede: remova o que não utiliza, ou pelo menos, desactive. Para o NT, sinta−se à vontade de parar um número máximo de serviços a partir do painel de configuração. Existem imensas "coisas" básicas para fazer e já existe muita documentação sobre o assunto por aí.

Os utilitários

Comecemos com o Unix, visto que é o único a levar em conta os problemas de segurança. De seguida, existe uma enorme quantidade de utilitários livres e a maioria trabalha (praticamente) nos clones de Unix.

Por enquanto, trabalharemos nas máquinas individuais, visto que securizar uma rede significa tornar seguros os seus elementos. A instalação destes utilitários é bastante simples, por essa razão não dedicaremos muito tempo ao assunto. Os seus parâmetros dependem dos sistemas, das necessidades... Cabe−lhe a si aplicá−los para os seus próprios casos. O primeiro utilitário chama−se *shadow utils*. E significa a encriptação de

passwords. Felizmente, faz parte de muitas distribuições Unix. O ficheiro /etc/shadow é "criado" a partir do ficheiro /etc/passwd.

Ainda melhor, o *PAM* (Módulos de Autenticação "Pluggable") permite restringir o acesso de utilizadores por serviço. Tudo é administrado a partir do directório contendo os ficheiros de configuração para cada serviço em questão, normalmente /etc/pam.d. Muitos serviços podem ser PAM "driven", tais como o ftp, login, xdm, etc. permitindo ao administrador escolher quem tem o direito de fazer o quê.

O próximo utilitário é obrigatório: um *TCPWrapper* . Também trabalha em quase todos os clones Unix. Brevemente, permite restringir o acesso por determinadas máquinas a alguns serviços. Estas máquinas são permitidas ou proibidas usando dois ficheiros: /etc/hosts.allow e /etc/hosts.deny. O TCPWrapper pode ser configurado de dois modos: quer movendo os demónios ou alterando o ficheiro /etc/inetd.conf. Mais tarde, veremos que o TCPWrapper trabalha bem em conjunto com outros utilitários. Pode encontrar o TCPWrapper em ftp://ftp.porcupine.org/pub/security

Um outro utilitário interessante é o xinetd. Brevemente, o xinetd é um substituto do inetd com mais alguns extras. Não insistiremos tendo em conta o que se disse acima acerca do inetd. Se estiver interessado, pode encontrá−lo em http://www.xinetd.org.

No Linux, existe um utilitário sem o qual não pode deixar de viver: chama−se Bastille−Linux. Encontra−o em http://www.bastille−linux.org. Este utilitário, escrito em Perl, não é só didáctico como muito eficiente. Depois de correr a script, responde a muitas questões e o Bastille−Linux age de acordo. Cada questão é explicada e as respostas por omissão são dadas. Pode desfazer as modificações, recomeçar uma nova configuração, verificar o que foi feito... Está tudo lá! Inclusive, até oferece uma configuração de firewall: mais acerca disto posteriormente. Na altura da edição deste artigo a Bastille−Linux tem a versão 1.1.1 mas a versão 1.2.0 já está disponível como uma possível candidata. Está melhorada e fornece uma GUI baseada em Tk e está em módulos Perl. (Nota do autor: este artigo foi escrito há muitos meses atrás. De facto a versão actual do Bastille−Linux é a 1.3.0).

Os sistemas de detecção de intrusos são também essenciais. Os dois com mais peso chamam−se snort e portsentry. O primeiro pode ser obtido em http://www.snort.org e o segundo a partir do website da Abacus, http://www.psionic.com. Estes utilitários não devem ser comparados: o primeiro é um NIDS (Sistema de detecção de intrusos em redes) fornecendo, principalmente, informação, enquanto que o segundo é considerado ser orientado a máquinas e mais activo. O snort tem imensas opções para supervisionar o tráfego da rede. Pode escutar tudo o que quiser: entrada, saída, dentro da firewall, fora da firewall. Claro, que pode depois criar logs enormes, mas você sabe o que quer! Uma versão Win32 está disponível e, é importante se considerarmos o número de utilitários livres nestes "sistemas".

