Criando túneis criptografados com OpenVPN

Segredos sem fio

As redes sem fio são tão práticas quanto perigosas. A criptografia WEP não parece ser, hoje, algo que impeça um invasor de entrar. Mas a solução está bem à mão: basta criptografar sua rede com um túnel OpenVPN.

por Achim Leitner

Quadro 1: Apenas acesso à Internet

O cenário mais simples possui um ou mais computadores isolados uns dos outros, que usam um ponto de conexão sem fio unicamente para acesso à Internet (ver **figura 1**). Além dos riscos que já afetavam as redes de cobre, precisamos considerar novas pragas como seqüestro de conexão, negação de serviço, *wardriving* ou vizinhos bisbilhoteiros. Os usuários mais paranóicos a respeito de invasões tendem a encarar uma rede sem fio como um vetor de invasão muito fácil de explorar. Entretanto, ataques a esmo podem vir tanto da selvageria da Internet quanto das poucas centenas de metros entre seu ponto de acesso WLAN e seu laptop.

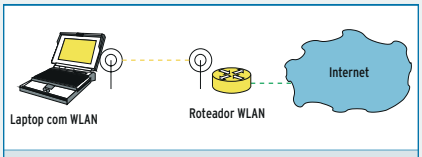

Figura 1: Conexão à Internet com um roteador WLAN.

Na verdade, é bastante saudável ser ligeiramente paranóico a respeito da segurança de qualquer tipo de informação. A única forma de se certificar de que seus dados e informações não foram manipulados é usar a criptografia a seu serviço, embaralhando o conteúdo e proibindo o acesso não autorizado. Em outras palavras, use **SSL/TLS** para baixar conteúdo da web e habilite SSL para seu programa de email. O SSL criptografa e autentica seus dados durante a transmissão. Para adicionar mais uma camada de proteção, use os padrões **PGP** ou **S/MIME**. Ambos criptografam as mensagens de email antes do envio, em vez de criptografar o tráfego já pronto. Para terminar, use **SSH** sempre que precisar fazer logins remotos.

A tecnologia das WLANs é essencialmente insegura – e muitas pessoas já começam a perceber isso. A criptografia embutida nelas é facílima de quebrar – ou mesmo de ser desabilitada nos casos mais graves. Enquanto um cracker malévolo precisaria entrar em sua casa para "farejar" a rede se ela for de cobre, qualquer um na rua pode caminhar por aí com um laptop e uma placa de rede WLAN e se conectar à sua rede sem fio. Amplificadores de sinal (as conhecidas "butinas") e antenas especiais estendem o alcance do sistema para algumas centenas de metros. Assustador, não?

Apesar do risco, as redes sem fio vieram para ficar. A possibilidade de navegar pela Internet com seu laptop estando na cama, na sacada ou no jardim é ótima. Ver suas receitas com o notebook na cozinha é especialmente tentador. Mesmo copiar arquivos do seu PC de mesa para o laptop sem precisar de cabos é algo espetacular. Tudo isso é realmente muito bom, desde que mantenhamos algumas regras básicas de segurança que diminuem o risco inerente às tecnologias sem fio.

Protegendo a rede

Antes de decidir que tipo de proteção você vai querer para seu ambiente, é preciso examinar mais de perto a forma como seus computadores estão ligados em rede e o tipo de tráfego que os links sem fio estarão transportando. A proteção embutida na própria tecnologia WLAN costuma ser suficiente para a maioria dos usuários; alguns até mesmo desabilitam toda e qualquer segurança. Se você, pelo contrário, precisa de mais do que o WEP pode oferecer, os protocolos de VPN (redes virtuais privadas) como o OpenVPN [1] são uma boa escolha – simples de usar, mas atuais e bem seguros. O OpenVPN criptografa e autentica as trocas de dados entre dois computadores quaisquer, estejam eles usando Linux ou Windows®.

