Uma plataforma para monitoração de malhas BGP

Luis Balbinot, Liane Tarouco

Programa de Pós-Graduação em Computação
Universidade Federal do Rio Grande do Sul – Instituto de Informática
Caixa Postal 15064 – 90501-970 – Porto Alegre, RS

{hades,liane}@inf.ufrgs.br

Abstract. *The continuing rapid growth of the Internet brings us routing complexity as well as global connectivity. The volume of routing information, including inappropriate information, is getting larger. Many network administrators check the routing information in their border routers manually using the command line interface as their daily job. To reduce the burden of network administrators, a tool is desired which monitors BGP (Border Gateway Protocol) routing information, so that it can provide network managers with more high level and summarized information and, potentially, identify and alert routing anomalies automatically. This paper describes the design of a monitoring platform for BGP meshes. The platform is able to establish peering sessions with BGP neighbors on the Internet routing mesh, so that it is able to collect and store all BGP messages exchanged in a database for off-line analysis and, at the same time, keep a local BGP routing table for on-line analysis. A set of measurements are presented to illustrate some real-life applications of the platform.*

Resumo. *O crescimento rápido e contínuo da Internet nos traz uma maior complexidade de roteamento e de conectividade global. O volume de informações de roteamento, incluindo informações inapropriadadas, está aumentando. Muitos administradores de redes verificam manualmente essas informações em seus roteadores de borda através de interfaces de linha de comando. Para reduzir a carga de trabalho desses administradores, uma ferramenta capaz de monitorar as informações de roteamento BGP (Border Gateway Protocol) é desejável, de forma a que ela seja capaz de apresentar informações de mais alto nível e, até mesmo, identificar e alertar sobre anomalias automaticamente. Este artigo descreve o projeto de uma plataforma de monitoração de malhas de roteamento BGP. A plataforma é capaz de estabelecer vizinhança com roteadores da malha BGP da Internet, conseguindo assim coletar e armazenar todas as mensagens BGP que são trocadas em um banco de dados para análise off-line e, ao mesmo tempo, consegue manter uma tabela de roteamento para análise on-line. Uma série de medições também são apresentadas, para exemplificar alguns usos práticos da plataforma.*

1. Introdução

Um Sistema Autônomo (AS, ou *Autonomous System*) é composto por um conjunto de roteadores e enlaces controlados por uma única instituição. O roteamento entre os ASs depende do *Border Gateway Protocol* (BGP) [1], um algoritmo de caminho-distância que realiza anúncios de blocos de endereços, ou prefixos. Cada prefixo consiste de um endereço de 32 bits e um comprimento de máscara (p. ex., 200.248.0.0/16 consiste de endereços que variam entre 200.248.0.0 até 200.248.255.255). Os ASs que são vizinhos utilizam o BGP para trocar mensagens de atualização nos anúncios de alcançabilidade dos diversos prefixos que estão nas suas tabelas de roteamento. Um roteador pode enviar um anúncio de uma nova rota ou pode fazer a remoção de uma rota que não esteja mais

disponível. Cada anúncio inclui a lista de ASs no caminho até o destino, juntamente com uma série de atributos. Já que os roteadores só enviam os anúncios quando algo se modifica, pode-se dizer que a Internet poderia atingir um estado "estável", onde os roteadores não mais enviariam informações de atualização de rotas. Mas, no entanto, a tabela de roteamento BGP da Internet está muito longe de ser estável.

As mudanças no roteamento BGP ocorrem por várias razões. A troca de mensagens de atualização depende da existência de uma sessão BGP ativa entre um par de roteadores. Uma falha de equipamentos ou de configuração pode ocasionar no fechamento dessa sessão BGP, forçando que cada roteador remova todas as rotas aprendidas de seu vizinho. Após ser restabelecida a sessão, os roteadores realizam a troca de suas informações de roteamento novamente. Cada roteador aplica uma série de políticas locais para selecionar a "melhor" rota para cada prefixo e para decidir quando realizar o anúncio dessa rota para os seus vizinhos. As mudanças dessas políticas podem ocasionar novos anúncios. Um grupo de ASs pode possuir políticas conflitantes, de forma que uma cadeia de anúncios/remoções de rotas possa acontecer repetidamente [2, 3]. Além disso, o roteamento intra-domínio ou mudanças de topologia podem fazer com que alguns roteadores escolham novas rotas BGP e façam o seu anúncio para ASs vizinhos.

As mudanças na malha de roteamento BGP podem causar diversos problemas de desempenho. Um único evento, como, por exemplo, a queda de um enlace, pode disparar uma grande seqüência de atualizações, à medida que os roteadores começam a escolher novos caminhos. Durante esse período de convergência, os pacotes que iam ao prefixo destino (que agora está inalcançável) podem ficar presos em laços de roteamento. A troca e o processamento de mensagens de atualização de rotas consomem largura de banda e recursos de CPU nos roteadores que "falam" BGP. Além disso, os novos anúncios vindos de ASs vizinhos podem causar uma mudança no caminho que o tráfego segue, podendo causar congestionamentos em alguns enlaces do AS. As mudanças muito freqüentes nos anúncios vindos de outros domínios também dificultam a engenharia de tráfego dentro do AS. Por exemplo, uma mudança em um anúncio BGP pode fazer com que o tráfego para um determinado prefixo saia do AS através de um roteador de borda diferente. Se as mudanças no roteamento BGP afetam uma grande parte do tráfego, informações sobre os anúncios BGP antigos não são uma boa base histórica para decisões operacionais futuras.

Para realizar a manutenção de um AS, os administradores das redes normalmente possuem acesso aos roteadores de borda através de interfaces de linha de comando, que são bastante rudimentares. A quandidade de informação que um administrador deve processar para compreender o estado de um único anúncio de prefixo é relativamente grande. Um sistema de monitoração dessas informações, capaz de armazenar todas as mensagens BGP que são trocadas dentro de um AS, bem como manter uma tabela de roteamento atualizada, é desejável para facilitar a operação de um AS. Este artigo apresenta uma plataforma de monitoração de malhas BGP capaz de fazer parte ativa de uma malha de roteamento, capturando e armazenando todas as mensagens de atualização trocadas entre vizinhos BGP, bem como também armazena toda a tabela de roteamento BGP que é mantida localmente pela própria plataforma. Um histórico de todos os anúncios e tabelas de roteamento pode ser consultado, visando-se compreender melhor como determinados anúncios estão sendo recebidos e aproveitados com o passar do tempo. A plataforma é bastante modular, permitindo que uma série de ferramentas de análise dessas informações sejam adicionadas de uma forma relativamente simples e sem interromper o processo de monitoração (permitindo, inclusive, o uso remoto dessas informações). Além de apresentar a plataforma, uma série de medições feitas com dados reais obtidos de ASs do núcleo da Internet, que foram injetados artificialmente na plataforma, são apresentadas como exemplos práticos da aplicação da platorma em ambientes reais.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta diversas abordagens existentes para realizar a monitoração de malhas BGP; a Seção 3 apresenta a plataforma que é proposta neste trabalho, descreve seu funcionamento e detalha todos os seus componentes internos; a Seção 4 ilustra, através de uma série de medições, como a plataforma pode ser utilizada para facilitar na administração de uma rede que utiliza o protocolo BGP; ao final, na Seção 5, o trabalho é concluído, deixando em aberto alguns pontos que podem ser abordados como trabalhos futuros.

