Postado por: http://sonysantos.blogspot.com/

Iptables NAT para crianças

--- Introdução ---

Na empresa em que trabalho sempre tivemos dificuldades de entender como funcionam os redirecionamentos entre os computadores da rede interna e a internet. Contratávamos os serviços de outra empresa quando algum programa precisava acessar a internet sem passar pelo firewall, ou quando alguém de fora precisasse acessar um computador específico de nossa rede.

Os tutoriais que encontramos na internet são todos muito grandes e complexos, falando de um monte de opções do iptables que não interessam para a gente, tornando a leitura cansativa e o aprendizado improdutivo. Depois de três semanas quebrando a cabeça, consegui entender o funcionamento dos redirecionamentos, e vim compartilhar com vocês, tentando suprir a necessidade de um texto mais ao alcance de quem nunca mexeu com isso, evitando todo conceito desnecessário.

Vou expor a minha experiência, na minha rede. Talvez na sua rede não funcione dessa forma, mas o texto pode inspirar as adaptações que forem necessárias. No entanto, use o texto por sua própria conta e risco, pois não posso me responsabilizar por quaisquer prejuízos que acidentalmente venham a ocorrer com o uso deste artigo.

--- Nossa configuração ---

Então, vamos ao que interessa. Temos uma rede interna 192.168.0.0/24, isto é, com os 24 primeiros bits constantes, com IPs variando de 192.168.0.0 a 192.168.0.255. Os IPs são fixos, isto é, cada computador tem seu nº de IP, que não se perde ao ser desligado. Temos um **gateway**, que é o computador que faz a ligação da rede interna com a internet; seu IP é 192.168.0.1. Qualquer requisição à internet vindo de um computador da rede interna tem que passar pelo gateway. Seu sistema operacional é **linux**, e nele está instalado também um **firewall**, que bloqueia acessos a páginas não autorizadas (orkut, youtube, etc.). O firewall que usamos é o **squid**.

O squid também é um **proxy**, isto é, ele armazena as páginas mais usadas em cache, diminuindo as requisições à internet e, conseqüentemente, tornando a navegação mais rápida e consumindo menos banda.

Para entender os redirecionamentos, precisamos entender como é a comunicação entre dois computadores e quais são os caminhos dos pacotes de dados.

--- A comunicação entre dois computadores ---

Toda comunicação tem um caminho de ida e de volta. O computador A envia uma requisição ao computador B e fica esperando uma resposta de B. O computador B processa a requisição e devolve a resposta ao computador A. O computador A recebe a resposta e a requisição fica satisfeita.

O computador B pode ter vários aplicativos diferentes rodando que podem responder ao computador A. Para que os aplicativos não briguem pela posse do pacote, cada um fica esperando sinal ("escutando") em uma **porta** diferente. Assim, o computador A envia seu pedido para a porta específica do aplicativo que ele quer acessar em B.

Por exemplo, B pode ter um servidor de internet, um servidor de e-mails e um servidor de telnet rodando. Cada um está escutando em uma porta. Se A quer acessar uma página de internet de B, ele manda uma requisição para o computador B na porta 80. Se quer baixar um e-mail de B, faz uma requisição à porta 110, e se quer fazer um telnet na máquina B, usa a porta 23.

Uma lista de portas usadas por aplicativos comuns pode ser encontrada em http://en.wikipedia.org/wiki/Well_known_ports.

Há vários tipos (protocolos) de pacotes de dados, mas o protocolo mais importante para nós é o TCP. Os pacotes TCP têm um cabeçalho que inclui o endereço IP e o número da porta de origem e de destino.

Vamos a um exemplo.
computador A: IP 192.168.0.6

computador B: IP 192.168.0.42

Computador A quer acessar uma página de internet no computador B, digamos, http://192.168.0.42/forum/. No computador B o servidor de internet está escutando na porta 80. Então ele manda um pacote requerendo essa página, com os seguintes dados:

Origem: 192.168.0.6:4432
Destino: 192.168.0.42:80
Requisição: http://192.168.0.42/forum/

O número depois de "IP:" é o número da porta. O computador A usa uma porta qualquer que esteja livre para fazer a requisição (no caso, 4432), mas a porta destino no computador B é bem definida; não pode ser outra (pois é na porta 80 que o servidor de internet de B está esperando requisições).

O computador A fica então ouvindo na porta 4432 (a porta escolhida no caso) à espera de uma resposta da porta 80 de B. Ele espera uma resposta exatamente assim:

Origem: 192.168.0.42:80
Destino: 192.168.0.6:4432
Resposta: *conteúdo da página*

Se tudo correr bem, B recebe a requisição de A, processa com o servidor de internet (porta 80), pega os dados da página requisitada e manda um pacote de volta para A, com o seguinte cabeçalho:

Origem: 192.168.0.42:80
Destino: 192.168.0.6:4432

que é justamente o que A estava esperando, satisfazendo a requisição. Se houver algum extravio no caminho e A receber um pacote de outro IP, de outra porta, ou para outra porta, em que ou de quem não estava esperando nada, o pacote é ignorado e perdido.

(Na verdade a comunicação envolve passos mais complexos e vários pacotes, mas esta simplificação será suficiente para nossos propósitos.)

--- O papel do gateway ---

Quando dois computadores da nossa rede (isto é, com IP 192.168.0.xxx) se comunicam, eles fazem isso diretamente. Um tem o IP do outro e manda o pacote diretamente para ele, sem passar pelo gateway.

No entanto, quando um computador da nossa rede precisa se comunicar com um computador de fora, para acessar a internet, a requisição tem que passar pelo gateway, que fará o roteamento do pacote, isto é, a comunicação entre as duas redes.