O portsentry tem uma característica interessante: pode bloquear portas segundo a nossa escolha. Pode redireccionar o atacante para um endereço inutilizado ou redireccioná−lo para a firewall. Claro que pode escolher quem bloquear e não bloquear. Agora podemos voltar ao TCPWrapper: o portsentry é capaz de escrever dentro do ficheiro /etc/hosts.deny se o pretender. Assim o portsentry torna−se mais eficiente. Não entraremos no debate acerca da filosofia do portsentry utilizando a blindagem de portas. É consigo: faça a sua escolha após aprofundar a matéria. Avise−se que o portsentry pode tornar uma máquina "invisível" o que não é mau! Por último o portsentry pode operar em modos diferentes, o mais avançado está "reservado" para o Linux (pelo menos por enquanto).

Não podemos falar de segurança sem mencionar a encriptação. Contudo a lei acerca de é diferente de país para país e por vezes é proibido o uso da encriptação.
Nota do Autor: a secção seguinte foi eliminada da versão Inglesa deste artigo visto só dizer respeito à lei Francesa.
Conclusão: se o seu país permite encriptação, instale clientes e servidores ssh nas suas máquinas Unix

(Segundo as suas necessidades!).

Para terminar os utilitários do Unix, mencionemos aqueles que pertencem aos Unixes proprietários. No Solaris, você tem o ndd, aset; no Irix pode usar o ipfilterd. O MacOS X dá−lhe alguns utilitários livres: ssh, ipfwadm...
Voltamos mais tarde a isto.

Agora, falemos acerca do único e solitário (felizmente!) Não Terminado 4.0. Aqui não podemos falar de utilitários livres... contudo, o homem de Redmond dá−nos algum material "livre" para melhorar as características do sistema (não tem nada haver com correcções de bugs, visto que não existem bugs!). Respeitante à segurança, o NT 4.0 é um modelo... de absurdidade. É um pouco como um mexeriqueiro! Não importa. Assim, só tem de obter o último service pack (6 na altura de edição deste artigo) e as HotFixes... que são patches de segurança. De seguida... pode obter alguns utilitários de software livre (com o significado de disponíveis livremente mas sem código fonte). E é tudo.

Para outros sistemas temos de pesquisar. Para o AmigaOS, o desenvolvimento não parece motivar muitas pessoas e a camada TCP/IP é um pouco velha. Contudo o dominio público ainda está lá para o manter ocupado. Respeitante ao BeOS, as coisas não estão melhor: este SO excelente para ter um futuro comprometido e a camada de rede chamada Bone ainda está sob trabalhos.
(Nota do Autor: infelizmente, o BeOS está morto. Umas poucas pessoas tentaram mante−lo vivo como um produto de software livre... e fazem um excelente trabalho.)
Mas aqui também, encontrará alguns utilitários do mundo do Unix para melhorar as coisas.

Securizando as máquinas

Agora, tem de configurar isto tudo! Novamente, consideremos que cada máquina Unix está "equipada" com utilitários shadow, PAM, TCPWrapper, que todo o serviço inútil foi parado ou removido, que as permissões foram fortalecidas nos directórios "sensíveis", etc.

Nas máquinas Linux, é altura de correr o Bastille−Linux. (Este utilitário deveria trabalhar na maioria das distribuições do Linux, contudo, originalmente, foi desenhado para a RedHat e Mandrake). Sinta−se à vontade de responder às questões de um modo restritivo.
Na máquina Linux usada como gateway, o sistema deve ser "minimalístico". Pode remover a maior parte dos servidores: http, ftp, etc. Remova o X11: não precisa dele! Remova o software não necessário... ou seja, praticamente tudo. Pare os demónios não utilizados. Deve obter um sistema onde o comando *ps ax* não preencha o ecrã da consola. Se utilizar o IP Masquerading, o comando *lsof −i* deve apresentar uma linha: uma que diz respeito ao servidor que está à escuta (supomos que não é uma ligação permanente).
Arbitrariamente, instalaremos o portsentry nas máquinas Linux e será lançado na altura de arranque, usando o modo "avançado" (reservado para o Linux, que é com as opções −atcp e −audp). Isto implica que o TCPWrapper e uma firewall tenham sido instalados. Mais acerca disto, posteriormente.