2. Abordagens para monitoração

Existem várias abordagens diferentes que podem ser adotadas para realizar a captura das informações de roteamento BGP. Neste trabalho é feita a distinção entre três tipos: *passiva*, *ativa e intrusiva*, e *ativa e não intrusiva*. As *passivas* estão baseadas em mecanismos que não interferem na operação da rede, ou seja, realizam o processo de captura de uma forma praticamente transparente e sem gerar nenhum tráfego *in-band* na infraestrutura de rede. As *ativas e intrusivas*, além de afetarem de alguma forma a operação da rede (gerando sobrecarga em roteadores, p. ex.), também geram tráfego *in-band* para transportar as informações capturadas nas medições. Por fim, as *ativas e não intrusivas* diferem-se das ativas e intrusivas pelo fato de que elas se aproveitam das informações que já estão trafegando pela rede, sem a necessidade de injetar mais tráfego para realizar as medições.

Um passo muito importante deste trabalho foi estudar e encontrar a melhor forma de captura para a extração das informações de mais alto nível que são desejadas. As próximas seções apresentam as abordagens mais comuns para a captura de informações de roteamento BGP que são utilizadas em ambientes reais de monitoração na Internet. As principais vantagens e desvantagens de cada uma são apresentadas brevemente.

2.1. Analisadores

Analisadores, ou *sniffers*, são a forma mais comum de monitoração passiva de redes. O uso de analisadores no processo de obtenção das informações da malha BGP se dá através da captura dos pacotes de uma sessão BGP estabelecida entre dois roteadores. O analisador deve ser capaz de capturar e desencapsular todos os pacotes até o nível de aplicação para conseguir extrair as informações necessárias sobre o BGP. Essa técnica possui alguns problemas. O primeiro deles está na decodificação do BGP, que deve ser feita pelo dispositivo de captura de pacotes, já que praticamente toda uma implementação do protocolo é necessária para que seja possível compreender o conteúdo dos pacotes capturados e também para manter uma tabela de roteamento local atualizada. Tal implementação seria relativamente complexa e talvez inviável. O segundo problema está nas conexões ponto-a-ponto: em redes de barramento é relativamente simples de se instalar um analisador (através do espelhamento de portas ou da captura direta em um barramento compartilhado), mas em enlaces de tecnologias mais complexas (orientadas a circuitos, p. ex.) essa tarefa já é mais difícil e demanda o uso de hardware específico e, normalmente, caro. A alta velocidade dos novos enlaces que estão sendo instalados também representa uma limitação para esta abordagem.

O ponto positivo desta técnica está no fato dela ser passiva e não intrusiva (em alguns casos podendo ser considerada ativa por inserir atrasos e perdas de potência em enlaces).

2.2. MIB BGP

A MIB BGP [4] é a única forma padronizada para a obtenção de informações relavantes sobre o protocolo BGP. Ela foi criada pelo IETF com o objetivo de fornecer um conjunto

de objetos gerenciáveis para a monitoração e configuração (de uma forma bastante limitada) do BGP em roteadores. A MIB é implementada em três tabelas, através das quais é possível obter acesso aos prefixos anunciados, seus respectivos atributos e as vizinhanças estabelecidas com outros roteadores. Existem também duas notificações para informar o estado de uma sessão BGP. Uma das limitações da aplicação dessa MIB na monitoração de uma malha BGP está no fato de não existirem informações armazenadas sobre o conteúdo das mensagens de atualização que são trocadas entre os vizinhos (exceto por alguns contadores). O suporte também é um problema, já que nem todos os roteadores implementam essa MIB (um problema comum na gerência SNMP), e os que implementam alguma forma de gerenciamento de BGP preferem utilizar MIBs proprietárias, dado que a MIB BGP está desatualizada (diversas extensões foram feitas ao protocolo BGP desde que a especificação da MIB foi publicada). Mas o principal problema está no volume de informações: a tabela de roteamento BGP da Internet (no núcleo *default-free*) possui aproximadamente 140.000 prefixos anunciados pela época da publicação deste artigo [5]; o transporte dessa grande quantidade de informações através do protocolo SNMP gera uma sobrecarga considerável, tanto no agente/gerente, que devem manipular *varbinds* de milhares de objetos, quanto na infraestrutura de rede.

Mesmo que essa não seja uma solução completa para a monitoração de uma malha BGP, a MIB BGP pode muito bem ser utilizada para complementar outras abordagens de monitoração, até mesmo em conjunto com outras MIBs. Os alarmes implementados por ela podem ser utilizados como fontes de correlação para outros eventos observados através da coleta das informações com uma outra abordagem.

2.3. Sessões interativas

Pode-se dizer que as sessões interativas são a forma mais proprietária para a obtenção das informações sobre o estado da malha BGP, já que o seu uso está fortemente atrelado ao sistema operacional instalado nos roteadores. O princípio básico dessa técnica está em simular o acesso remoto a um roteador, assim como se ele fosse feito por um operador humano trabalhando no console do equipamento. Isto é feito através de *scripts* especiais que fazem acesso remoto (via *telnet* ou *ssh*) a um roteador da rede, efetuam o processo de *login* e requisitam, via linha de comando, a listagem da tabela de roteamento BGP atual (p. ex., *'show ip bgp'* em um equipamento Cisco). Essa listagem é então armazenada em grandes arquivos de *dump*, que são posteriormente processados e analisados.

Essa abordagem possui a mesma limitação da MIB BGP, já que é impossível de se obter informações sobre as mensagens BGP que são trocadas entre roteadores vizinhos. Além disso, existe o problema da sobrecarga no roteador, já que ele precisa gastar recursos computacionais para atender o requerimento feito pelo *script*, mas isso pode ser resolvido com o uso de equipamentos dedicados, que é o que normalmente acontece na prática. Existe uma sobrecarga também na infra-estrutura de rede: estudos dos *dumps* disponíveis no projeto *Route Views* [6] mostram que alguns *dumps* chegam a ter até 450 MB, quando não compactados. Toda essa informação precisa ser transferida do roteador para a estação de monitoração de 8 a 12 vezes ao dia, dependendo da política de monitoração adotada. Por fim, existe também o problema de pós-processamento desses *dumps*, de forma que as informações textuais apresentadas pelo roteador consigam ser traduzidas de seu formato original, feito para a compreensão humana, para uma formato mais fácil de ser utilizado pelas ferramentas de análise. Apesar de todos os problemas, essa técnica é a mais popular na Internet, e vários projetos se utilizam dela.

2.4. Participação ativa na malha de roteamento

A participação ativa na malha de roteamento é a abordagem mais flexível, escalável e completa de todas. Além de não trazer a complexidade encontrada nas outras abordagens

apresentadas, a participação ativa na malha consegue capturar tanto as mensagens trocadas entre os vizinhos BGP quanto a tabela de roteamento BGP atual. Isso é possível pois a estação que realiza a captura das informações faz parte da malha de roteamento (faz *peering*), assim como um outro vizinho BGP qualquer. Essa abordagem é independente dos recursos implementados pelos equipamentos da malha BGP: não importa a tecnologia de rede, a velocidade dos enlaces ou a marca dos equipamentos, basta que uma sessão BGP seja estabelecida entre a estação de monitoração e um ou mais roteadores da malha, deixando toda a abstração de rede com o protocolo IP. Como o BGP é um protocolo incremental, a tabela de roteamento é enviada por completo somente quando a sessão é estabelecida. Durante todo o tempo de duração da sessão apenas pequenas mensagens de *keepalive* e de atualização de anúncios são trocadas. O processamento da tabela de roteamento é feito localmente na estação de monitoração.