O gateway deve ter pelo menos duas placas de rede (**interfaces** de rede): uma para nossa rede interna (a do IP 192.168.0.1) e outra para acessar a internet, ligada ao modem ADSL, com IP que, no caso, é fixo, e vou chamar de IP_EXTERNO, que é um IP válido na internet.

Suponhamos que queiramos acessar, do computador A, uma página de internet que esteja hospedada em um computador C fora de nossa rede, digamos, http://www.exemplo.com.br. A primeira coisa que o navegador de internet fará é tentar descobrir o endereço IP de www.exemplo.com.br. Vamos assumir que seja 200.201.202.203. Depois de obtido o IP de C, o pacote sairá de A da seguinte forma:

Origem: 192.168.0.6:1234
Destino: 200.201.202.203:80
Requisição: http://www.exemplo.com.br

O pacote chega no gateway pela interface interna e é encaminhado à externa, para sair em direção ao IP destino. Contudo, se o pacote sair com esse cabeçalho ele não terá como voltar. IPs que começam com 192.168 são IPs de rede interna, e não existem na internet.

Todos os IPs que começam com

10.
127.
192.168.
172.16. a 172.31.
são IPs de rede interna e não existem na internet, sendo portanto descartados pelos roteadores por onde passam os pacotes de internet.

Para que a comunicação funcione, o gateway precisa "traduzir" o endereço de origem para o seu endereço na internet, isto é, o IP_EXTERNO. Esse processo se chama "tradução de endereço de rede", mais conhecido pela sua sigla em inglês, **NAT** (network address translation). Assim, o pacote sai do gateway com o seguinte cabeçalho:

Origem: IP_EXTERNO:32765
Destino: 200.201.202.203:80

Observe que o número da porta de origem também é diferente, mas deve ser uma porta livre na interface externa do gateway.

O gateway mantém uma tabela interna onde relaciona as traduções de endereço. Neste caso específico, haverá a seguinte relação:

192.168.0.6:1234 <-> IP_EXTERNO:32765

O pacote chega então ao computador C (200.201.202.203), que processa o pedido e manda um pacote de volta, endereçado para IP_EXTERNO:32765:

Origem: 200.201.202.203:80
Destino: IP_EXTERNO:32765
Resposta: *conteúdo da página*

O gateway, que estava esperando a resposta nessa porta, recorre à sua tabela de NAT e traduz o cabeçalho de volta para 192.168.0.6:1234, enviando o pacote de retorno pela interface interna.

Origem: 200.201.202.203:80

Destino: 192.168.0.6:1234
Resposta: *conteúdo da página*

Finalmente o computador A recebe a mensagem que estava esperando de 200.201.202.203:80 na porta 1234, completando a comunicação.

Situação semelhante acontece quando é necessário que um computador de dentro da nossa rede precise ser acessado de fora. Como endereçar, na internet, uma requisição a um IP de rede interna? Realmente, não há como. O máximo que dá para fazer é endereçar ao gateway usando seu IP_EXTERNO. A solução também se dá com NAT, associando um número de porta do gateway a um computador/porta específico da rede interna.

Por exemplo, suponha que o computador D, fora de nossa rede, precise acessar um servidor de internet (poderia ser VNC ou outra coisa) que está no computador 192.168.0.2 em nossa rede, na porta 80. Ele não pode mandar um pacote para 192.168.0.2, mas pode mandar para IP_EXTERNO. No gateway, configuramos o NAT para desviar tudo o que for para a porta 80 redirecionando para 192.168.0.2:80 (poderiam ser portas diferentes).

Então, o pacote será assim:

Origem: 123.45.67.89:6767 (computador D)
Destino: IP_EXTERNO:80

O gateway recebe o pacote e desvia-o para 192.168.0.2, seguindo a regra configurada:

Origem: 123.45.67.89:6767
Destino: 192.168.0.2:80

O 192.168.0.2 processa o pedido e devolve a resposta para 123.45.67.89:6767:

Origem: 192.168.0.2:80
Destino: 123.45.67.89:6767

Como tudo o que for pra rede externa tem que passar pelo gateway, ao passar por lá ele traduz o endereço do pacote de volta para IP_EXTERNO:

Origem: IP_EXTERNO:80
Destino: 123.45.67.89:6767

E a comunicação se completa.

Mais tarde veremos como configurar o gateway para fazer o NAT.

--- O que acontece quando há um proxy ---

A intenção de se ter um proxy é fazer com que as requisições à internet passem primeiro pelo proxy para depois, se necessário (isto é, se a página requisitada ainda não estiver no cache do proxy e não for uma página bloqueada), irem à internet.

O proxy pode ser explícito ou transparente. No primeiro caso, os navegadores de todos os computadores da rede interna são configurados para acessar a internet via proxy (com o IP e a porta do proxy); no segundo, o gateway desvia todas as requisições endereçadas à internet para a porta do proxy, sem ser necessário configurar cada máquina. Na nossa empresa utilizamos o proxy de forma explícita.

Nosso proxy squid, que está instalado no gateway (192.168.0.1), escuta na porta 3128.

Se nosso computador A precisar requisitar aquela mesma página http://www.exemplo.com.br através do proxy, o pacote será assim:

Origem: 192.168.0.6:5678
Destino: 192.168.0.1:3128
Requisição: http://www.exemplo.com.br

(Lembrando que essa é uma simplificação do processo para tornar o texto mais didático.)

Ou seja: o destino do pacote é o proxy. Significa que o computador A estará aguardando uma resposta de 192.168.0.1:3128, e não de 200.201.202.203:80.