Para o Solaris usaremos os comandos *aset* e *ndd*. Posteriormente, mais acerca disto. O portsentry também será instalado. Podíamos adicionar IP Filter e substituir a versão padrão do RPCbind com a versão 2.1 disponível a partir de porcupine.org. Para o Irix, escolheremos o ipfilterd para a filtragem de pacotes como o nome indica. Faz parte das distribuições do Irix mas não é instalado por omissão.

Respeitante ao NT, as coisas complicam−se um pouco mais... A solução "fascista" consiste em bloquear as portas 137 e 139, que é a famosa NetBIOS (ou ainda melhor remover o NetBIOS)... mas depois nenhuma rede é vista (ou seja rede Windows) pode ser um pequeno problema quando diz respeito a um servidor de aplicações! Pode instalar o snort mas tal não o previne de tais máquinas serem mexericos. Assim, tem de ser muito restritivo acerca do acesso a partições, a directórios... desde que trabalhe com partições NTFS. Existe um programa livre disponível para se livrar da conta guest (convidado) mas o código fonte não está

disponível. Depois, instale todas as patches de segurança que encontrar! Por último mas não menos importante, proteja os seus escravos e tente−os tornar menos vulneráveis. É um pouco como ir pelo percurso de combate, mas é indispensável.

Para SOs "exóticos" terá de procurar e escolher. Como é habitual e, antes de tudo, as regras básicas devem ser aplicadas: quantos menos serviços melhor.

Protegendo a Rede

Se as máquinas foram devidamente "preparadas", está a maio do caminho. Mas precisa de avançar um pouco mais. Visto que estamos a falar de software livre escolheremos uma firewall livre para o gateway: bem é a máquina que lhe permitirá ter acesso ao mundo "selvagem". Arbitrariamente (novamente!) usamos uma máquina Linux: por isso podemos usar a firewall Bastille−Linux. Trabalha com ipchains ou ipfwadm segundo a versão do kernel. Se tiver um kernel com a versão 2.4, trabalhará com o iptables.

Uma pequena digressão: não é muito boa ideia envolver−se em todos os problemas iniciais quando a segurança é uma preocupação. A "corrida" para a última versão do kernel pode levar a uma situação muito negativa. Isto não quer dizer que o trabalho no novo kernel não esteja bom, contudo, o "casamento" com utilitários existentes, não desenhados para trabalhar desse modo pode ser um grande erro. Um conselho: Seja paciente! O novo utilitário de firewall, parte do novo kernel 2.4 é muito promissor mas provavelmente um pouco "novo". Tendo isto dito, é consigo...

Assim, o firewall do Bastille−Linux é quer simples e eficiente. Contudo, existe um utilitário muito mais elaborado, um pouco como uma "fábrica de gás", chamada T.REX. Está disponível a partir de http://www.opensourcefirewall.com. Se procura um utilitário livre e sofisticado, aqui está.

Existem outras soluções, tais como proxys, contudo nem sempre são as melhores. Uma outra digressão: os proxys são, muitas vezes, chamados de "firewalls". Mas, eles são duas coisas distintas. As firewalls de que estamos a falar usam a filtragem de pacotes e não fornecem métodos de autenticação. Existem dois tipos de servidores proxy: aplicações ou socks. Um proxy de aplicação faz todo o trabalho por si, administra toda a comunicação e permite−lhe autenticação do utilizador. É por causa disto que precisa de mais recursos que uma firewall. Mas, mais uma vez este tipo de utilitário só o protege por um pequeno período de tempo. Uma firewall pode ser "crackada" em cerca de 15 minutos. Bom saber isto, não é? Daqui a necessidade de ter as máquinas protegidas na sua rede: decidir securizar uma rede assentando num firewall ou num proxy é uma heresia!

