

Protegendo seus dados

Segurança de ponta a ponta

Quando os pacotes deixam o seu computador e passam a trafegar em redes públicas, nunca se sabe o que pode acontecer com eles. Sempre que a rede de um conjunto empresarial é baseada em *hubs*, qualquer computador conectado ao mesmo hub que você poderá interceptar todo o conteúdo que passa pela placa de rede do seu micro. Existem softwares (desenvolvidos para fins legítimos!), como o *Ethereal* [1], que facilitam a análise desse tipo de tráfego, com requintes como destacar os logins e senhas, remontar os emails que tenham sido fragmentados no trajeto e até exibir as imagens que estiverem sendo transferidas pelo seu navegador. E isso sem necessidade de acesso ao seu computador – é uma decorrência da maneira como os hubs funcionam (e uma boa razão para substituí-los por *switches*).

Por exemplo: você vai passar alguns dias na casa de um parente que tem Internet compartilhada. Como ter certeza de que suas senhas ou mensagens confidenciais não serão vistas por alguém mal-intencionado que more no mesmo condomínio? E se você precisar transferir um arquivo estritamente confidencial entre duas filiais de sua empresa via Internet, até que ponto você está disposto a confiar na segurança e discrição de todos os provedores pelos quais essa transmissão irá passar?

A solução ideal – e nem sempre possível – para esse tipo de situação costuma ser uma VPN (rede virtual privada). Mas se você tiver acesso a um servidor externo

em cuja segurança confia, o uso dos túneis baseados no SSH pode ser uma boa solução temporária.

Imagine a seguinte situação: você está em um condomínio usando o computador *C* e quer buscar seus emails no servidor pop.provedor.com.br (identificado pela letra *P*). Se você fizer esse acesso pela maneira direta tradicional, sua senha e suas mensagens poderão ser interceptados por outros computadores na mesma rede local.

Mas se você tiver acesso remoto (via SSH) a um outro computador *S* que considere seguro (por exemplo, um computador na sua empresa, ou mesmo em sua casa, permanentemente ligado à Internet), poderá criar o que chamamos de um *túnel SSH* entre a máquina *C* e a máquina *S* e acessar seu email através desse túnel. Assim, os dados trafegarão criptografados na sua rede local (fluxo número 1 do diagrama) e só serão decodificados entre *S* e *P* (fluxo 2). O servidor *P* não precisa de nenhuma configuração especial e nem saberá que parte do tráfego ocorreu de forma incomum.

Existem muitas formas de se obter esse efeito, mas a mais comum é através do próprio comando `ssh`. Supondo que o endereço do computador *S* seja 200.201.202.203 e que seu login seja *fulano*, você normalmente iria se conectar a ele pelo SSH usando o comando `ssh -l fulano 200.201.202.203`, certo? Mas, para formar o túnel, vamos agregar alguns parâmetros extras. Se você digitar (como root) o comando `ssh -l fulano -L 110:pop.provedor.com.br:110 200.201.202.203`,

Quando você não tem confiança na rede pela qual seus dados trafegam, os túneis SSH podem ser uma solução.

POR AUGUSTO CAMPOS

vai ser aberta uma sessão SSH comum, mas além disso a porta 110 local de *C* passará a ser a entrada do túnel criptografado que passa por *S* (onde é decodificado) e termina na porta 110 (normalmente usada pelos servidores de email POP) do servidor *P*. Essa configuração perdura enquanto a sessão SSH estiver aberta.

Parece complicado, mas o efeito é bastante simples: agora basta abrir o seu programa de email favorito e informar a ele que seu servidor POP deixou de ser o pop.provedor.com.br e passou a ser 127.0.0.1. Na próxima vez que você tentar acessar seu email, abra o túnel e os dados trafegarão criptografados por sua rede local, dificultando qualquer interceptação.

A mesma técnica pode ser usada para outros serviços de rede, basta saber suas portas de acesso [2] e modificar o exemplo acima. Quando se trata de túneis, o limite é a sua criatividade. Eles podem ser usados até mesmo para ultrapassar bloqueios impostos por alguns provedores. ■

INFORMAÇÕES

[1] *Ethereal*: www.ethereal.com

[2] Lista de portas: www.iana.org/assignments/port-numbers

[3] Artigo da O'Reilly: www.oreillynet.com/pub/a/wireless/2001/02/23/wep.html

[4] Quebrando Firewalls com OpenSSH: soutptonuts.sourceforge.net/sshtips.htm

[5] Firewall Piercing mini-HOWTO: www.faqs.org/docs/Linux-mini/Firewall-Piercing.html