Como o tráfego BGP é inevitável (ao contrário do tráfego de mensagens SNMP, por exemplo) o impacto dessa abordagem na infra-estrutura de rede é potencialmente baixo. A sobrecarga na CPU do roteador que faz vizinhança com a estação de monitoração também é virtualmente nula, já que ele não precisa processar nenhuma mensagem enviada pela estação, que fica em um estado passivo na malha. O roteador só precisa repassar todas as mensagens recebidas de outros vizinhos, o que é uma tarefa relativamente simples. É importante notar que neste ponto talvez exista uma pequena sobrecarga devido à filtros aplicados à distribuição dos anúncios, mas isso não deveria ser feito em primeiro lugar, já que é importante que a estação receba *todos* os anúncios feitos na malha, sem nenhuma filtragem. Mas, por outro lado, é recomendado que o roteador aplique alguns filtros de entrada para ignorar quaisquer anúncios vindos da estação de medição.

Assim como é o caso do uso de analisadores, essa abordagem também necessita de uma implementação do protocolo BGP para operar, já que as mensagens BGP devem ser processadas para gerar a tabela de roteamento. Felizmente existem diversas implementações do protocolo BGP disponíveis para o sistema operacional Unix. Essas implementações estão disponíveis gratuitamente e com o código-fonte aberto, e podem ser facilmente adaptadas para exportar os dados necessários para a plataforma de monitoração. Através da modificação dessas implementações pode-se ter controle sobre a forma como as informações são exportadas. Uma das principais vantagens de se ter esse controle é a de que o pós-processamento dos dados pode ser realizado antes da exportação dos mesmos, economizando assim recursos futuramente: as informações que são exportadas podem ser armazenadas em um formato que pode ser lido diretamente pelas ferramentas que realizam a análise.

3. Plataforma proposta

A plataforma de monitoração proposta é apresentada na Figura 1. Cada um dos componentes, bem como a ligação entre eles, são explicados em detalhes nos próximos parágrafos, de acordo com os números indicados na figura (que aparecem entre parênteses). As ligações que aparecem em pontilhado indicam apenas a possibilidade de comunicação dos componentes, mas que não têm nenhum uso previsto neste trabalho.

O ponto principal de entrada de informações na estação de monitoração se dá pelo módulo de roteamento BGP (implementado pelo software Zebra [7]), que estabelece uma sessão BGP (1) com um ou mais vizinhos da malha de roteamento. Essa sessão deve, preferencialmente, não ter nenhum filtro associado, como é mencionado na seção anterior. A configuração do software de roteamento é feita de forma que ele realize *dumps* da tabela de roteamento e das mensagens BGP trocadas entre ele e seu(s) vizinho(s) (2) em intervalos regulares. O formato do nome dos arquivos de saída gerados pelo Zebra pode

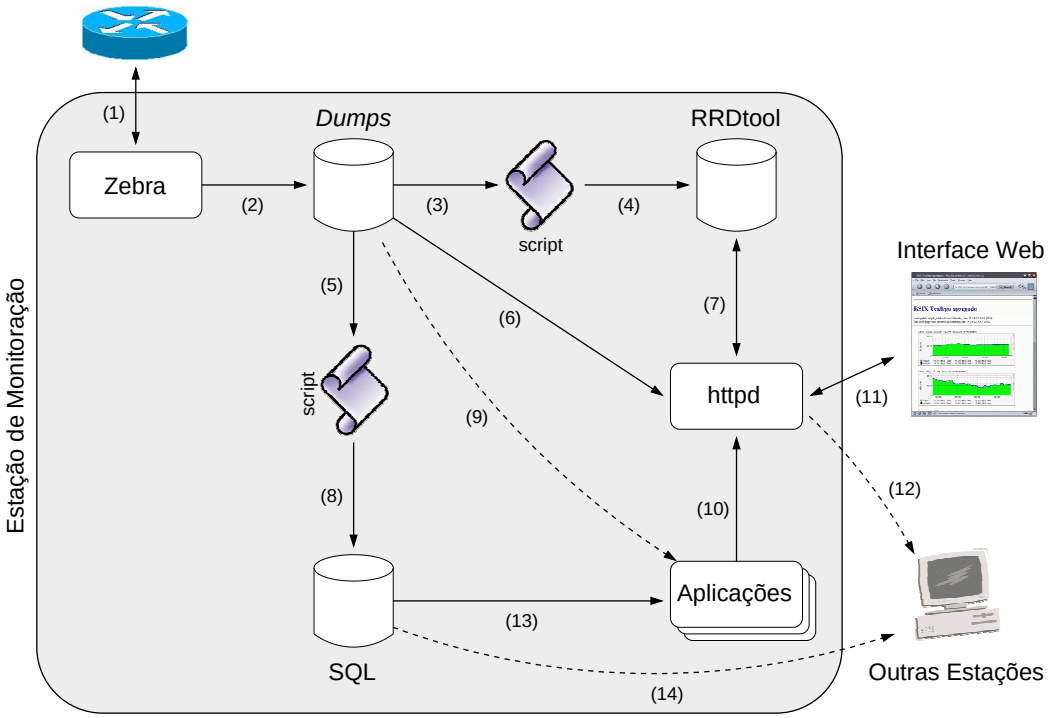

Figura 1: Plataforma de Monitoração

ser configurado de forma a manter uma estrutura de arquivos bem organizada (a Figura 2 ilustra a estrutura de diretórios adotada no protótipo deste trabalho). A localização e identificação dos *dumps* fica mais fácil dessa forma, tanto para os usuários humanos quanto para as ferramentas de análise.

Os *dumps* são, então, acessados por um conjunto de *scripts* (3) que extraem informações relevantes sobre a malha de roteamento. O resultado desses *scripts* é injetado (4) em um banco de dados temporal (neste trabalho foi adotado o RRDtool [8], amplamente utilizado em ferramentas de monitoração de redes), de onde essas informações temporais podem ser extraídas mais tarde para a geração de gráficos. Esse tipo de informação temporal é caracterizado principalmente por estatísticas como, por exemplo, a quantidade de prefixos anunciados, a distribuição por tamanho de prefixo, a quantidade média de endereços por anúncio, enfim, qualquer valor mensurável que possa ser extraído de um *dump*. Assim como acontece em (3), os *dumps* também são processados por um outro conjunto de *scripts* (5) que simplesmente traduzem as informações armazenadas de forma binária (que é o formato original dos *dumps*) para o formato de texto puro e então as injeta em tabelas dentro de um banco de dados SQL (8). A estrutura das tabelas utilizadas para armazenar essas informações, e as informações que são armazendas, são apresentadas nas próximas seções.