Fora das muralhas de sua rede doméstica ou empresarial, a Internet é repleta dos mesmos tipos de risco que uma WLAN. Os potenciais invasores podem "farejar" seu tráfego e manipular seus dados; até mesmo a injeção de conteúdo malicioso é possível. Precisamos, portanto, distinguir entre dois casos:

- PCs e laptops que usem a rede sem fio e o roteador WLAN apenas para acesso à Internet (ver **quadro 1**);
- Redes domésticas ou empresariais em que a WLAN é usada para estender, ou mesmo substituir, uma rede com cabos (ver **quadro 2**).

Aproveitadores e malfeitores

As redes WLAN introduziram uma nova categoria de risco, até então nunca registrada em redes tradicionais: pessoas completamente estranhas à organização proprietária da rede podiam usar os pontos de acesso sem fio para obter acesso *gratuito e irrestrito* à Internet. A extensão do dano que isso pode causar depende de tipo de contabilidade que seu provedor usa para cobrar de você o acesso. Se o preço é fechado, não há muito problema em deixar seu vizinho navegar às suas custas. Mas se seu provedor de acesso tiver colocado um

"taxímetro" na sua conexão – ou seja, se cobra por tempo de conexão ou por volume de dados, coisa muito comum hoje em dia – compartilhar a conexão pode ser verdadeiramente desastroso para sua saúde financeira. Uma maneira de impedir que isso ocorra é filtrar os endereços MAC no seu roteador WLAN e usar criptografia WEP.

Nenhuma dessas medidas vai garantir uma proteção perfeita, mas a imposição de um obstáculo extra a ser transposto pode afastar uma grande quantidade de candidatos a malfeitores – ninguém mais poderá usar a velha desculpa do "ops, me enganei de rede" ou do "ah, pensei que podia". Certifique-se de que os filtros de MAC e o WEP estejam habilitados o tempo todo. A não ativação desses recursos simples é um convite aberto para que aproveitadores, espiões e crackers usem às suas custas sua estrutura de rede e acesso.

Mais proteção significa bastante suor, já que o trabalho envolvido é complexo e em quantidade. Felizmente, um sucessor para o WEP já foi anunciado. Em junho de 2004, o IEEE introduziu um padrão de conexão mais seguro chamado 802.11i, também conhecido como WPA-2. Infelizmente, essa tecnologia é restrita a novos adaptadores e há ainda muita confusão a respeito de sua implementação. O novo padrão especifica um certo número de técnicas, mas nem todas são seguras. Entre as recomendações estão o formato AES-CCMP para criptografia e 802.1x para autenticação e gerenciamento de chaves.

Protegendo-se com uma VPN

No Linux, podemos montar uma rede sem fio segura e livre de aproveitadores sem ter que comprar uma nova placa de rede. Se seu hardware não oferece o tipo de proteção de que você precisa, a resposta está no software. O protocolo VPN (*Virtual Private Network* ou Rede Privada Virtual) criptografa e autentica dados na camada IP. Um ponto de conexão VPN recebe, criptografa e assina os dados para transmiti-los pelo link de rádio. Do outro lado, o outro ponto de conexão descompacta os pacotes que recebeu e os entrega a seus destinos.

Uma VPN usa os recursos de uma rede sem fio mas se parece, do ponto de vista das estações, com uma rede adicional – uma rede virtual. A **figura 2** explica o princípio: o laptop e a estação de trabalho possuem uma conexão WLAN. Ambos são acessíveis pelos seus endereços IP reais na rede sem fio. ➡

Quadro 2: A WLAN em uma rede empresarial

Uma rede doméstica ou uma pequena rede empresarial com um grupo de computadores a proteger é algo bem mais complexo que um ou dois computadores que não se conectam entre si e apenas acessam a Internet – cenário discutido no **quadro 1**. Redes mais complexas tipicamente usam um firewall para proteger a conexão à Internet; os usuários costumam se sentir seguros atrás de um deles. Os firewalls normalmente previnem contra conexões não autorizadas vindas de fora para dentro da rede. Entretanto, esse clima acolhedor pode levar as pessoas a baixar a guarda.