Outro método para reduzir os riscos numa rede é a encriptação. Por exemplo, usando o telnet é como se fizesse os piratas caminharem num tapete vermelho. É um modo de lhes dar as chaves da loja. Não só como conseguem ver os dados a circular, mas ainda melhor, podem obter a password em texto limpo: simpático, não é? Assim, sinta−se à vontade para utilizar o ssh com os "ditos protocolos" (ou em vez de). Se precisar de usar o telnet (?), envie os dados através de uma ligação segura. Por outras palavras, redireccione a porta de telnet para uma segura. Encontrará mais detalhes acerca disto no artigo intitulado "Através do túnel". (LinuxFocus, May2001, article 202). (Sem ads!)

OK, tentámos melhorar a segurança, mas agora devíamos verificar o nosso trabalho. Para fazer isto, tornemo−nos "piratas", ou uma espécie de: utilizaremos os seus utilitários. Feio, não é? Nesta área também existe uma simpática colecção de programas e, mais uma vez, arbitrariamente, escolheremos dois deles: nmap e nessus. Não existe redundância, visto que, por exemplo, o segundo requer o primeiro. Estes utilitários são scanners de portas, apesar de o nessus ser um pouco mais do que isto. O Nessus informa−o de vulnerabilidades do sistema comparando os resultados do scan às vulnerabilidades da base de dados. Correndo

estes utilitários numa rede permite−lhe descobrir as fraquezas de cada máquina, independentemente do SO. Os resultados são bastante reveladores tornando estes utilitários obrigatórios. Pode encontrar o nmap em http://www.insecure.org e o nessus em http://www.nessus.org.

Desde o inicio deste artigo que estamos a falar como securizar uma rede local nas quais algumas máquinas estão abertas ao mundo exterior. O caso de um servidor de acesso à Internet (ISP), obviamente que é um pouco diferente e não entraremos em muitos detalhes. Digamos que tudo o que falámos é valido mas ainda tem de utilizar métodos mais elaborados, como as VPN (Redes Virtuais Privadas), LDAP para autenticação (por exemplo), etc. É quase outra matéria visto que as restrições são mais numerosas segundo este caso. Não falemos de sites de e−business, onde as coisas são excessivas. Sites seguros, dizem eles! Não me... diga que manda o seu número do cartão de crédito através da Internet? Se é o caso, é muito corajoso. Se consegue ler francês espreite o website. http://www.kitetoa.com, vale a pena.

Particularidades dos Sistemas

Como já mencionado, os sistemas não são iguais à frente do inimigo. Alguns têm grandes habilidades enquanto que outros são mexericos. Paradoxalmente (bem, não realmente!), os SOs livres estão entre os melhores. Os diferentes BSD's (OpenBSD, NetBSD, FreeBSD...), os diferentes Linuxes estão um pouco mais à frente quando a segurança é uma preocupação. Novamente é resultado de um trabalho excelente da comunidade de software livre. Os outros mesmo com a etiqueta do Unix, estão um pouco menos avançados. Quando não são Unix é muito pior!