É possível que uma estação remota obtenha os *dumps* originais que estão armazenados na estação de monitoração, já que os diretórios contendo esses *dumps* ficam disponíveis através de um servidor HTTP (6), que também está executando na estação de monitoração. Dessa forma, outras pessoas (12) podem ter acesso aos dados capturados com a finalidade de realizar outros relatórios, análises, pesquisas, enfim, vários estudos diferentes, de uma forma similar à que ocorre no projeto *Route Views*, que disponibiliza publicamente todos os dados capturados. O mesmo é possível com as informações que são armazenadas no banco de dados SQL mas, nesse caso, usando os recursos e protocolos específicos do banco de dados em uso (14). O servidor HTTP também pode ter acesso as informações armazenadas no banco de dados temporal do RRDtool (7) através do sistema de arquivos local. Isso é possível através das APIs implementadas pelas ex-

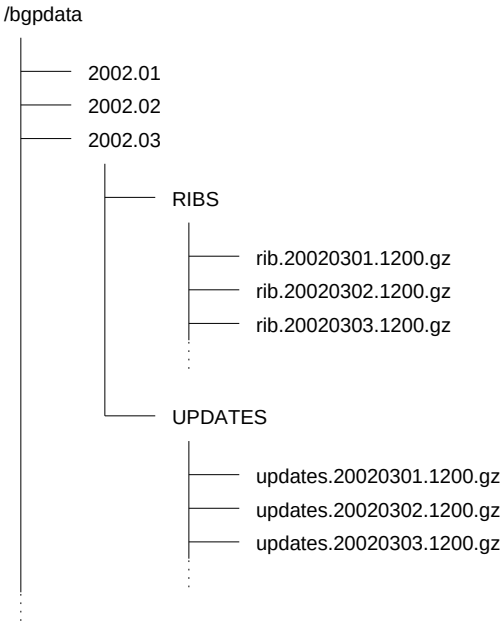

Figura 2: Estrutura de diretórios para armazenamento dos *dumps*

tensões de programação disponíveis no servidor HTTP (com o suporte a linguagens como Perl e PHP). Como é característica do RRDtool, os gráficos dos dados temporais podem ser gerados em demanda e em tempo real, evitando que eles tenham que ser gerados continuamente (escalonados em um processo do *cron*, p. ex.), mesmo que não exista nenhum interesse em visualizá-los, economizando recursos computacionais na estação de monitoração.

As aplicações que analisam as informações coletadas têm acesso aos dados através da API do banco de dados SQL (13). Também é possível que elas façam acesso direto aos *dumps* (9), caso necessário. Essas aplicações podem ser escritas em diversas linguagens, mas todas têm em comum o fato de que a sua saída é sempre feita através do servidor HTTP (10). Elas também podem ser ativadas periodicamente (gerando relatórios continuamente, mesmo que ninguém queira acessá-los) ou podem ser disparadas em demanda (quando o usuário fizer um pedido de relatório pela interface web). É através da interface web que todas as informações geradas e obtidas pela estação de monitoração são acessadas (11). Está fora do escopo deste trabalho o desenvolvimento de todo um sistema integrado para o controle dessas informações, já que o seu foco está na metodologia de captura e nos resultados que podem ser obtidos. Dessa forma, os dados que são visualizados pela interface web ficam restritos aos documentos que estão na estrutura de diretórios do servidor HTTP, onde as aplicações simplesmente "jogam" as suas saídas (arquivos HTML, XML, imagens, texto puro, etc.), sendo responsabilidade de cada aplicação cuidar do formato de seus arquivos saída. De certa forma essa abordagem fornece um maior grau de flexibilidade, deixando a semântica das informações aos cuidados das aplicações.

3.1. O software GNU Zebra

O software GNU Zebra é composto por um conjunto de *daemons* que implementam os protocolos de roteamento TCP/IP mais comuns. O software, que é *multi-threaded* e modular, inclui uma interface completa para o desenvolvimento de novos módulos de roteamento. O Zebra também é utilizado como implementação de referência por fabricantes de roteadores, já que o seu código-fonte é distribuído segundo a licensa GNU GPLv2 [9]. O software é compatível com a maioria dos sistemas Unix no mercado.

No contexto da monitoração da malha BGP, o papel do Zebra é muito simples: ele

```
01: route-map nothing deny 1
02: !
03: router bgp 1916
04:  bgp router-id 200.132.0.15
05:  neighbor 200.132.0.17 remote-as 1916
06:  neighbor 200.132.0.17 activate
07:  neighbor 200.132.0.17 update-source 200.132.0.15
08:  neighbor 200.132.0.17 route-map nothing out
09: !
10: dump bgp updates /bgpdata/%Y.%m/UPDATES/updates.%Y%m%d.%H%M 15m
11: dump bgp routes-mrt /bgpdata/%Y.%m/RIBS/rib.%Y%m%d.%H%M 2h
12: !
13: end
```

Figura 3: Configuração do GNU Zebra

deve ser configurado para estabelecer vizinhança com um ou mais roteadores do núcleo da malha que deseja-se monitorar e então programado para exportar todas as informações que são coletadas e processadas pelo seu módulo de roteamento BGP. A Figura 3 ilustra um exemplo de configuração utilizada em alguns testes. Os comandos de configuração do Zebra são praticamente os mesmos utilizados em roteadores da marca Cisco. Nas linhas 5-8 a vizinhança BGP é estabelecida, nesse caso entre a estação de monitoração e um roteador pertencente a Rede Nacional de Pesquisa (RNP) no POP-RS (AS1916) – a sessão nesse caso é I-BGP, já que a estação também está no AS1916 (linha 3). Na linha 8 é aplicado um *route-map* (um tipo de filtro de anúncios) que bloqueia todos os anúncios originados pela estação de monitoração (vide definição do *route-map* na linha 1). Dessa forma é possível forçar a estação a ficar em modo "passivo" na malha, evitando problemas caso algum usuário malicioso consiga acesso indevido e comece a propagar anúncios espúrios na malha.

A parte da configuração que mais interessa para a monitoração está nas linhas 10 e 11, onde é configurada a geração dos *dumps*. Na linha 10 é programada a geração dos *dumps* das mensagens de *update* trocadas entre a estação e seu(s) vizinho(s). O mesmo é feito na linha 11, só que para os *dumps* da tabela de roteamento. É importante notar que ambas as linhas possuem caracteres especiais para controle do nome do arquivo gerado, onde esses caracteres são substituídos na hora da criação do arquivo (por exemplo, uma seqüência %Y é substituída pelo ano e %H pela hora), possibilitando que seja mantida uma estrutura de *dumps* mais organizada, como já foi ilustrado na Figura 2. Nesse exemplo, o *dump* dos *updates* é feito a cada 15 minutos e o *dump* da tabela de roteamento é feito a cada 2 horas. O intervalo entre a geração dos *dumps* depende das políticas adotadas em cada sítio de monitoração.

3.2. Banco de dados

Para armazenar as informações obtidas dos *dumps* foram projetadas duas tabelas que são implementadas no banco de dados SQL. Considerando-se um pior caso, onde a malha BGP trabalha em *full route*, o número de entradas nas tabelas é muito grande e espera-se que elas tenham milhões de entradas. Como solução de banco de dados foi adotado o MySQL [10], que tem como característica principal de seu projeto o uso em ambientes com grandes quantidades de informação onde o desempenho no acesso é um fator crítico. A Figura 4 ilustra a estrutura das tabelas *Messages*, onde são armazenados os dados relevantes sobre as mensagens trocadas entre os vizinhos (*updates*), e *Tables*, onde são armazenadas as tabelas de roteamento (*ribs*). As tuplas foram projetadas de forma que seja possível acessar todas as informações de um mesmo *dump*, bastando que se saiba a hora e data da sua captura, mantendo-se, portanto, a informação temporal dos anúncios.

