O que você precisa saber sobre segurança com IPv6

Modo de segurança

O protocolo IPv6 surgiu nos primeiros dias da web e contém alguns recursos que eram realmente importantes nos anos 90. Mas pelos padrões de hoje, alguns deles podem não ser vistos como melhorias em relação ao IPv4, exigindo de administradores atenção extra com a segurança.

por Ralf Spenneberg

O IPv6 traz diversas mudanças ao protocolo de Internet (IP). As mais conhecidas são o aumento do espaço de endereço de 32 para 128 bits e máscaras de rede mais fáceis. Em vez de máscaras de rede de comprimento variável (VLSM), o IPv6 é compatível apenas com máscaras de 64 bits (`/64`); e os endereços de broadcasts foram abandonados dando lugar a endereços unicast, multicast e anycast.

Além do escopo global que garante a acessibilidade na Internet IPv6, o protocolo IPv6 usa escopos locais diferentes para restringir a acessibilidade de uma área menor. O escopo *link-local* restringe o intervalo de endereços IP à rede local; os escopos *site-local* e *organization-local* combinam múltiplas redes através de roteadores. No IPv6, os sistemas ainda precisam do endereço MAC das placas de rede para se comunicarem. O protocolo ICMPv6 encontra esse dado na camada 3 (do modelo OSI), enquanto o IPv4 ainda usa o *Address Resolution Protocol* (ARP) na camada 2. Os desenvolvedores renomearam esse processo para *Neighbor Discovery Protocol* (NDP) [1].

A técnica de *Path MTU Discovery* agora é obrigatória no IPv6; então, o ICMPv6 tem um papel muito mais importante do que em redes IPv4. Também há compatibilidade com a família de protocolos IPsec – antiga companheira da área de segurança e que administradores usam regularmente para configurar redes virtuais privadas.

Fãs do *Network Address Translation* (NAT) inicialmente ficarão desapontados. Apesar dos desenvolvedores continuarem tentando [2], ainda não há NAT para endereços IPv6 (NAT66). Em outras palavras, cada sistema que precisa se comunicar com outros na Internet IPv6 necessita de um endereço global. Isso significa que os sistemas – sejam eles impressoras de rede ancestrais, clientes Windows sem atualizações ou dispositivos esquecidos – estão sempre acessíveis, a menos que um firewall os bloqueie.

Nada de novo?

Muitos recursos de segurança não mudaram em relação ao IPv4. Os protocolos de transporte na Camada 4 (TCP, UDP) ainda funcionam do mesmo modo; e não há necessidade de configurar os protocolos de aplicativos como HTTP, SMTP e FTP de nenhuma outra maneira, além dos novos endereços. O IPv6 também não aumenta a segurança dos protocolos: um agressor ainda poderia realizar um ataque *SYN flood* (uma variação do DoS) no IPv6 do mesmo modo que no IPv4.

Além disso, há pouquíssimas mudanças na camada 2 do modelo OSI.

O protocolo ARP foi abandonado, o que vai ajudar a evitar alguns tipos de ataques de falsificação, mas não todos (**quadro 1**). Um agressor poderia obter os mesmos resultados com falsificação NDP. Desenvolvedores do IPv6 reconheceram essa forma de ataque e especificaram modificações no protocolo com o RFC 3791 (*Secure Neighbor Discovery Protocol*, SEND) [3]. O SEND usa endereços gerados com criptografia de chaves públicas para autenticar mensagens NDP. Infelizmente, poucos sistemas operacionais são compatíveis com essa extensão, incluindo o Windows 7.

Um administrador só pode combater a falsificação NDP com switches multilayer inteligentes que mantêm uma tabela interna com todos os endereços IPv6 e endereços MAC, para poder identificar e descartar mensagens falsificadas. Fornecedores chamam essa abordagem no IPv4 de *Dynamic ARP Inspection* (DAI) [4].

O NAT não existe mais – sem nenhum substituto à vista – o que não é uma notícia totalmente boa. Inicialmente, administradores que precisam gerenciar FTP, Oracle SQL Net ou H.323 ficarão felizes com a conveniência que essa mudança traz. Esses protocolos negociam novas portas dinamicamente durante as operações. Mapear isso corretamente com tradução de endereços se tornava muitas vezes um grande desafio: o algoritmo NAT precisa entender e rastrear o protocolo para identificar a nova porta negociada.