Os perigos potenciais incluem servidores NFS ou Samba com compartilhamentos que não precisam de senhas, servidores de impressão que transferem dados sem proteção e sistemas com login via *Telnet* ou *rlogin*. Considera-se que todos os computadores ligados na rede confiam nela e uns nos outros, além de ter uma devoção quase fanática pelos poderes do firewall.

Relações de confiança são coisas perigosas em redes tradicionais, mas muito mais que isso em redes sem fio. Um invasor que esteja do lado de dentro de um firewall pode iniciar ataques internos. Em uma rede tradicional com cabos, um espião ou sabotador precisaria estar presente fisicamente em sua casa ou departamento para iniciar um ataque. Com uma WLAN, o invasor precisa estar apenas na vizinhança – cabos e tomadas na parede não são mais necessários.

A única maneira de proteger conexões sem fio contra invasões é usar criptografia. A primeira tentativa de se normatizar a criptografia para WLANs falhou vergonhosamente: o **WEP** é facílimo de quebrar e não faz o serviço direito. Entretanto, uma **VPN** permite que se acrescente uma camada extra de proteção.

Glossário

SSH: *The Secure Shell* (shell seguro) permite que usuários de Linux acessem computadores remotos com segurança. Toda a sessão, incluindo a troca de senhas, é criptografada. Substitui o velho e inseguro *Telnet*.

PGP: O *Pretty Good Privacy* (livremente, algo como *privacidade danada de boa*) é usado para criptografar e assinar digitalmente mensagens de email. O OpenPGP é a implementação padrão e o GnuPG uma alternativa mais recente.

S/MIME: *Secure/Multipurpose Internet Mail Extensions* (Extensões de Correio Eletrônico Multipropósito, com camada de criptografia) é uma outra maneira de criptografar e assinar digitalmente mensagens de email, uma alternativa ao PGP/GnuPG.

SSL/TLS: O *Secure Sockets Layer* (camada de conexão segura) é um protocolo de criptografia criado pela Netscape. O SSL é um método testado e aprovado para criptografar transmissões de dados. O *Transport Layer Security* (segurança na camada de transporte) é um aprimoramento do processo construído sobre o SSL.

VPN: *Virtual Private Network* (Rede Privada Virtual). Usa uma rede real pré-existente para emular uma segunda rede, esta virtual. O software de VPN criptografa o tráfego antes de enviá-lo pelo cabo.

WEP: *Wired Equivalent Privacy* (Privacidade Equivalente a das redes com cabos) foi a primeira tentativa dos fabricantes de dispositivos WLAN para criar um padrão de protocolo seguro. Usando criptografia, esperava-se obter o mesmo nível de proteção que nas redes com cabos. Quase imediatamente após o lançamento, ficou claro que o protocolo era constrangedoramente falho e inseguro.

Endereços privados: Os endereços IP são únicos no mundo todo. É a única maneira de assegurar que todos os pacotes IP encontrem o caminho de casa. Em contrapartida, os endereços privados são válidos apenas na rede local e não são roteados na Internet pública. Isso permite que muitas redes particulares que não se conectam entre si usem os mesmos endereços privados. Para uso particular, os endereços que devem ser usados são: 10.x.x.x (classe A), de 172.16.y.y até 172.31.y.y. (classe B) e 192.168.z.z (Classe C).

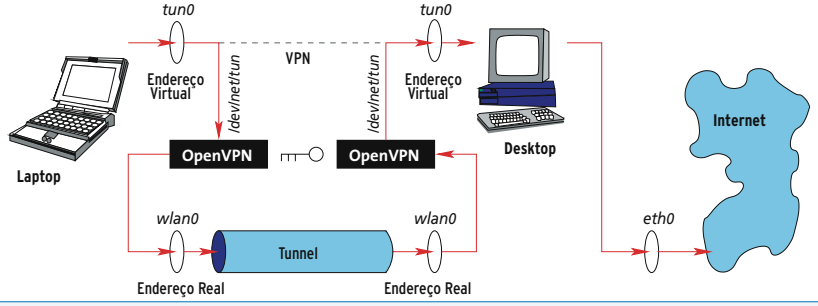

Figura 2: A VPN funciona por tunelamento. Nas extremidades do túnel há pontos de acesso com os endereços IP reais dos nós envolvidos.