O desempenho no acesso ao banco de dados é um ponto de grande importância neste trabalho, já que a quantidade de informações armazenadas é potencialmente grande.

Messages
Seq : MEDIUMINT UNSIGNED
Time : TIMESTAMP(14)
Type : ENUM('A','W')
PeerAddr : CHAR(15)
PeerAS : SMALLINT UNSIGNED
PrefixAddr : CHAR(15)
PrefixLen : TINYINT UNSIGNED
ASPath : TEXT
Origin : ENUM('?','I','E')
NextHop : CHAR(15)
LocalPref : SMALLINT UNSIGNED
MED : SMALLINT UNSIGNED
Community : TEXT

Tables
Seq : MEDIUMINT UNSIGNED
Time : TIMESTAMP(14)
FromAddr : CHAR(15)
FromAS : SMALLINT UNSIGNED
PrefixAddr : CHAR(15)
PrefixLen : TINYINT UNSIGNED
ASPath : TEXT
Origin : ENUM('?','I','E')
NextHop : CHAR(15)
LocalPref : SMALLINT UNSIGNED
MED : SMALLINT UNSIGNED
Community : TEXT
Agg : ENUM('Y','N')
AggAddr : VARCHAR(15)
AggAS : SMALLINT UNSIGNED

Figura 4: Estrutura das tabelas que armazenam as informações dos *dumps*

Todas colunas das tabelas no banco de dados são suficientes para armazenar exatamente a quantidade de informação que lhes diz respeito, sem desperdícios. Essa é uma otimização simples, mas que reflete em ganhos consideráveis na economia de espaço de armazenamento e velocidade de acesso com o passar do tempo. Também é importante notar que essas tabelas podem crescer demasiadamente. É recomendado que existam processos periódicos de poda dessas tabelas, que pode ser feito de forma temporal, já que existe informação de tempo associada às tuplas. Para evitar que as informações sejam perdidas com as podas, elas podem ser simplesmente transferidas para outras tabelas dedicadas unicamente à manutenção de dados históricos.

3.2.1. Tabela *Messages*

Como já foi mencionado, a tabela *Messages* armazena as informações coletadas nos *dumps* das mensagens trocadas entre os vizinhos. Abaixo segue uma descrição de cada coluna da tabela:

- ***Seq.*** Representa o número de seqüência da entrada na tabela, que é diferente para cada entrada de um mesmo *dump* (mas que pode se repetir em *dumps* diferentes).
- ***Time.*** Uma marcação de tempo única para todas as entradas que compartilham um mesmo *dump*. Na prática, reflete a hora/data da captura das informações, não a hora/data em que a mensagem foi enviada.
- ***Type.*** Indica o tipo da mensagem, que pode assumir os valores 'A' (para uma mensagem fazendo um anúncio, ou *announcement*) e 'W' (para uma mensagem de remoção de rota, ou *withdrawal*). Mensagens do tipo 'W' só se utilizam das colunas até *Prefi xLen*, enquanto que as do tipo 'A' se utilizam de todas as colunas da tabela.
- ***PeerAddr.*** É o endereço IP do vizinho de onde as informações sobre o caminho anunciado foram aprendidas.
- ***PeerAS.*** É o número de AS do vizinho de onde as informações sobre o caminho anunciado foram aprendidas.
- ***PrefixAddr.*** É o bloco de endereços IP recebido no anúncio. O tamanho do prefixo é especificado na coluna *Prefi xLen*. Todos os bits além do tamanho do prefixo são zerados automaticamente.
- ***PrefixLen.*** O tamanho, em bits, do prefixo do bloco de endereços IP anunciado.
- ***ASPath.*** É a seqüência de ASs no caminho anunciado. Não é aplicada nenhuma semântica especial nessa coluna e todo o caminho é composto por uma seqüência

de números inteiros separados por espaços em branco (ou seja, é responsabilidade da aplicação que utiliza esses dados fazer a separação desses campos, quando necessário).

- ***Origin.*** É a origem da rota, que pode ser '?' para um caminho de origem incompleta (p. ex., quando uma rota é redistribuida dentro do BGP), 'I' quando a origem é um IGP e 'E' quando a origem é um EGP (BGP propriamente dito).
- ***NextHop.*** Contém o endereço IP do roteador de borda que deve ser utilizado para alcançar o prefixo anunciado. Em alguns casos muito especiais, o roteador vizinho que fez o anúncio pode não ser o *next hop* para o prefixo.
- ***LocalPref.*** Indica a preferência que o vizinho que originou a mensagem BGP tem por essa rota anunciada.
- ***MED.*** É a métrica utilizada para discriminar múltiplos pontos de saída para um AS adjacente.
- ***Community.*** Contém informações sobre as comunidades em uso. Nem sempre está presente, já que as comunidades não são implementadas por todos os roteadores na malha.

3.2.2. Tabela *Tables*

A tabela *Tables* armazena as informações coletadas nos *dumps* das tabelas de roteamento da estação de monitoração. Algumas colunas dessa tabela possuem o mesmo significado da tabela *Messages* e as suas descrições não serão repetidas. São elas: *Seq*, *Time*, *Prefi xAddr*, *Prefi xLen*, *ASPath*, *Origin*, *LocalPref*, *MED* e *Community*. As colunas *PeerAddr* e *PeerAS* não existem nesta tabela. As demais colunas são descritas abaixo:

- ***FromAddr.*** É o endereço do vizinho que fez o anúncio.
- ***FromAS.*** É o número do AS do vizinho que fez o anúncio.
- ***Agg.*** Define quando um anúncio recebido foi agregado por algum outro roteador. Pode assumir dois valores: 'Y' quando um anúncio é resultante de um bloco que sofreu agregação e 'N' caso contrário. Caso o valor seja 'Y', as duas colunas seguintes (*AggAddr* e *AggAS*) são preenchidas.
- ***AggAddr.*** Endereço do roteador que realizou a agregação dos prefixos.
- ***AggAS.*** Número do AS do roteador que realizou a agregação dos prefixos.

4. Medições

As próximas seções apresentam uma série de medições que foram realizadas com base em dados coletados de roteadores do núcleo da Internet. Todos os dados foram obtidos do projeto *Route Views* e injetados artificialmente nas bases de dados da plataforma. Os dados utilizados vêm de outras fontes, e não da captura feita pela própria plataforma, pois os dados que foram coletados pela plataforma até então não são suficientes para a geração de medições mais complexas.

Foram definidos dois tipos de medições que podem ser realizadas com os dados obtidos: as medições de *estatí sticas*, que trabalham com os dados da tabela de roteamento, e as medições de *instabilidade*, que trabalham com os dados obtidos da troca de mensagens entre os vizinhos. Nas próximas seções é apresentada uma série de medições de estatísticas que foram realizadas com base nos dados coletados no núcleo da Internet até outubro de 2002[1]. Para cada medição realizada é feita uma breve análise dos valores

[1]Alguns dos gr´aficos que est˜ao nas pr´oximas seç ˜oes apresentam algumas falhas nas mediç ˜oes que podem ser notadas pelas grandes variaç ˜oes que causam alguns picos e vales. Em alguns casos essas variaç ˜oes s˜ao naturais, e isso ´e indicado no texto que comenta o gr´afico correspondente.