Todos os utilitários mencionados neste artigo foram desenvolvidos para os SOs livres. A maior parte dos Unixes proprietários pode beneficiar deles. Contudo estes sistemas proprietários também têm os seus utilitários. Por exemplo, respeitante ao Solaris, mencionámos o *ndd* e *aset*. Apesar de uma vasta ideia, os sistemas da Sun não são modelos de segurança. Um utilitário como o aset, permite melhorar as coisas quando as permissões de segurança estão em questão. O aset oferece três níveis de protecção: baixo, médio e alto. Pode corrê−lo a partir de uma shell ou de um trabalho do cron. Numa rede a situação muda, o que é verdade às 5 da manhã pode ser falso às 5.30 da manhã. Assim o interesse em correr os comandos periodicamente para manter a homogeneidade. É por isto que o aset tem a possibilidade de ser manuseado através do cron. Verificando assim as permissões dos directórios dos ficheiros... em cada 30 minutos, em cada hora ou o que pretender,
O ndd permite alterar os parâmetros da pilha IP. Por exemplo, pode ser usado para esconder impressões digitais. Um sistema identificável é mais vulnerável, visto que os piratas sabem melhor como "atacar". Com o ndd, pode alterar o Tamanho Máximo de um segmento TCP (MSS). Por omissão, este tamanho é de 536 sobre o Solaris 2.6. O comando *ndd −set /dev/tcp tcp_mss_def 546* altera−o para 546. Quanto maior for o MSS melhor (mas não muito!). O Nmap, consegue, por exemplo, descobrir esta fraqueza. Usando o ndd corta o chão dos seus pés. Se tiver máquinas a correr Solaris sinta−se à vontade para usar o ndd. Existem muitas opções: verifique as páginas man.
Pode também utilizar o filtro IP, um utilitário de filtragem de pacotes. Está disponível em ftp://coombs.anu.edu/pub/net/ip−filter.

Respeitante ao Irix, a situação já é novamente diferente. O SGI (ex Silicon Graphics), como o nome indica, foi desenhado para gráficos. A segurança não era a principal preocupação. A necessidade de não conhecer leis tornou imperativo providenciar modos de reduzir os riscos. O ipfilterd é fornecido nas distribuições do Irix mas não é instalado por omissão: terá de o procurar! O ipfilterd é, obviamente, utilizado para filtrar pacotes permitindo−lhe negar o acesso a quem quiser. Assenta num ficheiro de configuração chamado ipfilterd.conf e é aqui que as coisas se complicam. A sintaxe deste ficheiro é por vezes peculiar e não permite espaços inesperados ou linhas vazias. Assim para permitir que uma máquina chamada "mars" fale com a máquina "jupiter" (a qual é a estação de trabalho), terá de digitar uma linha do tipo:

accept −i ec0 between jupiter mars
As máquinas não listadas neste ficheiro não conseguiram aceder a jupiter. Ainda pior: se não alterar o parâmetro *ipfilterd_inactive_behavior* usando o systune, ninguém acederá à máquina! Eficiente, não é? Este parâmetro tem o valor de 1, por omissão e precisa de modificar para 0 com o comando *systune −i ipfilterd_inactive_behavior 0*.
Uma outra coisas muito conhecida, melhor recordas, o Irix tem uma "grande" vulnerabilidade, chamada fam (File Alteration Monitor). Este programa está encarregue de uma funcionalidade muito interessante, fazer a comunicação entre vários demónios. Por exemplo, é o que permite que se obtenham ícones bonitos no seu explorador de ficheiros. Contudo, só existe uma coisa a fazer: desactive−o! Triste mas verdade.

Para terminar com os sistemas Unix, mencionemos que o QNX é muito vulnerável mas pode beneficiar os utilitários livres. O Mac OS X fornecer alguns destes utilitários.