O squid receberá o pacote e processará a requisição. Se a página requisitada for uma página bloqueada, ele monta um pacote com a resposta de página bloqueada e o retorna para 192.168.0.6:5678. (Lembre-se de que o squid também é um firewall e pode bloquear algumas páginas, de acordo com sua configuração.) Se for uma página permitida e se essa página estiver em seu cache, o squid monta um pacote com a página do cache e o envia como resposta ao micro A. Finalmente, se a página requisitada for permitida e não estiver no cache, o próprio squid fará a requisição dessa página na internet. Então, do gateway, sairá um pacote assim:

Origem: IP_EXTERNO:33007
Destino: 200.201.202.203:80
Requisição: http://www.exemplo.com.br

(O IP_EXTERNO é definido no squid pela opção tcp_outgoing_address no arquivo de configuração do squid.)

Quando o servidor 200.201.202.203 recebe o pacote, processa-o e envia a página para IP_EXTERNO:33007. O squid recebe a página, guarda-a no cache se julgar necessário, e a encaminha para 192.168.0.6:5678, com origem 192.168.0.1:3128, conforme o cliente A estava esperando.

Observe que, quando há proxy, não é necessário fazer o NAT, pois ao requisitar uma página na internet o proxy já usa o IP_EXTERNO, válido na internet.

--- Por que isso não funciona sempre ---

A grande maioria das páginas e aplicações (como o MSN, por exemplo) funciona muito bem com o proxy squid. Há, porém, alguns casos de aplicativos que não sabem trabalhar com proxy, e que necessitam de conexão direta com seu servidor na internet. Esse é o caso, por exemplo, da Conectividade Social da Caixa e alguns programas de home banking.

Nesse caso, será preciso fazer NAT, pois, como a requisição não passará pelo proxy, será necessário traduzir o endereço de origem para que o pacote possa retornar.

Também é importante lembrar que quando algum computador de fora precisa fazer uma requisição a algum computador interno, o pedido não passa pelo proxy, e é preciso fazer NAT.

Há vários modos de configurar NAT, e vamos usar o **iptables**, uma ótima ferramenta que temos em nosso gateway. Mais informações sobre o iptables e como obtê-lo podem ser encontradas no site de seu fabricante, http://www.netfilter.org.

Como criar uma regra no iptables:

iptables *tabela cadeia protocolo origem destino ação*

tabela: O iptables trabalha com 3 tipos de tabela: **filter**, **mangle** e **nat**. Para nosso caso só interessa a última. É usada com a chave **-t**; portanto usaremos sempre **-t nat**.

cadeia: A tabela nat tem 3 tipos de cadeia: **PREROUTING**, **OUTPUT** e **POSTROUTING**. As chaves para especificar a cadeia são: **-A** (append, isto é, adicionar a regra no fim da lista), **-I** (insert, isto é, inserir a regra no início da lista ou em um lugar específico) e **-D** (delete, para apagar a regra).

protocolo: Usaremos geralmente **-p tcp**. O protocolo é um ítem opcional, mas é útil quando precisamos identificar um pacote pela sua porta de origem ou de destino, pois ele aceita um parâmetro complementar **--sport** ou **--dport** (source/destination port) que permite especificar a porta.

origem: **-s** seguido do número do IP ou da rede de origem. (O número da rede é o número do primeiro IP da rede seguido da máscara; no nosso caso, 192.168.0.0/24.)

destino: **-d** seguido do número do IP ou da rede de destino.

ação: **-j** seguido da ação. A ação depende da cadeia. Para as cadeias PREROUTING e OUTPUT, a ação deve ser **DNAT** (destination nat), que muda o endereço de destino do pacote. Para a cadeia POSTROUTING, a ação deve ser **SNAT** (source nat), que muda o endereço de origem do pacote, ou **MASQUERADE**, que é um tipo de SNAT.

O iptables tem muito mais opções, mas aqui estão as que nos interessam. *Protocolo*, *origem* e *destino* são ítens opcionais, mas serão nossa forma de identificar quais pacotes devem ser afetados pela regra. É importante que todas as opções sejam mantidas conforme apresentei aqui, em maiúsculas ou minúsculas. Por exemplo, "-t nat", "-j" e "--sport" têm que estar em minúsculas, e "-A", "PREROUTING", "MASQUERADE" e "SNAT" têm que estar em maiúsculas.

Antes de podermos usar essas opções, temos que conhecer o caminho dos pacotes num gateway com iptables.

--- O fluxo dos pacotes pelas cadeias do iptables ---

Há basicamente três caminhos diferentes pelos quais um pacote pode caminhar dentro do gateway:

1. Entrada -> PREROUTING -> roteamento -> POSTROUTING -> saída
2. Entrada -> PREROUTING -> roteamento -> gateway
3. Gateway -> OUTPUT -> roteamento -> POSTROUTING -> saída

(Há outras cadeias, mas estou considerando apenas as do NAT.)

Esse é o fluxo dos pacotes de requisição. Os pacotes de resposta voltam pelo mesmo caminho, no sentido inverso.

Quando um pacote de requisição entra no gateway, seja pela interface interna ou pela externa, ele passa primeiro pela cadeia PREROUTING, que significa "antes do roteamento". No roteamento, o gateway decide para onde vai o pacote (para qual interface de rede, etc.), analisando o endereço de destino.

Se o destino for outro micro, o pacote segue para a cadeia POSTROUTING ("depois do roteamento") e, em seguida, para a saída (caminho 1). Quando o outro computador responder, o pacote retornará por esse mesmo caminho, no sentido inverso.