Outros protocolos que não usam portas, como o ESP do IPsec, podem finalmente ser implementados sem soluções provisórias (como NAT Traversal) através de encapsulamento UDP, do modo que seus desenvolvedores originalmente projetaram.

Privacidade ameaçada

A falta de proteção para os dados é outro aspecto negativo:

- Administradores precisam de um firewall confiável para proteger sua estrutura interna de rede e esconder os IPs utilizados.
- Provedores de Internet podem relacionar conexões individuais a computadores específicos da rede cliente sem a necessidade de cookies
- Computadores são acessíveis de fora da rede local através de seus endereços IPv6.

Usuários domésticos não precisam mais configurar o encaminhamento de portas em roteadores DSL para poderem acessar seus computadores a partir de outros endereços. No entanto, essa nova rota direta para dentro da rede também está aberta para agressores: políticas de firewall robustas são vitais para proteger sistemas globalmente acessíveis.

O IPv6 gera um endereço baseado no endereço MAC (leia a seção sobre configuração automática, mais adiante neste artigo). Como o computador sempre cria a parte host de seu endereço IP (o identificador EUI) do mesmo jeito – baseado no endereço MAC – ele tem um identificador único por vários dias (**figura 1**). Se a máquina é um laptop que se conecta a várias redes, apenas o endereço de rede muda. Se o usuário acessar um servidor na Internet, o dono do servidor pode saber que se trata da mesma máquina devido ao identificador idêntico no endereço IPv6.

Um operador de website também pode ver a partir de qual rede o usuário do laptop visitou o site. Tudo o que ele precisa fazer é consultar o endereço de rede usando `whois` (**listagem 1**). Os endereços IPv6 são registrados pelo provedor como os anteriores; o comando `whois` pode identificar a empresa para quem o endereço está designado. No pior cenário, seria possível, por exemplo, criar um perfil de movimentação para um vendedor que está viajando.

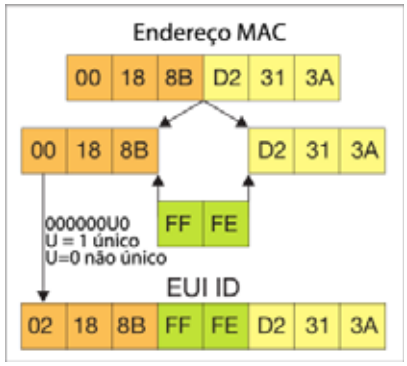

Figura 1 O ID de um endereço IPv6 é construído com o endereço MAC: acrescenta-se `FFFE` e o segundo bit do primeiro byte é invertido.

Identificador EUI aleatório

Para evitar o rastreamento, o RFC 4941 introduziu as extensões de privacidade do IPv6 [5], que faz o sistema operacional criar o identificador EUI de modo aleatório (em vez de usar o endereço MAC). Todos os sistemas operacionais modernos são compatíveis com essa função, e o Windows Vista e Windows 7 habilitam isso por padrão. No Linux, o administrador pode habilitar a função temporariamente com o comando:

```
sysctl -w net.ipv6.conf.all.
➥ use_tempaddr=2
```

Para tornar essa mudança permanente, é preciso modificar as definições do arquivo `/etc/sysctl.conf`, conforme a distribuição ou sistema operacional [6].

Mas esse tipo de designação tem um lado ruim: sempre que um sis-

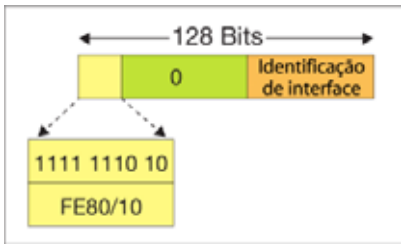

Figura 2 O computador pode gerar um endereço link-local sem ajuda externa.

tema operacional habilitar a placa de rede (ou pelo menos uma vez por dia), ele cria um novo identificador. Se um vírus ou worm atingir a rede e não for descoberto pelo administrador até o dia seguinte, será impossível relacionar os endereços IP logados pelo firewall aos sistemas individuais, porque ninguém tem registro dos endereços IPv6 aleatórios.

Firewall em vez de NAT

Com o IPv6, administradores não têm opção a não ser introduzir regras robustas no firewall. Felizmente, com regras orientadas a estados você alcança o mesmo nível de segurança de um cenário NAT. Clientes só podem estabelecer conexões de rede em uma única direção.