Devemos falar um pouco acerca da referência absoluta entre os sistemas de rede: o único e solitário NT 4.0. Securizar isto é um ponto de vista utópico, apesar do que o rei de Redmond (e muitos outros) dizem. Simular um ataque com o nessus, por exemplo é um pesadelo. Desde que o NetBIOS esteja activo o nessus fornecer−lhe−á os nomes de cada máquina no dominio e com os seus utilizadores correspondentes, incluindo os administradores. A resposta é: livre−se do NetBIOS! Correcto, mas como já mencionado, sem NetBIOS não há rede... Terá de escolher o seu lado.
O nessus, gentilmente, o informa que pode fazer login como convidado (guest) com uma sessão NULA (NULL) (ou seja um utilizador nulo e uma palavra−passe nula). E no fim removê−lo. Sim, mas como? E é tudo assim!
Assim, reduza o acesso às partições (NTFS), aos directórios. Para as partições FAT... não há solução. Contudo, segundo o software pode precisar das partições FAT: algum software não trabalha em NTFS. Para terminar isto, evite o grande IIS, especialmente como servidor de ftp. De facto não o instale. Se hoje em dia, muitos ISP são suficiente malucos para o usar, nós só podemos sugerir que use o Apache como alternativa... Não desperdicemos o nosso tempo com o IIS, existe muita documentação acerca da matéria.
De facto, existe um modo de tornar o mexerico num filtro (os buracos são menores!). O problema é que é um longo caminho a percorrer e nem toda a revista seria suficiente. Mencionemos só o mais importante. Obviamente, que o objectivo não é securizar com utilitários de software livre: estamos a falar do mundo da Microsoft! A primeira sugestão é usar o MSCE (Microsoft Security Configuration Editor) disponível a partir do ServicePack 4 com a MMC (Microsoft Management Console). Contudo, seja muito cuidadoso! Se cometer um erro, ganhou. Claro que isto é uma versão inglesa. Se tiver uma versão estrangeira do sistema (não Inglesa), seja avisado que a mistura de línguas num deu muitos bom resultados no mundo da Redmond. Foi avisado. De seguida, entre as várias medidas de segurança, deve "securizar" a conta do administrador, ou até mesmo desactivá−la. Dê uma vista de olhos pelo *passprop* disponível a partir do SP3. Pode também proteger as palavras−passe utilizando o dll *passfilt* através do registo (sempre pensei que as pessoas que inventaram isto estavam sobre a influência de LSD...). Desactive a famosa conta de convidado. Não é muito útil (veja acima), mas melhora as coisas. Mas, pode restringir o seu acesso aos logs a partir do registo. Em "HKEY_LOCAL_MACHINE", crie as chaves *System\CurrentControlSet\Services\EventLog\Application*, Segurança e Sistema (estas duas últimas devem substituir Aplicação). O seu nome é "RestrictGuestAccess", o tipo é REG_SZ e o valor é 1. Pode encriptar as palavras−passe com *syskey*. Cuidado, é uma operação irreversível! Pelo menos, algumas boas notícias: pode restringir o acesso da conta convidado. Brinquemos novamente com o registo, ainda em "HKEY_LOCAL_MACHINE". Desta vez a chave chama−se *System\CurrentControlSet\Control\Lsa*. O nome é "RestrictAnonymous", e o tipo é "REG_DWORD" e o valor é 1. Contudo, o mundo da Microsoft é maçador: fique avisado que esta alteração pode modificar alguns serviços de rede... Entre as coisas mais importantes, pode restringir o acesso a algumas portas, utilizando a aplicação de rede no painel de controle. Nas propriedades do TCP/IP seleccione "Advanced" e verifique a caixa "Active security" (acredito que é este o seu nome, mas eu não tenho este tipo de coisa em casa para poder verificar). Da janela "Security", verifique "Allow only" e seleccione as portas que quer activar. Aqui, tenha precaução também. Deve saber o que está a fazer, caso contrário alguns serviços não trabalharam mais. Muito mais pode ser feito, mas isto é o essencial. Para aprender mais, pode visitar o sans.org: toneladas de

documentos disponíveis.