Se o destino for o próprio gateway, como no caso das requisições ao proxy, o pacote segue para o processamento interno (para o squid, por exemplo, ou para outra aplicação do gateway, dependendo do número da porta). Esse é o caminho 2. O gateway processa a requisição e gera um pacote de resposta, que retorna para o cliente pelo mesmo caminho 2.

Quando o gateway faz uma requisição (por exemplo, quando o squid requisita uma página da internet), ela segue pelo caminho 3. Claro que a resposta do outro micro voltará também por esse caminho.

Muito bem: agora vem a parte legal: o que dá pra fazer nas cadeias do iptables.

Perceba que PREROUTING e OUTPUT estão antes do roteamento. Essas cadeias são úteis para quando for preciso mudar o destino do pacote, justamente antes do roteamento. Por isso a elas está associada a ação DNAT, que é a tradução do endereço de destino.

Por outro lado, o pacote que passa pela cadeia POSTROUTING já está pronto para sair. Essa é a cadeia usada quando precisamos mudar o endereço de origem do pacote, com a ação SNAT.

Agora chegou a hora dos exemplos detalhados, com os quais iremos entender certinho o que acontece.

--- Exemplo 1: Home Banking ---

Alguns aplicativos de home banking não trabalham com proxy, e precisam se comunicar direto com o servidor.

O pacote que sai do computador interno em direção ao servidor do banco tem que ter o endereço de origem traduzido para um IP válido na internet, isto é, um SNAT;

portanto, usaremos a cadeia POSTROUTING.

Para identificarmos quais pacotes devem ser afetados pelo SNAT, temos que recolher o maior número de informações possíveis:
- o IP do servidor do banco (chamemos de IP_BANCO);
- a porta em que o servidor do banco atende ao Home Banking (PORTA_HB);
- o computador da rede interna que irá usar o aplicativo (IP_CLIENTE).

Vamos, agora, montar o iptables. Recordemos sua estrutura:

iptables *tabela cadeia protocolo origem destino ação*

tabela: Sempre **-t nat**.

cadeia: Vamos adicionar a regra na cadeia POSTROUTING; portanto usaremos **-A POSTROUTING**.

protocolo: Como temos a porta do home banking nos servidor, usaremos **-p tcp --dport PORTA_HB**.

origem: **-s IP_CLIENTE**

destino: **-d IP_BANCO**

ação: **-j SNAT --to IP_EXTERNO**. Isso é o que vai fazer o pacote ter o IP_EXTERNO como endereço de origem.

Eis nosso comando completo:

iptables -t nat -A POSTROUTING -p tcp --dport PORTA_HB -s IP_CLIENTE -d IP_BANCO -j SNAT --to IP_EXTERNO

(Lembre-se de que PORTA_HB, IP_CLIENTE, IP_BANCO e IP_EXTERNO devem ser substituídos pelos valores reais.)

Vamos acompanhar os pacotes afetados por essa regra.

Primeiro, o computador cliente fará uma requisição ao servidor do banco. O pacote sairá assim:

Origem: IP_CLIENTE:3307
Destino: IP_BANCO:PORTA_HB
Requisição: ...

Ao chegar no gateway, o pacote passa pelo PREROUTING sem ser alterado, pois nenhuma regra lá o afeta; segue para o roteamento, que o encaminha a cadeia POSTROUTING (caminho 1). Chegando lá, é afetado pela regra que acabamos de criar, pois todos os critérios da regra se encaixam com as características do pacote. O pacote é, então, alterado para:

Origem: IP_EXTERNO:12057
Destino: IP_BANCO:PORTA_HB
Requisição: ...

E no gateway haverá uma tabela de conversão com a seguinte informação:

IP_CLIENTE:3307 <-> IP_EXTERNO:12057

A requisição sai do gateway para a internet, até o servidor do banco. Chegando lá, é processada e é o servidor gera um pacote de resposta:

Origem: IP_BANCO:PORTA_HB
Destino: IP_EXTERNO:12057
Resposta: ...

O gateway recebe a resposta, que chega na cadeia POSTROUTING. O gateway consulta sua tabela de SNAT e converte o pacote para:

Origem: IP_BANCO:PORTA_HB
Destino: IP_CLIENTE:3307
Resposta: ...

O pacote segue dentro do gateway para a cadeia PREROUTING. Como essa cadeia não alterou o pacote em sua ida, também não o altera em seu retorno. O pacote segue, então para o IP_CLIENTE, completando a comunicação. Tudo isso sem passar pelo proxy squid.

--- O uso do MASQUERADE ---

O MASQUERADE é um SNAT que coloca automaticamente no pacote o IP da interface de saída, sem que você precise especificar o IP_EXTERNO. Ele é útil quando o IP_EXTERNO é um IP variável ou quando não se sabe qual é a interface de saída do pacote (cada placa de rede tem um IP). Assim, a ação do exemplo acima pode ser **-j MASQUERADE**, isto é:

iptables -t nat -A POSTROUTING -p tcp --dport PORTA_HB -s IP_CLIENTE -d IP_BANCO -j MASQUERADE

Isso funcionará da mesma maneira em nosso exemplo.

Pelo fato de o MASQUERADE checar o IP da interface de saída para todo pacote que se encaixa na regra, ele é mais lento que o SNAT. Portanto, se você sabe exatamente qual IP de origem o pacote deve ter, prefira usar SNAT.

--- Exemplo 2: a Conectividade Social ---

A Conectividade Social (CS) da Caixa é um caso semelhante, com as seguintes ressalvas:
- A Caixa não tem apenas um IP de destino, mas um conjunto; no caso, 200.201.160.0/20;
- A CS usa a porta 80, isto é, a mesma para as requisições de páginas de internet;
- A CS usa as configurações de proxy do Internet Explorer.