Na verdade, o uso puro de NAT é menos seguro, pois apenas obscurece os endereços IPs, em vez de evitar o acesso. Com roteamento de fonte e conhecimento dos endereços IP, um agressor poderia até abrir conexões em sistemas NAT, se o firewall assim permitisse. Dito isso, o kernel Linux não introduziu filtros *stateful* para IPv6 até a versão 2.6.20. Assim, distribuições antigas (por exemplo, RHEL 5 e CentOS 5) são inúteis como firewalls IPv6.

Para garantir que apenas clientes internos possam se conectar a sistemas na Internet e, ao mesmo tempo, bloquear conexões até as máquinas locais, você pode usar regras de firewall como os do script simples da **listagem 2**.

As quatro regras dão permissão para todos os pacotes que pertençam às conexões `ESTABLISHED` ou que contenham mensagens de erro para elas (`RELATED`). Conexões `NEW` só são permitidas se vierem da rede interna. Se quiser usar protocolos como FTP, que negociam dinamicamente novas portas, é preciso carregar os módulos correspondentes (por exemplo, com `modprobe nf_conntrack_ftp`).

Uma tarefa muito mais difícil é proteger o próprio firewall. Em um cenário somente IPv4, você pode simplesmente dispensar todos os pacotes endereçados a ele nas cadeias `INPUT` e `OUTPUT`, a menos que precise acessá-lo através da rede. Se você fizer isso com o IPv6, não será possível nenhuma comunicação através do firewall, já que o IPv6 usa NDP em vez de ARP. Todo sistema que quiser usá-lo para se comunicar com endereços além do setor local precisa descobrir o endereço MAC do firewall; o que não é um problema no IPv4, já que o iptables ignora os pacotes ARP na Camada 2 (a resolução de endereços MAC funciona independentemente das regras de iptables).

Com o IPv6, a resolução de endereços MAC depende do ICMPv6 na camada 3. Se as regras do ip6tables dispensarem todos os pacotes, isso inclui as mensagens ICMPv6, e a resolução de endereços MAC não funcionará. Em outras palavras, você precisa permitir pelo menos as mensagens ICMPv6. Como o IPv6 usa o protocolo ICMPv6 para diversas funções, o RFC 4890 [7] fornece instruções detalhadas para ajudar administradores de firewall a definir as regras certas.

Listagem 1: whois 2001:67c:24::/48

```
01 Querying whois.ripe.net
02 [whois.ripe.net]
03 inet6num:       2001:67c:24::/48
04 netname:        SPE6-NET
05 descr:          OpenSource Training Ralf Spenneberg
06 country:        DE
07 org:            ORG-OTRS1-RIPE
08 admin-c:        RS9110-RIPE
09 tech-c:         RS9110-RIPE
10 status:         ASSIGNED PI
11 mnt-by:         RIPE-NCC-END-MNT
12 mnt-by:         MNT-Jansen
13 mnt-lower:      RIPE-NCC-END-MNT
14 mnt-router:     MNT-Jansen
15 mnt-domains:    MNT-JANSEN
16 source:         RIPE # Filtered
```

Listagem 2: ip6tables

```
01 LAN=eth0
02 INTERNET=ppp0
03 IPT=/sbin/ip6tables
04
05 $IPT -P FORWARD DROP
06 $IPT -F FORWARD
07 $IPT -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
08 $IPT -A FORWARD -i $LAN -o $INTERNET -m state --state NEW -j ACCEPT
```

Autoconfiguração

Um dos melhores recursos do protocolo IPv6 é a *Stateless Automatic Autoconfiguration* (SLAAC) [8] para os endereços IPv6 de sistemas individuais. Como você não precisa configurar um servidor DHCP, isso efetivamente acaba com um ponto de falhas.

Como já mencionei, os sistemas geram um identificador baseado em seus endereços MAC. Primeiro, eles usam o identificador para gerar

um endereço link-local ao anexar o identificador à rede `fe80/64` (**figura 2**).

Hosts podem usar esse endereço para se comunicarem na rede local, como era possível no IPv4 com APIPA (Windows [9]) ou no Avahi (Linux [10]) como "endereço de IP de conexão local" em uma rede Zeroconf [11].

Os hosts, então, usam mensagens de solicitação de roteamento ICMPv6 [1], as quais os roteadores respondem com anúncios de roteamento ICMPv6 contendo um prefixo global. Os sistemas criam um endereço IP global com esse dado, anexando seu identificador ao prefixo global (**figura 3**). Se diversos roteadores responderem, ou se o anúncio conter múltiplos prefixos, o sistema criará múltiplos endereços IPv6.