A VPN dá aos computadores endereços IP adicionais. Os dados endereçados aos endereços virtuais são encapsulados pela VPN e enviados ao endereço IP real da máquina que deve receber o pacote, que por sua vez descompacta os dados e os trata como se tivessem chegado pelo endereço virtual. Com isso, criamos um túnel virtual dentro da conexão real entre o laptop e o PC.

Regras adicionais no firewall asseguram que ambas as máquinas aceitarão apenas os dados que chegam pelo túnel. Isso significa que os dados injetados diretamente na WLAN por um invasor serão inócuos, pois todas as estações e servidores envolvidos vão ignorá-los.

OpenVPN

O princípio de funcionamento das VPNs é usado por muitos protocolos, produtos e projetos. O OpenVPN [1] é uma abordagem testada e aprovada que se mostra estável e simples, funcionando sem a necessidade de mexer com o kernel ou com as pilhas de rede IP. Como o programa é baseado no protocolo de criptografia TLS, além de possuir uma implementação eficiente e enxuta, o OpenVPN construiu para si uma reputação de extremamente seguro.

Nas duas extremidades do túnel, o OpenVPN recolhe os pacotes de dados destinados à outra ponta e usa uma chave local para criptografar os pacotes antes do envio. Do outro lado do túnel, os dados são descriptografados e sua origem é verificada. A extremidade do túnel em que os dados foram recebidos aceitará apenas pacotes criptografados com a chave correta. Quaisquer outros pacotes, criptografados ou não, serão sumariamente ignorados. Com esse esquema, os pacotes são transportados em "embalagens" seguras e invioláveis, mesmo em ambientes extremamente hostis.

O exemplo a seguir considera `wlan0` como a conexão sem fio. O PC possui, ainda, uma placa de rede Ethernet comum, com um cabo CAT5, chamada `eth0`. As outras máquinas na rede e a Internet estão acessíveis pela conexão Ethernet (ver **figura 2**).

O procedimento simples descrito nas seções a seguir supõe o uso de IPs estáticos. Os computadores precisam de endereços que não mudem a cada reinício.

Para começar...

Se ainda não o tiver feito, instale o pacote do OpenVPN antes de qualquer coisa (ver **quadro 3**). O OpenVPN não modifica seu kernel. Em vez disso, para rotear os pacotes de dados, ele usa o driver TUN/TAP [4]. A maioria das distribuições Linux instala esse módulo do kernel por padrão, portanto basta carregá-lo na memória. Se emitido como *root*, o comando a seguir faz exatamente isso:

```
modprobe tun
```

O Linux normalmente não usa arquivos de dispositivo para interfaces de rede; trocando em miúdos, não há um `/dev/eth0`. Isso pode parecer inconsistente, mas ele não é necessário porque a interface de *sockets* lida com a parte de comunicação. A interface TUN se aproveita disso e quebra as regras criando um arquivo de dispositivo, que permite que um *daemon* rodando no espaço do usuário sorrateiramente capture os pacotes IP, reempacote e mande para a interface virtual.

O *daemon* grava os dados no arquivo `/dev/tun0` ou `/dev/net/tun` (ver **quadro 1**) e eles chegam ao kernel pela interface `tun0`. Cada pacote que passa por `tun0` chega ao kernel via `/dev/tun0` ou `/dev/net/tun` (ver **figura 2**). A interface funciona da mesma forma que qualquer interface de rede; é possível associar um endereço IP a ela, usá-la para roteamento e aplicar regras de firewall. A única diferença é que não se usa uma placa Ethernet para colocar os dados no cabo; em vez disso, usamos um dispositivo para enviar esses mesmos dados a um *daemon*.