Para que a CS não use o proxy, é preciso incluir os servidores da Caixa entre as exceções do proxy, nas opções do Internet Explorer.

Adicionalmente, a porta 80 deve estar livre para a CS, isto é, não pode haver outros programas utilizando a porta ao mesmo tempo, no computador cliente.

A solução completa para a Conectividade Social em nossa rede você encontra em http://sonysantos.blogspot.com/2008/02/conectando-se-conectividade-social.html. Com a bagagem atual fica bem mais fácil entender o que está acontecendo lá.

--- Exemplo 3: Um servidor de internet em nossa rede ---

Voltemos ao exemplo de liberar acesso externo a um servidor de internet localizado em 192.168.0.2. Como já vimos, um computador na internet não poderá endereçar sua requisição diretamente para um IP interno; então ele vai ter que endereçar a IP_EXTERNO, e iremos configurar o gateway para que desvie toda requisição à porta 80 para 192.168.0.2:80 (as portas não precisam ser as mesmas no gateway e no micro da rede interna, pois o NAT traduz tanto o IP quanto a porta, se necessário).

Isso significa que vamos fazer um DNAT (destination nat), pois estamos alterando o endereço de destino.

Um DNAT deve ser feito *antes* do roteamento, pois é no roteamento que o gateway decide para onde vai o pacote. Não adianta mudar o endereço de um pacote depois que ele for direcionado. Portanto, estaremos trabalhando com a cadeia PREROUTING.

Basicamente nossa única restrição para identificar os pacotes será verificar se estão indo para a porta 80 (**-p tcp --dport 80**) do gateway. Se quiséssemos restringir o acesso a apenas um computador, bastaria criar uma regra de identificação por origem com o IP do micro (a opção **-s**), mas não é o caso.

Nosso iptables fica, então, bem simples e auto-explicativo:

iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to 192.168.0.2:80

Acompanhemos o trajeto dos pacotes:

Origem: 123.45.67.89:6767 (computador D)
Destino: IP_EXTERNO:80

Entrando na cadeia PREROUTING, a nossa regra iptables identifica o pacote e realiza o DNAT, desviando-o para 192.168.0.2:

Origem: 123.45.67.89:6767
Destino: 192.168.0.2:80

O gateway manterá internamente a seguinte relação:
IP_EXTERNO:80 <-> 192.168.0.2:80

O pacote segue para o roteamento e, em seguida, para a cadeia POSTROUTING. Como não há lá nenhuma regra em que esse pacote se encaixa, o pacote sai para o nosso servidor de internet.

Lá a requisição é processada e o servidor envia a resposta de volta para o IP de origem, isto é, 123.45.67.89:6767:

Origem: 192.168.0.2:80
Destino: 123.45.67.89:6767

Como o destino é um IP externo, o pacote de resposta vai para o gateway, fazendo o caminho inverso, passando pela cadeia POSTROUTING (onde não se modifica), roteamento e, em seguida, PREROUTING, onde foi armazenada a relação IP_EXTERNO:80 <-> 192.168.0.2:80. O pacote é corrigido e segue de volta para o computador D:

Origem: IP_EXTERNO:80
Destino: 123.45.67.89:6767

Isso funcionará muito bem para um computador de fora da nossa rede acessar nosso servidor; porém, não funcionará se quisermos acessá-lo internamente através do gateway. Vamos entender por quê.

--- Exemplo 4: acessando 192.168.0.2 internamente ----

De um computador interno, podemos acessar o 192.168.0.2 de várias maneiras:

- Normalmente, através do proxy squid;
- Sem proxy, através do gateway (usando seu ip interno ou externo);
- Sem proxy, diretamente para 192.168.0.2.

Desta última forma, basta configurar o browser para não usar proxy para 192.168.0.2 e a comunicação será direta entre os dois computadores, sem passar pelo gateway e sem problemas adicionais.

Podemos também acessar através do gateway, já que o gateway está desviando todo acesso à porta 80 para 192.168.0.2. Para isso, não podemos usar proxy, que envia os pacotes para a porta 3128, em vez da 80. Podemos endereçar o gateway tanto pelo seu IP interno quanto pelo externo; dos dois modos os pacotes chegarão lá e serão direcionados a 192.168.0.2. Mas isso não é suficiente para a comunicação funcionar, quando o pacote vem da rede interna.

Acompanhemos uma requisição a partir de 192.168.0.6, via gateway, sem proxy:

Origem: 192.168.0.6:912
Destino: 192.168.0.1:80 (poderia ser IP_EXTERNO:80)

Ao emitir o pacote, o cliente fica aguardando, na porta 912, uma resposta de 192.168.0.1:80 (ou IP_EXTERNO:80, se for o caso).

Ao passar pelo PREROUTING, muda para:

Origem: 192.168.0.6:912
Destino: 192.168.0.2:80

Sem regras adicionais no POSTROUTING, o pacote segue para o servidor.

Lá, o servidor manda uma resposta para:

Origem: 192.168.0.2:80
Destino: 192.168.0.6:912

Como o destino é outro computador da rede interna, o pacote irá direto para ele! O

pacote de resposta não vê necessidade de passar pelo gateway; só passaria por lá se o destino fosse um IP externo.

O pacote chega a 192.168.0.6. Mas, como este estava esperando uma resposta de 192.168.0.1 (ou IP_EXTERNO), o pacote de resposta não é reconhecido e é desprezado.