Para a autoconfiguração funcionar, as regras de firewall obviamente precisam permitir essas mensagens de solicitação ICMPv6; contudo, você não pode filtrá-las de modo stateful porque o servidor de configuração as anunciam em multicasts. Infelizmente, solicitações de roteamento e mensagens de anúncio também não podem ser autenticadas. Em outras palavras, seria possível um agressor enviar mensagens de anúncio de roteamento e efetivamente falsificar um roteador. As ferramentas para esse tipo de ataque já existem [12].

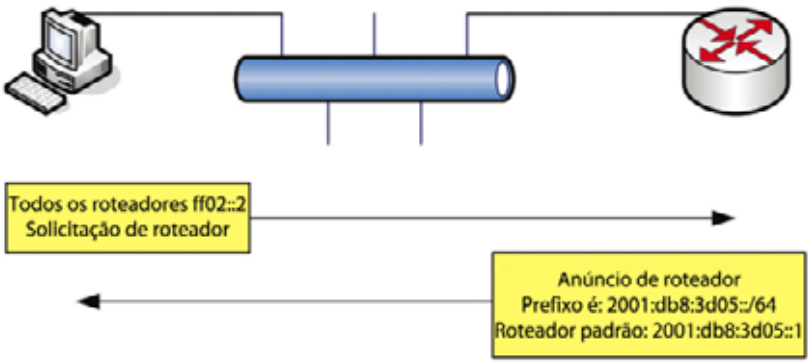

Figura 3 O cliente usa o endereço link-local para se comunicar com todos os roteadores da rede através do endereço multicast `ff02::2`.

Em condições de produção, a autoconfiguração stateless na verdade não é adequada para redes modernas. Os desenvolvedores não levaram em consideração a distribuição de servidores DNS e outros dados quando criaram as especificações há 15 anos. O RFC 5006 [13] adiciona suporte a servidor de DNS mas não é compatível com todos os sistemas operacionais. De qualquer modo, administradores estão acostumados a usar DHCP para distribuir dados do tipo domínio DNS, servidor NTP ou servidor de boot PXE.

Com o IPv6, administradores não podem simplesmente evitar o DHCP, ou pelo menos sua versão stateless. O sistema operacional usa a autoconfiguração para descobrir seu próprio endereço IPv6 e, então, o DHCP para pegar mais dados, como os do servidor DNS e domínio DNS. O Windows Vista e o 7 fazem isso automaticamente, se a flag `OtherConfig` estiver definida no anúncio de roteamento.

Entradas DNS e DHCP

Finalmente, em algumas configurações você quer o que servidor DNS acrescente automaticamente os hostnames de todos os sistemas que inicializem em seu diretório; isso é típico em um cenário Active Directory, por exemplo.

Em sistemas Linux isso dá um bom trabalho, o que tem levado muitos administradores a usar uma solução provisória no IPv4 e a autorizar o servidor DHCP a modificar os dados na zona DNS dinamicamente.

Com o IPv6, você não pode usar a autoconfiguração e o DHCP stateless porque o servidor DHCP precisa saber o endereço IPv6 do cliente para adicionar um registro. Em outras palavras, você precisa de um servidor DHCP stateful para designar endereços IPv6 a partir de uma fila.

Quadro 1: Falsificação ARP

Redes com switches tornam o sniffing bem difícil. Um switch detecta qual sistema está em cada porta e encaminha pacotes Ethernet somente para essa porta. Para isso, ele usa uma tabela de endereços MAC identificados e as portas correspondentes. Apesar disso, agressores ainda podem fazer um sniff em todo o tráfego com um ataque do tipo interceptador (*man in the middle*).

Antes de um cliente com IPv4 poder se comunicar com um servidor, ele precisa descobrir o endereço MAC do parceiro de comunicação. Para tanto, ele usa o protocolo ARP, enviando uma requisição ARP ao endereço ethernet de broadcast, perguntando por exemplo "Qual é o endereço MAC do IP 192.168.0.5?". Um agressor pode interferir aqui, respondendo ao cliente com uma resposta ARP falsificada que contém seu próprio endereço MAC. O cliente então envia o pacote não para o alvo IPv4 desejado, mas para o endereço MAC falsificado. Um switch Camada 2 apenas usa esse endereço MAC e, assim, encaminha o pacote ao agressor.

O invasor pode ler o pacote e então só precisa substituir o MAC de destino com o endereço correto e enviar de volta ao switch. O pacote então é enviado ao destinatário correto e a conexão está pronta.