A insuportável leveza das coisas

Bem, fez tudo isto. Pode correr o nessus para verificar toda a rede e ainda obter buracos na segurança. Não iremos dizer de onde vêm... Nós, realmente sabemos! Tente iludir estes sistemas substitutos. Não removerá os buracos do NetBIOS, mas limitará os estragos. Crie sub−domínios. Não entre como administrador. Aplique as actualizações. Por último, tente esconder todas as máquinas por detrás das máquinas Unix usadas como gateways. Infelizmente, a relatividade da segurança, não vem somente dos produtos feitos na Redmond. Um rede está viva: existe sempre algo a passar−se. Um bom administrador é "paranóico", assim, verifique, frequentemente o "inventários de correcções". Escreve scripts para automatizar as verificações. Por exemplo para controlar, regularmente, os programas SUID/SGID, os ficheiros críticos e os logs... Para ter mais alguns amigos, bloqueie as floppies dos utilizadores ou os dispositivos CDROM. Não permita que os seus utilizadores façam download de software sem o seu consentimento, especialmente quando se tratam de executáveis, como no Mundo da Microsoft. Previna os seus utilizadores de abrir os documentos em anexo, especialmente os de formato Word ou Excel recebidos por mail. Sim, eu sei que é fascismo, mas o que é que podemos fazer contra os vírus em macros? Não usar produtos como o Outlook. Mais uma vez deve saber o que quer! Eu sei que o que digo é inútil, mas podemos falar de segurança em tais produtos? O famoso "I love you" não lhe ensinou a lição.
Respeitante ao Unix, os download devem também ser controlados. Os checksums não foram inventados por acidente.

Ganhe o hábito de controlar a sua rede regularmente, através dos logs, de scripts, scans... Verificará: que as coisas mudam bastante rápido e nem sempre no bom sentido.
Por último, não dissemos ainda uma palavra acerca do assunto, não se esqueça das cópias de segurança. A estratégia é não modificar: diariamente, semanalmente ou mensalmente. Uma máquina Unix também pode ter problemas, mesmo que seja pouco usual. E, por vezes, os utilizadores cometem erros... mas não frequentemente. É bem sabido que os problemas vêm das máquinas ou dos departamentos encarregues por elas:−(

Pelo menos, está terminado!

Se chegou a esta secção então é corajoso. O problema é que só passámos ao de leve pela matéria! A segurança não tem fim e não diz respeito somente às redes. Aplicações vulneráveis podem comprometer a rede. Uma firewall mal configurada é mais perigosa que não haver firewall. Uma máquina Unix, muitas vezes, guarda milhares de ficheiros. Quem é que pode ter a certeza que nenhum é vulnerável? Quem é que pensa que um pirata tentará quebrar uma chave de 128 bits? Não seja doido: ele tentará encontrar uma porta por detrás da casa. Novamente, pode instalar todos os utilitários de segurança disponíveis, mas se deixar um pequeno buraco é aqui que o "mau" entrará.

A segurança é também um comportamento: siga o que se está a passar. Por exemplo visite os websites de segurança numa base regular, o mesmo para sites dos editores dos seus SOs... Por exemplo a Sun, publica todos os meses as actualizações recomendáveis. A Microsoft, frequentemente, fornece ServicePacks ou Correcções. Os distribuidores de Linux publicam erratas para cada nova vulnerabilidade descoberta. O mesmo se aplica para os diferentes BSD's. Se não utiliza os produtos correspondentes a uma actualização remova−os do seu disco rígido. E por aí adiante: a lista de coisas a fazer é muito longa. Em suma, este trabalho não deve conhecer demissões.

Por último, digamos isto novamente, tudo isto contribui para a sua rede ser menos vulnerável. Não espere obter uma rede 100% segura, mesmo num determinado tempo (bem, talvez, se todas as máquinas forem paradas). Tendo isto dito, não é um requisito ser paranóico para fazer este trabalho... mas ajuda! Mas não o seja no dia−a−dia da sua vida, será mais simpático para as pessoas à sua volta...

Referências

- http://www.linuxsecurity.com
- http://www.sans.org
- http://www.infosyssec.org
- http://www.securityfocus.com
- http://www.cs.purdue.edu/coast/hotlist/

A vida é triste: vamos é ter algum divertimento!

Um outro modo de fazer o trabalho;−)

Webpages maintained by the LinuxFocus Editor team © Georges Tarbouriech "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org	Translation information: fr −−> −− : Georges Tarbouriech <georges.t(at)linuxfocus.org> fr −−> en: Georges Tarbouriech <georges.t(at)linuxfocus.org> en −−> pt: Bruno Sousa <bruno(at)linuxfocus.org>

2005−01−10, generated by lfparser_pdf version 2.51