A solução é acrescentar uma regra na cadeia POSTROUTING mudando a origem para 192.168.0.1 quando for acessado o 192.168.0.2:80 a partir de um micro interno (192.168.0.0/24). Isso forçará o pacote a retornar pelo gateway.

iptables -t nat -A POSTROUTING -p tcp --dport 80 -s 192.168.0.0/24 -d 192.168.0.2 -j SNAT --to:192.168.0.1

(Funcionaria também com **-j MASQUERADE**, como já vimos.)

Este comando complementa o do exemplo anterior, devendo ambos serem usados em conjunto.

Finalmente, se quisermos usar o proxy, devemos lembrar que o pacote não será afetado pelas regras do PREROUTING, uma vez que está sendo direcionado para a porta 3128 em vez da 80:

Origem: 192.168.0.6:3537
Destino: 192.168.0.1:3128
Requisição: http://192.168.0.1/intranet

O squid, então fará uma requisição para o próprio gateway, criando um pacote que passará pelo caminho 3, isto é, pela cadeia OUTPUT. É lá que devemos incluir uma regra DNAT para desviar os pacotes com destino à porta 80 para 192.168.0.2:80.

iptables -t nat -A OUTPUT -p tcp --dport 80 -j DNAT --to 192.168.0.2:80

Isso pode ainda não funcionar, dependendo da configuração do squid.

O squid tem uma opção chamada tcp_outgoing_address, com a qual é programado o IP de saída do pacote do gateway, de uma requisição do proxy.

Ele pode estar configurado apenas para requisições à internet, usando o IP_EXTERNO (veja /etc/squid/squid.conf):

tcp_outgoing_address IP_EXTERNO

Com esse IP, o pacote será endereçado à interface externa, mesmo se forçássemos uma mudança do IP de origem com SNAT, mas isso só pode ser feito no POSTROUTING, ou seja, depois de o pacote ter sido roteado para a interface externa.

Para solucionar isso, basta configurar o squid para que as requisições direcionadas a um micro da rede interna tenha o IP de sua interface interna, ou seja, no caso, 192.168.0.1. Fazemos isso da seguinte maneira:

acl destino_interno dst 192.168.0.0/24
tcp_outgoing_address 192.168.0.1 destino_interno
tcp_outgoing_address IP_EXTERNO

que diz ao squid que todo pacote direcionado a um IP interno deve sair com IP de origem 192.168.0.1, e que o resto deve sair como IP_EXTERNO.

As alterações devem ser feitas no arquivo /**etc**/**squid**/**squid.conf**. Para que a nova configuração entre em vigor, execute:

/etc/init.d/squid reload

--- Exemplo 5: VNC em mais de uma máquina e com acesso restrito ----

O exemplo 3 também pode ser adaptado para acesso a outros servidores na rede, como pcAnywhere, VNC, SQL Server, etc. Basta saber o IP e a porta em que cada aplicativo está à escuta e fazer as adaptações, lembrando que é possível restringir o acesso a apenas alguns computadores para aumentar a segurança, conforme veremos no exemplo abaixo.

Suponha que dois computadores de nossa rede precisem ser acessados externamente via VNC, digamos, 192.168.0.10 e 192.168.0.33. Como o VNC é um tanto perigoso, vamos permitir o acesso a apenas alguns computadores:

- O 192.168.0.10 poderá ser acessado apenas pelo 189.0.10.13;
- O 192.168.0.33 poderá ser acessado apenas por 64.16.211.1 e 64.16.211.7.

Da mesma forma que no exemplo 3, os computadores externos deverão endereçar suas requisições para IP_EXTERNO, e o gateway desviará para os computadores da rede interna.

Mas como diferenciar qual requisição vai para o micro 192.168.0.10 e qual vai para o 192.168.0.33? A resposta é simples: basta configurar uma porta para cada um no gateway.

O UltraVNC costuma usar a porta 5900. Podemos criar a seguintes regras:

A porta 5900 no gateway desvia para 192.168.0.10:5900;
A porta 5901 no gateway desvia para 192.168.0.33:5900 (sim, 5900, e não 5901; assim não precisamos ficar alterando as configurações do VNC em cada computador; imagina se tivéssemos que fazer isso em 200 máquinas!).

Nossos iptables ficariam assim:

Liberando acesso VNC para 189.0.10.13 ao computador 192.168.0.10 pela porta 5900:

iptables -t nat -A PREROUTING -p tcp --dport 5900 -s 189.0.10.13 -j DNAT --to 192.168.0.10:5900

Liberando acesso VNC para os computadores 64.16.211.1 e 64.16.211.7 ao computador 192.168.0.33 pela porta 5901:

iptables -t nat -A PREROUTING -p tcp --dport 5901 -s 64.16.211.1 -j DNAT --to 192.168.0.33:5900

iptables -t nat -A PREROUTING -p tcp --dport 5901 -s 64.16.211.7 -j DNAT --to 192.168.0.33:5900

Esses computadores externos deverão acessar o VNC pelo IP externo com portas 5900 e 5901, respectivamente. Qualquer outro computador que tentar acessar essas portas não se encaixarão nas regras do PREROUTING, não sofrerão NAT e não serão desviados. O gateway pensará que a requisição está indo para ele. Mas, como ele não está escutando nessas portas, os pacotes serão desprezados. Isso impede tentativas de acesso feitas por computadores não autorizados, aumentando a segurança.

--- Conclusão ---

Espero que essa introdução ao NAT com iptables tenha sido didática o suficiente para que você tenha entendido o caminho dos pacotes e possa, a partir disso, criar os próprios comandos iptables para as necessidades de sua rede. Eu gostaria de ter feito desenhos e animações, mas é provável que eu venha acrescentar essas coisas no futuro.