Clientes Windows aceitam isso automaticamente se o anúncio do roteador tiver a definição `ManagedFlag`. Em sistemas Linux, você precisa instalar e configurar de antemão um cliente DHCP adequado. A redundância é importante aqui; se o servidor DHCP falhar, haverá problemas na rede local.

A morte do Smurf

O IPv6 acaba com os broadcasts, o que é uma coisa boa. Simples ataques de amplificação, como a técnica Smurf [14], são assim impossíveis e tempestades de broadcasts são coisas do passado também. Embora o NDP use endereços multicast para descobrir o endereço MAC de seus parceiros de comunicação, os grupos multicast são escolhidos de modo inteligente para que a comunicação seja restrita ao único computador que o host precisa encontrar, na maioria dos casos.

Outros serviços que dependiam da comunicação broadcast agora usam multicast; por exemplo, o DHCPv6 usa os dois endereços bem conhecidos `ff02::1:2` e `ff05::1:3`. O primeiro é um endereço link-local multicast (`ff02`) reservado para todos os agentes DHCP (servidores e relays). O segundo é um endereço site-local multicast (`ff05`) reservado para servidores DHCP.

Graças a essas designações, o IPv6 tipicamente consegue restringir a manipulação de pacotes na pilha IP dos sistemas envolvidos na troca. Isso reduz dramaticamente o tráfego de rede, comparado com comunicações broadcast IPv4.

O IPv6 também tem compatibilidade obrigatória com a família do protocolo IPsec. Mas isso não melhora automaticamente a segurança: se o administrador não configurar esse recurso, o tráfego não será criptografado. A presença do IPsec não necessariamente significa mais segurança; apenas economiza o trabalho de ter que instalar programas. Questões de interoperabilidade e erros de configuração ainda existem.

Dia da caça

Considerando o estado da arte e redes típicas de 15 anos atrás, os inventores do IPv6 acertaram em muitas coisas. Desde então, os sistemas operacionais e o gerenciamento de rede com o IPv4 se desenvolveram substancialmente; enquanto com o IPv6, isso não aconteceu. A autoconfiguração só configura o endereço IP e, se os clientes forem modernos, designa o servidor DNS. Para todos os outros dados, você ainda precisa de um servidor DHCP.

DNS e firewalls estão se tornando mais importantes e, sem o IPsec, o IPv6 não irá melhorar a segurança de uma rede corporativa. Pelo contrário, significa que administradores terão que prestar ainda mais atenção se os clientes estão protegidos ou acessíveis globalmente. ■

Mais informações

[1] NDP: http://en.wikipedia.org/wiki/Neighbor_Discovery_Protocol

[2] Rascunho NAT66: http://www.ietf.org/proceedings/08nov/slides/behave-14.pdf

[3] Protocolo Secure Neighbor Discovery (SEND): http://www.faqs.org/rfcs/rfc3971.html

[4] Dynamic ARP Inspection (DAI): http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SXF/native/configuration/guide/dynarp.html

[5] Extensões de privacidade IPv6: http://www.faqs.org/rfcs/rfc4941.html

[6] Habilitando extensões IPrivacy: http://blog.philippklaus.de/2011/05/ipv6-privacy-extensions/

[7] Filtrando mensagens ICMPv6 em firewalls: http://www.faqs.org/rfcs/rfc4890.html

[8] Stateless Automatic Autoconfiguration (SLAAC): http://www.faqs.org/rfcs/rfc4862.html

[9] APIPA: http://msdn.microsoft.com/en-us/library/aa505918.aspx

[10] Avahi: http://avahi.org/

[11] Zeroconf: http://en.wikipedia.org/wiki/Zero_configuration_networking

[12] Suíte de ataque THC IPv6: http://www.thc.org/thc-ipv6/

[13] Opção IPv6 para configuração DNS: http://www.faqs.org/rfcs/rfc5006.html

[14] Técnica de ataque Smurf: http://en.wikipedia.org/wiki/Smurf_attack

O autor

Ralf Spenneberg é instrutor freelancer de Unix/Linux, consultor, autor e CEO de sua própria empresa de treinamento. Ralf publicou diversos livros sobre detecção de invasão, SELinux, firewall e VPNs. A segunda edição de seu último livro, *VPN on Linux*, foi publicada recentemente.

Gostou do artigo?

Queremos ouvir sua opinião. Fale conosco em cartas@linuxmagazine.com.br

Este artigo no nosso site: http://lnm.com.br/article/5871