O OpenVPN e as chaves de criptografia

O OpenVPN precisa de chaves de criptografia para implementar segurança nas comunicações. No caso mais simples, os usuários dos dois computadores de nosso exemplo compartilham um segredo. O comando a seguir cria uma chave e a armazena em um arquivo chamado `chave.secreta`:

```
openvpn --genkey --secret chave.secreta
```

Apenas as duas máquinas devem conhecer a chave, e apenas o root deve ter acesso a elas – quem conhece a chave pode facilmente penetrar no túnel. É muito importante assegurar que a chave não seja "farejada" por algum marginal enquanto está sendo copiada entre as duas máquinas. Lembre-se: alguém pode ter "grampeado" seu link sem fio. O melhor a fazer é copiar a chave em um disquete

Quadro 3: Instalação

O OpenVPN é bem fácil de instalar. Provavelmente sua distribuição já o tem empacotado. Para os corajosos, o código fonte para a versão estável 1.6.0 está disponível no site oficial do projeto [1]. Os comandos a seguir descompactam o pacote, compilam o software e o instalam com privilégios de root.

```
tar -xvzf openvpn-1.6.0.tar.gz
cd openvpn-1.6.0
./configure --disable-lzo
make
su
make install
```

Para desabilitar a compressão de dados, rode o comando config com o parâmetro `--disable-lzo`. Como os dados não podem ser comprimidos depois da criptografia, essa biblioteca é recomendada para conexões muito lentas. A biblioteca está disponível em [2]. O que você certamente precisa é da biblioteca OpenSSL e seus arquivos de desenvolvimento. No SuSE, instale os pacotes `openssl` e `openssl-devel`. No Debian, ambos estão disponíveis no sistema APT. Outras distribuições também incluem os pacotes *OpenSSL*; consulte a documentação oficial para saber como instalá-las. Quando tudo o mais falhar, consulte o site oficial do projeto OpenSSL [3] para mais informações sobre o programa, como obtê-lo e componentes associados.

O kernel atual possui nativamente um dispositivo de túnel; o pacote está disponível em [4] para versões mais antigas. Se você quiser compilar seu próprio kernel, o módulo TUN está localizado na seção *Network device support* opção *Universal TUN/TAP device driver support* do configurador `make xconfig`. Claro que você pode, se quiser, apenas compilar o módulo sem ter que substituir o kernel todo. Depois de configurar o Kernel do Linux, digite:

```
make modules
make modules_install
```

O próximo passo é criar um arquivo de dispositivo em `/dev/net/tun`. Se o diretório `/dev/net/` não existir, crie-o com `mkdir /dev/net/` e, só depois, crie o dispositivo:

```
mknod /dev/net/tun c 10 200
```

– e não se esqueça de formatá-lo depois que terminar! Se os programas *OpenSSH, PGP, GnuPG* ou qualquer outro semelhante já estiverem instalados, é uma boa idéia usá-los também.

Cavando o túnel

A próxima etapa é criar o túnel. Para isso, o OpenVPN precisa do endereço IP *estático* da máquina de destino, o nome do dispositivo de tunelamento (o padrão é `tun0`), os dois endereços IP virtuais para a VPN e o arquivo contendo a chave de criptografia. No laptop, digite o comando abaixo:

```
openvpn --dev tun0 --remote ➥
[IP_Real_PC] --ifconfig ➥
[IP_Virtual_Laptop] [IP_Virtual_PC] ➥
--secret chave.secreta
```

Rode os comandos como *root*. No PC de mesa, o comando com o endereço IP modificado é:

```
openvpn --dev tun0 --remote ➥
[IP_Real_Laptop] --ifconfig ➥
[IP_Virtual_PC] [IP_Virtual_Laptop] ➥
--secret chave.secreta
```

Os endereços virtuais do túnel são mais ou menos arbitrários, mas eles necessariamente têm de ser **endereços privados**. Os endereços virtuais devem estar em uma classe diferente dos reais para facilitar o roteamento e tornar fácil a diferenciação entre as redes virtuais e as reais.