O efeito dos comandos iptables é perdido após o servidor ser reiniciado, de forma que é necessário colocá-los em um script que seja executado durante o boot.

Para saber quais regras iptables estão em vigor, use **iptables *tabela* -L**, isto é:

iptables -t nat -L

Para acompanhar o trajeto dos pacotes, use o comando **tcpdump**, restringindo a um IP e/ou uma porta específica:

tcpdump -n -i any host 192.168.0.2 and port 80

Isso irá monitorar todo tráfego destinado ao nosso servidor interno de internet. **-n** significa mostrar números de IPs em vez de nomes das máquinas; acho mais fácil trabalhar assim. **-i any** significa monitoração em todas as interfaces.

Há um tutorial muito interessante (porém longo e em inglês) sobre iptables em

http://iptables-tutorial.frozentux.net, escrito por Oskar Andreasson - meus agradecimentos a ele.

Embora este seja apenas um post, ficou bem grandinho. Como disse um amigo meu, "the book is on the iptables".

Postado por Sony Santos às 17:29:00 0 comentários Links para esta postagem
Marcadores: iptables, linux, técnico

Quinta-feira, 13 de Março de 2008

A série Cosmos está de volta!

Pena que descobri meio tarde para quem queria ver todos os episódios, mas a TV Escola está reapresentando a série, em vários horários, de 10 a 13/3 e de 9 a 11/4.

Para quem não conhece, Cosmos é uma série científica apaixonante, sobre astronomia, biologia, etc., apresentada e co-escrita por Carl Sagan, que fez sucesso na década de 80, levando uma geração inteira a se apaixonar pela ciência e o estudo do universo. Essa série foi uma das principais responsáveis por minha opção por física.

Hoje serão apresentados 2 episódios seguidos, de 1h cada, nos seguintes horários:

7h às 9h / 9h às 11h / 13h às 15h / 17h às 19h / 22h às 24h

Mais informações sobre a programação da TV Escola pode ser obtida aqui.

Postado por Sony Santos às 09:04:00 0 comentários Links para esta postagem
Marcadores: ciências

Segunda-feira, 10 de Março de 2008

Ache no Google (Radio Gaga)

Cante o refrão de "Radio Gaga" (Queen) com uma nova letra:

"Cê não sabe,

Ache no Google!
Não me pergunte,
Ache no Google!
Cê não sabe,
Ache no Google!
Ache no Google!

Lá tem de tudo
O que quiser,
Tem de tudo, sim!"

(Por mim e minha esposa.)

Postado por Sony Santos às 09:00:00 0 comentários Links para esta postagem
Marcadores: paródias

Quarta-feira, 5 de Março de 2008

Não às pesquisas com células-tronco embrionárias

Ontem o Jornal Nacional fez uma reportagem enorme defendendo as pesquisas com células-tronco embrionárias, isto é, que criam seres humanos e os matam no início do seu desenvolvimento para roubar-lhe as células.

Isso está em discussão em Brasília.

Nós, católicos de verdade, que participamos ativamente na Igreja e temos ciência do valor da vida humana, somos contra esse tipo de pesquisa. Apoiamos as pesquisas com células-tronco adultas, tiradas da medula ou de outras partes da pessoa adulta, e que podem ser usadas para os mesmos tipos de terapias das células embrionárias. O Brasil está bastante avançado na pesquisa de células-tronco adultas.

Portanto, rigorosamente não há necessidade de pesquisa com células-tronco embrionárias. Não se justifica curar alguém às custas da vida de outra pessoa.

Postado por Sony Santos às 09:19:00 0 comentários Links para esta postagem
Marcadores: valorização da vida

Segunda-feira, 25 de Fevereiro de 2008

Conectando-se à Conectividade Social

Tenho visto pela internet que a conexão à *Conectividade Social* da Caixa tem sido um problema para muita gente, e que as soluções apresentadas tendem a variar de caso para caso.

Onde trabalho, até agora só era possível acessar a Conectividade Social pela internet discada, que não impunha nenhuma barreira entre a máquina local e os servidores da Caixa. Nosso sonho, no entanto, sempre foi acessar a CS pelo ADSL (Speedy) compartilhado na rede interna, mas nunca havíamos conseguido.

Depois de muito pesquisar, tentativas-e-erros e várias descobertas, cheguei a uma solução, e resolvi compartilhar. A solução é atrelada à configuração da nossa rede, e pode não funcionar em outras configurações; no entanto, o procedimento descrito aqui poderá fornecer pistas para quem quiser adaptar em sua rede. Boa sorte!

Nossa rede: 192.168.0.0/24, gateway 192.168.0.1, firewall iptables, proxy squid não transparente na porta 3128; computadores locais Windows XP com avast! antivírus.

Passos para liberar a Conectividade Social:
1. Liberar acesso aos IPs da Caixa, isto é, permitir que os computadores locais acessem diretamente os IPs da Caixa, sem passar pelo proxy squid (para quem não consegue acesso via squid). Isso deve ser feito no firewall. Como aqui não é proxy transparente, a regra que funcionou foi:

iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -d 200.201.160.0/20 -j MASQUERADE

2. Configurar os computadores locais para não usarem o proxy squid para os sites da Caixa. Este é o complemento da etapa anterior. O objetivo é o mesmo, mas lá é mostrada a parte da configuração feita na máquina do proxy, e aqui na máquina do cliente que for usar a CS. Quem usa proxy transparente não precisa desta etapa.