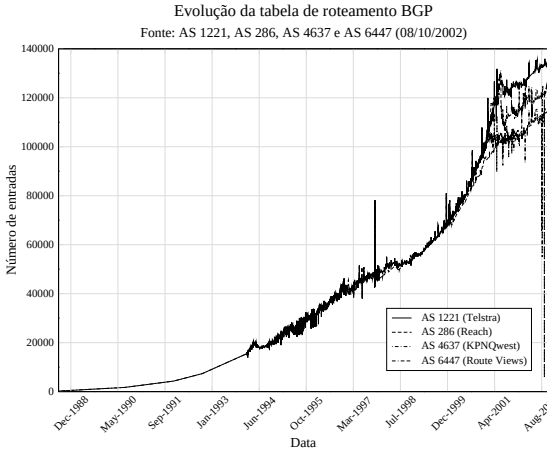

Figura 5: Evolução da tabela de roteamento

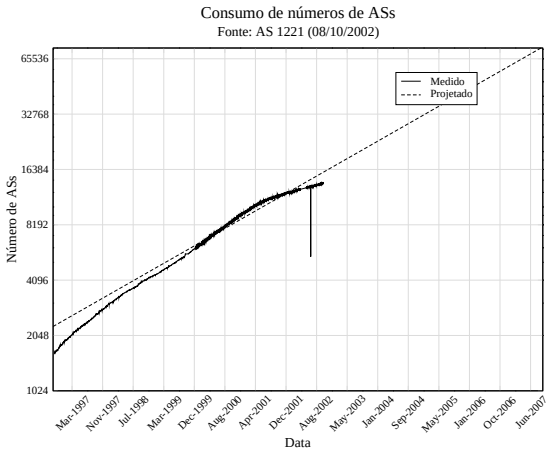

Figura 6: Alocação dos números de AS

encontrados. As medições de instabilidade não são apresentadas neste artigo, mas elas seguem a mesma taxonomia definida por Labovitz et al. em [11].

4.1. Evolução da tabela de roteamento

A Figura 5 apresenta a evolução da tabela de roteamento nos últimos anos. Até meados de 1994, quando a Internet ainda utilizava o conceito de redes *classfull* no roteamento, a tabela estava crescendo em uma escala exponencial. Logo após a implementação do CIDR (*Classless Inter-Domain Routing*), pode-se notar que o crescimento ficou linear, com uma taxa de crescimento anual de aproximadamente 20.000 novas entradas. Se o CIDR não tivesse sido implementado, os roteadores de núcleo simplesmente não iriam aguentar a grande quantidade de entradas na tabela de roteamento e a Internet teria entrado em colapso.

Também é importante notar que as medições foram feitas com dados capturados em 4 ASs diferentes, já que o número de entradas na tabela de um AS depende da complexidade da sua malha de interconexão com outros ASs. Mas pode-se notar que o maior valor observado ficou bem próximo de 140.000 entradas.

4.2. Alocação dos números de AS

Cada rede *multi-homed* (ou seja, que possui mais de uma saída para o tráfego) na Internet que deseja expressar alguma política externa de roteamento distinta deve utilizar um número de AS que é associado aos prefixos anunciados com essa política. Em geral, cada rede possui um único número de AS, e o número de ASs na tabela de roteamento *default-free* aponta, consequentemente, para o número de entidades que possuem políticas de roteamento únicas e que fazem parte do núcleo da Internet. A tendência de alocação dos números de AS durante os quatro últimos anos é exponencial (ver Figura 6). O crescimento do número de ASs pode ser correlacionado com o crescimento na quantidade do espaço de endereços coberto pela tabela BGP. Ao final do ano 2000, a taxa de consumo anual de endereços estava em 7%, enquanto que a taxa de consumo anual dos números de AS estava em 51%. Os ASs estão anunciando cada vez mais endereços menos agregados. Isso aponta para níveis de detalhes de roteamento cada vez mais finos, que são anunciados na tabela de roteamento global, uma tendência que gera uma certa preocupação.

Se a taxa de crescimento continuar com tendência atual, os números de AS (que são de 16 bits) irão se esgotar em meados de 2006. Uma solução para este problema já está sendo estudada pelo IETF e visa modificar o protocolo BGP para permitir que ele carregue números de AS de 32 bits [13]. Enquanto as modificações do protocolo

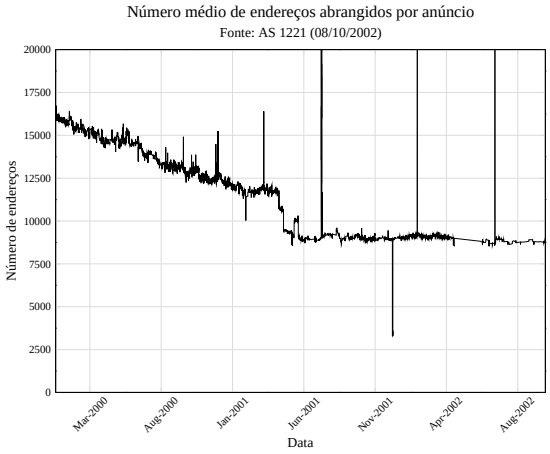

Figura 7: Número médio de endereços abrangidos por anúncio

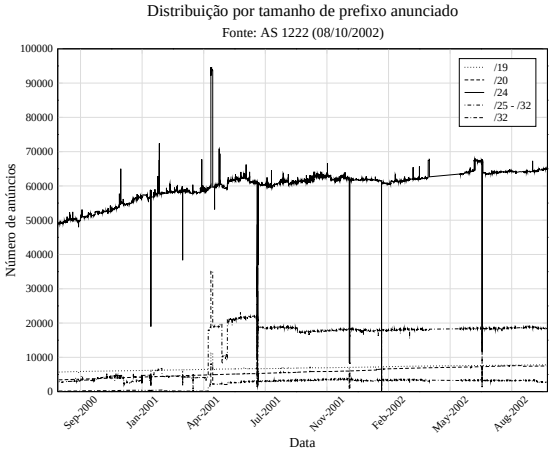

Figura 8: Distribuição por tamanho de prefixo anunciado

são relativamente simples, a maior preocupação está na transição gradual para esse novo protocolo por parte da comunidade da Internet (o que demanda atualização de software, troca de equipamentos, etc.).

4.3. Tamanho médio dos prefixos anunciados

O objetivo da agregação promovida pelo CIDR era permitir anúncios de grandes blocos de endereços agregados na tabela BGP. Para verificar se isso ainda é verdade, os dados capturados sobre os agregados durante os últimos 36 meses foram analisados. Esses dados indicam um declínio na média de endereços por anúncio de prefixo de 16.000 endereços individuais, em novembro de 1999, para 8.800, em outubro de 2002 (ver Figura 7). Isso corresponde a um aumento médio no tamanho dos prefixos de /18,03 para /18,89. Essa tendência é motivo para preocupação, já que ela implica na distribuição do tráfego sobre um número cada vez maior de entradas nas tabelas de roteamento.