Atribuindo endereços

Vamos considerar que o endereço real na placa de rede WLAN do laptop seja 172.16.0.1. No PC, esse endereço é 172.16.0.2. A VPN precisa de endereços privados para suas interfaces virtuais, como 10.0.0.1 para o laptop e 10.0.0.2 para o PC. Neste caso, o comando no laptop é o mostrado abaixo:

```
openvpn --dev tun0 --remote 172.16.0.2 ➥
--ifconfig 10.0.0.1  10.0.0.2 ➥
--secret chave.secreta
```

E no PC devemos digitar:

```
openvpn --dev tun0 --remote 172.16.0.1 ➥
--ifconfig 10.0.0.2  10.0.0.1 ➥
--secret chave.secreta
```

Quadro 4: As funções do OpenVPN

Além da VPN simples mostrada no exemplo, do tipo "cliente-para-rede", o OpenVPN pode também conectar duas redes completas, bastando para isso mudar a configuração de roteamento. Em modo de *bridging*, o OpenVPN pode conectar duas seções de uma LAN de modo transparente, permitindo que usem o endereçamento de uma mesma rede.

A solução com chave compartilhada descrita no artigo não é uma boa solução se a VPN possuir muitos nós. Mas essa é a melhor característica do TLS: ele pode trabalhar com chaves públicas X.509. A versão 2.0 do OpenVPN (ainda em beta) torna as coisas ainda mais simples para os administradores: não é necessário criar uma configuração no servidor para cada cliente da VPN; tudo o que se precisa são certificados X.509 válidos. Além disso, o novo servidor deve ter um desempenho muito superior sob carga pesada.

Trabalhando em modo UDP, o OpenVPN não distingue entre clientes e servidores; em vez disso, trabalha de forma parecida com a das redes ponto-a-ponto. A opção `--float` permite que o túnel continue funcionando mesmo que o IP na outra ponta seja alterado – por exemplo, quando é preciso "resetar" as placas de rede à força. As conexões TCP são mantidas ativas até que sejam normalmente encerradas, o que é útil quando se precisa transferir arquivos enormes via FTP.

Se for preciso enviar arquivos gigantescos pelo túnel, uma dica é habilitar a opção `--shaper [bandwidth]`, que restringe a velocidade de entrada do túnel a uma quantidade específica de bytes por segundo. Para restringir a banda passante em ambas as direções, basta especificar essa restrição em ambos os lados. O OpenVPN pode abrir mais de um túnel entre dois pontos ao mesmo tempo e determinar diferentes bandas passantes para cada um deles – o que é muito útil para tarefas administrativas. Para ajustar o tipo de dado que passa por cada túnel, configure a tabela de roteamento dos sistemas envolvidos.

As versões 1.5 e posteriores do OpenVPN também trabalham com TCP. Se algum dos nós estiver protegido por um firewall que bloqueie qualquer coisa que não seja TCP, essa pode ser a única alternativa. A desvantagem é clara: se algum problema ocorrer na rede, a combinação VPN-sobre-TCP tornará as coisas bem piores. Sempre que possível, configure o OpenVPN para usar o tradicional UDP.

Use o *ping* para se certificar de que o túnel está funcionado. No laptop, digite `ping 10.0.0.2`; isso deve funcionar e indicar que o endereço virtual para o PC pode ser alcançado.

Se tudo correr como planejado, podemos rodar o *daemon* do OpenVPN em segundo plano; o *daemon* irá registrar suas mensagens no *syslog*. Use a opção `--daemon` para habilitar o suporte ao *syslog*. Observe que é preciso especificar o caminho completo para o arquivo que contém a chave secreta.

O caminho do bem

Seu túnel está rodando sem problemas e os pacotes estão saindo pelo outro lado. Mas tanto o laptop quanto o PC precisam saber quais pacotes devem rotear pelo túnel. Em outras palavras, é preciso conhecer o endereço IP virtual na outra ponta. O OpenVPN faz isso sozinho, mexendo nas rotas para refletir os novos endereços. Todos os outros endereços são roteados como antes – contornando o túnel.