Painel de Controle -> Opções da Internet -> Conexões -> Configuração da LAN -> Avançadas -> Não usar proxy para endereços iniciados por (lá embaixo)

Acrescente nesse campo o endereço *.caixa.gov.br, separando com ";". Por exemplo, se nesse campo estiver escrito 192.168.0.1, vai ficar assim: 192.168.0.1;*.caixa.gov.br

Confirme clicando em Ok até fechar as opções de internet.

3. Liberar a porta 80 do computador (que for usar a CS). A porta 80 pode estar sendo ocupada por alguns programas, como o apache e o avast!. O apache dificilmente estará sendo usado, e desabilitá-lo não ocasionará problemas. O problema é desabilitar o antivírus. Você não pode ficar sem proteção.

Porém, se o avast está usando a porta 80, então ele não está protegendo os computadores de sua rede! Pois os computadores da sua rede acessam o squid pela porta 3128, e não pela porta 80! Então o avast deveria estar protegendo a porta 3128!

Corrigindo isso, você mata 2 coelhos com uma só cajadada: libera a porta 80 para a CS e protege o computador na porta 3128.

Para isso, clique na bolinha do avast na bandeja do sistema (a bolinha azul com a letra "**a**"). Na janela que abrir, se houver um botão escrito "Detalhes... >>", clique nele; senão, não se preocupe.

Na coluna de "Provedores instalados", selecione a "Proteção da Internet", e clique em Personalizar. No campo "Redirecionar porta(s) HTTP", coloque a porta do seu proxy squid (no caso 3128), e confirme com Ok. Isso resolve este último passo.

Com isso consegui acesso à CS via banda larga. Espero que lhe seja útil.

Postado por Sony Santos às 18:34:00 0 comentários  Links para esta postagem
Marcadores: iptables, squid

Sexta-feira, 18 de Janeiro de 2008

Proibido pra mim

Justiça achou meu argumento engraçado
Proibido pra mim, no way!
Disse que eu não podia jogar
Nem levou a sério o que falei

Eu vou fazer de tudo que eu puder
Eu vou roubar essa lan house pra mim
Vou poder jogar a qualquer hora
E ai de quem me impedir

Sem vocês
Como é que eu vou ser aprendiz?
Sem vocês
Como é que eu vou ser aprendiz?
de Guerra!

Eu me flagrei pensando em vocês:
CS, GTA, BOPE e EverQ
Em uma noite especialmente boa
Mas a justiça bloqueou meu lazer...

Eu vou fazer de tudo que eu puder
Eu vou roubar essa lan house pra mim
Vou poder jogar a qualquer hora
E ai de quem me impedir

Sem vocês
Como é que eu vou ser aprendiz?
Sem vocês
Como é que eu vou ser aprendiz?
de Guerra!

Notícia-musa: Counter-Strike e Everquest proibidos no Brasil

Música original: Proibida pra mim (Charlie Brown Jr.)

Postado por Sony Santos às 14:19:00 0 comentários Links para esta postagem
Marcadores: humor, paródias

Segunda-feira, 14 de Janeiro de 2008

"Setor elétrico vive de boatos"

Quando vi a manchete "Lula nega apagão e diz que setor elétrico vive de boatos", não consegui evitar certas interpretações.

Uma manchete alternativa seria: "Setor elétrico contribui para a economia de energia elétrica, vivendo só de boatos".

Ou ainda: Lula tranqüiliza o povo e garante que não haverá apagão. "Como o setor elétrico vive de boatos", afirma, "para suprir a demanada basta contratar fofoqueiros e especuladores, porque isso não está em falta".

Postado por Sony Santos às 13:08:00 0 comentários Links para esta postagem
Marcadores: crônicas, humor

Segunda-feira, 7 de Janeiro de 2008

Objetivo e Meta são a mesma coisa

Fácil comprovar. Vá nas ferramentas de idiomas do Google e traduza "objetivo" para inglês. Depois, traduza o resultado novamente para o português, usando a mesma ferramenta do Google e, tchanaaaan, você obterá "meta". CQD!

Um amigo esteve à procura da diferença entre esses dois termos. Ora, eles têm o mesmo significado. Se as pessoas querem fazer uso diferente desses termos, essa diferença é subjetiva e pessoal.

Ter um objetivo a alcançar é o mesmo que ter uma meta a atingir, seja ela quantitativa ou não, seja a meta principal ou parte de uma maior. Ambas as palavras podem ser encontradas no dicionário como "alvo".

Um objetivo a alcançar é um projeto a realizar. Da mesma forma que um projeto

pode ser dividido em partes e sub-partes, tarefas e sub-tarefas, um objetivo maior, principal e último pode ser alcançado após a realização de passos intermediários menores, mas cada um desses passos é um pequeno objetivo. Em outras palavras, cada passo é uma pequena meta dentro (como parte necessária) de uma meta maior, principal e final que se persegue. Chamar um de objetivo e outro de meta será apenas uma convenção pessoal e de alguns autores; não existe um padrão real em nosso idioma.

De qualquer forma, se você assumir uma definição, mantê-la e usá-la, poderá tornar mais fácil e prática a organização de seus projetos.

Postado por Sony Santos às 18:47:00 0 comentários Links para esta postagem
Marcadores: língua portuguesa

O número para o qual discou não existe

Esta Operadora de Telefonia Informa:

O número para o qual você discou não existe.

Não está na lista telefônica, não faz parte do conjunto dos números reais, e não foi citado por nenhum matemático famoso.

Por favor, invente um outro número e tente novamente, mas desta vez invente um número que já existe.

Esta Operadora de Telefonia agradece a compreensão.