Um cenário em potencial é o de que o tamanho dos anúncios continue a crescer. Com o uso cada vez maior de mecanismos de tradução de endereços, como o NAT, e a constante preocupação com a escassez de endereços do protocolo IPv4, esse cenário é bem possível. Projeções para o tamanho médio dos prefixos, usando-se as tendências atuais do número de entradas na tabela BGP e do número total de endereços abrangidos pela tabela BGP, apontam um crescimento no tamanho médio do prefixo anunciado de 1 bit a cada 29 meses. Isso causa implicações diretas nos algoritmos de busca de rotas utilizados no projeto de roteadores.

4.4. Distribuição do tamanho dos prefixos

Um grande esforço foi feito em meados dos anos 90 para evitar o uso extensivo do espaço de endereçamento da antiga Classe C do IPv4 e para encorajar os grandes provedores a realizarem o anúncio de blocos maiores de endereços. Esse esforço foi reforçado pelas autoridades responsáveis pelo registro de endereços, que obrigavam a alocação mínima de blocos de prefixo /19 e, mais recentemente, /20. Essas medidas foram introduzidas quando haviam entre 20.000 e 30.000 entradas na tabela BGP global. É interessante notar que aproximadamente cinco anos depois, 53.000 das 96.000 entradas da tabela BGP eram de blocos /24 (ver Figura 8). Em termos absolutos, o prefixo /24 é o que mais cresce na tabela BGP.

As entradas desses blocos de endereços menores na tabela BGP também apresentam um nível de variação muito maior em escala de horas. Mesmo que um grande

número de pontos de roteamento BGP estejam utilizando *route flap dampening*[2], ainda existe uma grande quantidade oscilação dessas entradas nessa área em particular da tabela BGP (quando as mensagens do protocolo BGP são analisadas de acordo com cada tamanho de prefixo). Dado que o número desses pequenos prefixos está crescendo rapidamente, existe a preocupação com o nível total de fluxo de mensagens BGP, com um número também crescente de anúncios e retiradas de rotas por segundo, apesar do uso de *flap dampening*. Essa preocupação fica pior com a observação de que, em termos de estabilidade da malha BGP sob pressão de escalabilidade, não é o tamanho absoluto da tabela BGP que é o mais importante, mas sim a taxa das computações dinâmicas de caminhos que ocorrem quando esses anúncios e retiradas são recebidos pelos roteadores. As retiradas de prefixos são a maior preocupação, dado o número de estados transientes intermediários que o algoritmo de vetor-distância do BGP explora quando processa uma retirada de prefixo. Observações experimentais recentes indicam um tempo de convergência típico de aproximadamente dois minutos para a propagação de uma retirada de prefixo por todo o domínio BGP [12]. Um aumento na densidade da malha BGP, em conjunto com o aumento da taxa dessas mudanças dinâmicas, gera sérias implicações na manutenção da estabilidade geral do sistema BGP, à medida que ele continuar crescendo.

As políticas de alocação de blocos de endereços também tiveram algum impacto na distribuição dos prefixos na tabela de roteamento. A prática inicial era alocar blocos de tamanho mínimo /19, e as 10.000 entradas de prefixos entre os tamanhos /17 e /19 são uma conseqüência dessa política. Recentemente a política foi modificada, e o tamanho mínimo de bloco que pode ser alocado agora é /20, que totalizam aproximadamente 4.000 das entradas na tabela BGP, e, em termos relativos, é um dos prefixos que está crescendo mais rapidamente.

O número de entradas correspondentes a blocos de endereços bem pequenos (menores que /24), enquanto que em menor número em proporção ao resto da tabela BGP, são as que crescem mais rapidamente em termos relativos. O número de prefixos entre /25 e /32 na tabela de roteamento está crescendo mais rapidamente, em termos de variações percentuais, do que qualquer outra área de tabela de roteamento. Se filtros por tamanho de prefixo fossem amplamente utilizados atualmente, a prática de realizar anúncios de blocos pequenos em políticas de roteamento distintas não teria nenhum benefício em particular, já que o anúncio seria filtrado mais cedo ou mais tarde na malha BGP global. O crescimento do número desses pequenos blocos de endereços e a diversidade de *AS paths* associados com seus anúncios, apontam para um uso relativamente limitado de mecanismos de filtro por tamanho de prefixo na Internet atualmente. Na ausência de nenhuma pressão corretiva (na forma da adoção de regras de filtragem) o crescimento dos anúncios de blocos pequenos deverá se manter.

4.5. Agregações e *holes*

Com a estrutura de roteamento CIDR é possível realizar o anúncio de prefixos mais específicos de agregados existentes. O objetivo desse anúncio mais específico é fazer um "furo" na política do anúncio do agregado maior, criando uma política diferente para o prefixo de endereço mais específico. Esse mecanismo pode ser utilizado não somente para anunciar uma política de conectividade diferente, mas também pode ser utilizado como uma espécie de mecanismo rudimentar para realizar balanceamento de carga e *backup* mútuo para redes *multi-homed*. Nesse modelo, uma rede pode realizar o mesmo anúncio de agregado para cada conexão, mas fazendo um conjunto de anúncios mais es-

[2]O mecanismo de *route flap dampening* funciona como uma puniç ˜ao para vizinhos que n˜ao se comportam bem na malha, deixando-os de "castigo"por um certo tempo, evitando assim que eles consigam provocar uma sobrecarga na CPU dos demais roteadores da malha.

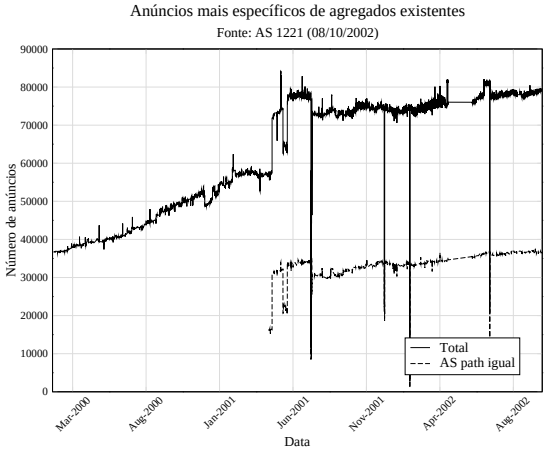

Figura 9: Anúncios mais específicos de agregados existentes

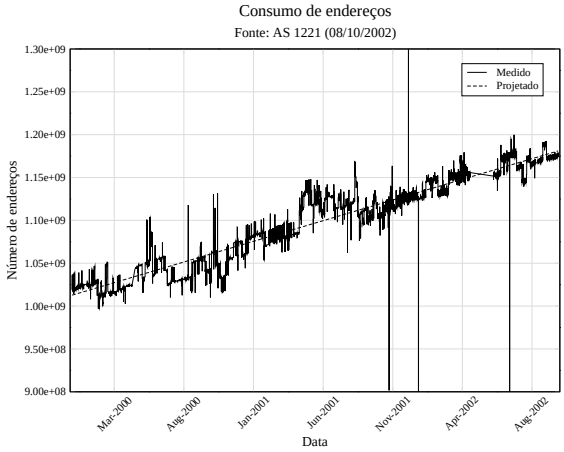

Figura 10: Consumo de endereços IPv4

pecíficos em outras conexões, alterando cada anúncio específico de forma que a carga em cada conexão seja aproximadamente balanceada (o balancemento depende totalmente do conhecimento empírico do operador de cada rede). As duas formas de "furos" podem ser facilmente distinguidas em uma tabela de roteamento: enquanto que a abordagem de diferenciação de políticas usa um *AS path* que é diferente do anúncio do agregado, as configurações de balanceamento de carga e *backup* mútuo usam o mesmo *AS path* para ambos agregado e anúncios mais específicos.