A rota do PC até o laptop deve funcionar perfeitamente se usarmos os endereços virtuais. Os endereços antigos devem ser usados *apenas* como pontos de entrada para o túnel. Nenhum outro tipo de tráfego deve passar por eles além desse.

Já o caminho do laptop de volta para o PC – e de lá para qualquer outro computador na rede ou na Internet – precisa de alguma edição manual. A rota padrão precisa ser reconfigurada. Os comandos a seguir dizem ao laptop para enviar todo e qualquer pacote pelo túnel e não por outras rotas:

```
route del default
route add default gw 10.0.0.2
```

Os pacotes endereçados ao endereço real do PC (172.16.0.2) não são afetados por essa configuração. Isso é desejável, já que o túnel usa esses endereços. Precisamos, então, informar ao PC que ele deve redirecionar os pacotes recém-decodificados, se necessário. O comando a seguir cuida disso:

```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Brigada de incêndio

Estamos quase lá em ambos os lados. O laptop e o PC já usam alegremente o túnel da VPN; seus dados estão protegidos e ninguém consegue farejá-los. Mas ainda é possível injetar tráfego malicioso na rede "real", permitindo que o invasor possa, ao menos, navegar na Internet às

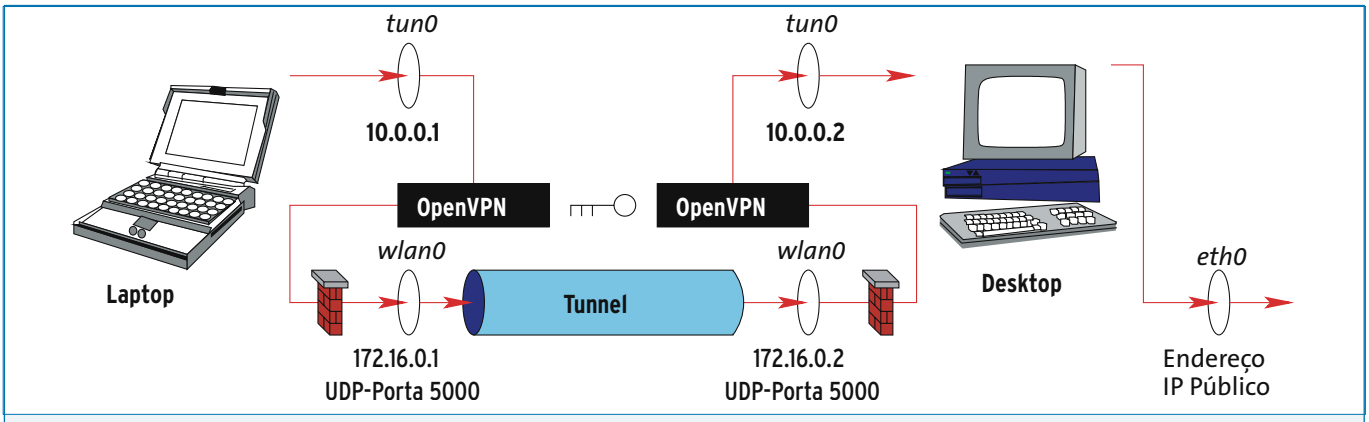

Figura 3: As regras de firewall evitam que invasores entrem em sua WLAN. Apenas o túnel OpenVPN pode ser usado para troca de dados entre estações.

suas custas. Mesmo que isso não cause impacto na sua fatura do provedor no fim do mês, talvez você não queira compartilhar sua banda com ninguém. Todos os serviços de rede que estejam disponíveis no laptop e no PC (como servidores Web, SSH, FTP e Samba, entre outros) podem ser usados e atacados pelo invasor por meio da rede sem fio. Por último, mas não menos importante, quaisquer pacotes injetados diretamente na WLAN contornaram o firewall, que reside entre sua rede e a Internet.

A distribuição do OpenVPN possui um script de firewall de exemplo [1]. Mas é preciso algumas regras a mais para a nossa combinação de WLAN com túnel. A **figura 3** mostra em que lugar essas regras se aplicam.

Cai fora!

O OpenVPN usa o protocolo UDP para enviar pacotes criptografados para a porta 5000, no outro lado do túnel. Como ele usa a interface `wlan0` para isso, é preciso permitir que a porta UDP 5000 aceite conexões. A regra a seguir faz com que os pacotes entrantes sejam aceitos por essa porta:

```
iptables -A INPUT -i wlan0 -p udp ➥
--dport 5000 -j ACCEPT
iptables -A INPUT -i wlan0 -j DROP
```

A última linha assegura que o computador não aceitará qualquer outro pacote a não ser os da WLAN. A primeira regra pode restringir ainda mais o que é permitido na interface, se especificarmos a opção `-s Endereço_IP_Real` para testar o endereço IP de origem. Em nosso caso, o endereço IP real do outro lado do túnel, ou seja, `-s 172.16.0.2` para o laptop (esse é o endereço do PC). Precisamos, ainda, de restrições no envio e redirecionamento de pacotes:

```
iptables -A OUTPUT -o wlan0 -p udp ➥
--dport 5000 -j ACCEPT
iptables -A OUTPUT -o wlan0 -j DROP
iptables -A FORWARD -i wlan0 -j DROP
```

As extremidades do túnel só redirecionam pacotes originados por endereços conhecidos, e apenas se o computador nesse endereço tiver usado a chave de criptografia correta. Isso significa que você pode confiar, aceitar e processar pacotes vindos de um dispositivo `tun`. Obviamente, ainda precisamos permitir que as máquinas enviem pacotes pelo túnel. Os comandos a seguir liberam esse tipo de tráfego:

```
iptables -A INPUT -i tun0 -j ACCEPT
iptables -A OUTPUT -o tun0 -j ACCEPT
```

Isso é tudo o que precisamos para o laptop; ele não se conecta a outras redes nem tem de redirecionar pacotes.

Por aqui, cavalheiro

O PC precisa de uma regra de redirecionamento, bem como de mascaramento de IP para que o laptop consiga navegar mundo afora (observe: o PC é quem tem a conexão com a Internet):

```
iptables -A FORWARD -i tun0 -j ACCEPT
iptables -t nat -A POSTROUTING -o eth0 ➥
-j MASQUERADE
```

As regras de mascaramento fazem com que o PC empreste seu endereço IP público aos pacotes vindos do laptop pelo túnel. Os endereços privados, como vimos, não são roteados na Internet, mas com esse arranjo o PC auxilia o laptop na navegação, roteando os pacotes vindos do túnel para a sua própria interface de rede e, dela, para a Grande Rede. Se o PC usa um modem ADSL para conexão à Internet, substitua `eth0` por `ppp0`.

Limitações de Segurança

Uma rede é tão segura quanto o computador mais vulnerável conectado a ela. Uma pessoa não autorizada que tenha acesso físico ao laptop rodando o OpenVPN pode ler a chave, gravá-la num disquete e usá-la para entrar na rede. Resumindo: computadores portáteis e dispositivos sem fio precisam de uma proteção muito maior contra roubo e acesso não autorizado do que os equipamentos comuns.

Com essas regras básicas de segurança, vemos que o OpenVPN oferece uma solução de VPN segura e relativamente fácil de usar. Em vez de tentar consertar o que não presta, o OpenVPN simplesmente contorna a risível criptografia WEP usada por padrão nas redes sem fio, implementando seu próprio esquema de forma independente. Com isso, garantimos segurança confiável, o conforto de uma rede sem cabos para estorvar nosso dia-a-dia e, principalmente, muito menos dores de cabeça para os administradores. ■

Informações

[1] OpenVPN: http://openvpn.sourceforge.net/
[2] Biblioteca LZO: http://www.oberhumer.com/opensource/lzo/
[3] OpenSSL: http://www.openssl.org/
[4] Driver para o TUN/TAP: http://vtun.sourceforge.net/tun/