É difícil entender se o uso desses anúncios mais específicos teve a intenção de ser uma exceção a uma regra mais geral ou não, fugindo do objetivo original do CIDR, mas parece haver o uso em grande escala desse mecanismo na tabela de roteamento. Aproximadamente 80.000 anúncios, ou 60% de toda a tabela de roteamento, estão sendo utilizados para criar furos nos anúncios existentes de agregados (ver Figura 9). De todos esses, uma maioria de aproximadamente 43.000 rotas usam *AS paths* distintos, o que mostra, novamente, a conseqüência de níveis mais finos de ajuste das políticas de roteamento na malha de interconexão.

Mesmo que os dados relativos a esses tipos de anúncios estejam sendo coletados a relativamente pouco tempo (em relação ao tempo de coleta da tabela de roteamento completa), o nível de crescimento é um forte indicativo de que a diferenciação de políticas em níveis mais finos dentro dos agregados dos provedores é o que está causando o crescimento geral da tabela de roteamento.

4.6. Consumo de endereç os

Através das medições realizadas foi possível rastrear a quantidade total de endereços que são cobertos pelos anúncios da tabela de roteamento BGP. No período entre novembro de 1999 e outubro de 2002, o uso de endereços cresceu de 1,02 bilhões para 1,175 bilhões (ver Figura 10). Existem alguns prefixos /8 que são periodicamente anúnciados e retirados da tabela BGP e, se o efeito causado por esses prefixos for removido, pode-se notar que o crescimento tem sido praticamente constante durante os últimos anos, como pode ser observado na projeção de endereços na Figura 10. Segundo a projeção, o valor final da quantidade de endereços cobertos pelos anúncios da tabela BGP está em aproximadamente 1,181 bilhões de endereços em outubro de 2002.

A taxa de crescimento anual fica um pouco abaixo de 7% e, com essa taxa, espera-se que o espaço de endereçamento IPv4 ainda seja suficiente para aproximadamente 19

anos[3]. Comparado com os 100% de crescimento no número de anúncio de rotas no mesmo período, parece que a maior parte do crescimento da Internet, em termos de crescimento em número de dispositivos conectados, está ocorrendo atrás de diversas formas de mecanismos de tradução de endereços. Para tentar contornar a natureza finita do espaço de endereçamento disponível, a Internet parece ter adotado até então a abordagem dos NATs, apesar de suas diversas desvantagens [14]. Isso também suporta o aumento já observado dos pequenos fragmentos de endereços com políticas distintas anunciados na tabela BGP, já que esses pequenos blocos abrangem redes arbitrariamente grandes localizadas atrás de um ou mais desses dispositivos.

5. Conclusões e trabalhos futuros

Este artigo apresentou uma plataforma para a monitoração de malhas BGP que tem a finalidade de auxiliar na administração das redes que fazem parte do núcleo da Internet. Os administradores dessas redes realizam a monitoração da malha BGP através da linha de comando no console dos roteadores de borda de seus ASs, o que é uma tarefa relativamente complexa, que consome bastante tempo e demanda uma equipe capacitada. Foi mostrado como a metodologia de captura de informações dessa plataforma consegue trazer essas informações até uma base de dados de fácil consulta (SQL), permitindo que um histórico do estado da malha seja mantido para análises futuras, que podem auxiliar na engenharia de tráfego e na detecção de problemas. Esse tipo de ferramenta é muito importante e desejável na gerência de uma malha BGP, já que existem muitos parâmetros para serem monitorados. Por exemplo, através da base de dados é possível extrair informações como *"que AS está anunciando a rede 200.132.0.0/16 para o meu AS?"* ou *"quais foram as variações no atributo MED no anúncio do bloco 200.132.0.0/16 nos últimos 2 dias?"*. Esse tipo de informação é muito valiosa para a monitoração de uma malha de interconexão complexa, principalmente em grandes provedores de *backbone* e/ou trânsito.

Uma série de *scripts* trabalham sobre os dados coletados, gerando diversos tipos de relatórios e estatísticas. Algumas dessas estatísticas foram apresentadas na Seção 4. Esses *scripts* podem ser criados pelos administradores à medida que novas estatísticas e relatórios são necessários. O acesso à base dados é feito de uma forma bastante direta, através da API do banco de dados fornecida pela linguagem de programação adotada. Os resultados desses *scripts* ficam disponíveis através de um servidor web.

Como trabalho futuro pode ser deixada a tarefa de sincronização dos dados capturados por diversas estações de monitoração distribuídas. Atualmente cada estação é independente e auto-suficiente. Talvez uma estrutura hierárquica, com várias estações de monitoração em diversos ASs diferentes, onde apenas uma das estações possua os *scripts* e a interface web, e as demais fiquem somente com a tarefa de captura das informações. A correlação de informações de roteamento aprendidas em vários ASs pode ser uma ferramenta bastante útil para a detecção de problemas na malha global de roteamento, além de auxiliar na monitoração da abrangência e atraso dos anúncios.

Referências

[1] REKHTER, Y.; LI, T. **A Border Gateway Protocol 4 (BGP-4).** [S.l.]: Internet Engineering Task Force, 1995. RFC 1771.

[2] VARADHAN, K.; GOVINDAN, R.; ESTRIN, D. **Persistent route oscillations in inter-domain routing.** [S.l.]: USC/ISI, 1996. (96-631).

[3]Esta estimativa foi feita sem desconsiderar endereç os reservados como, por exemplo, as remanescentes classes D e E e os endereç os utilizados em redes privadas.

[3] GRIFFIN, T. G.; WILFONG, G. An Analysis of BGP Convergence Properties. In: SIGCOMM, 1999. **Proceedings...** [S.l.: s.n.], 1999, p.277–288.

[4] WILLIS, S.; BURRUSS, J.; CHU, J. **Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIv2.** [S.l.]: Internet Engineering Task Force, 1994. RFC 1657.

[5] Telstra. **Página com estatísticas BGP sobre o AS1221 (Telstra).** Disponível em: http://bgp.potaroo.net/. Acesso em: setembro de 2002.

[6] Route Views. **Página do projeto Route Views da Universidade de Oregon.** Disponível em http://www.routeviews.org/. Acesso em: setembro de 2002.

[7] Zebra. **Página do software Zebra.** Disponível em http://www.zebra.org/. Acesso em: dezembro de 2002.

[8] RRDtool. **Página do software RRDtool.** Disponível em http://www.rrdtool.org/. Acesso em: setembro de 2002.

[9] GNU GPLv2. **Licenç a GNU GPL versão 2.** Disponível em http://www.gnu.org/licenses/gpl.html. Acesso em: dezembro de 2002.

[10] MySQL. **Página do software MySQL.** Disponível em http://www.mysql.org. Acesso em: dezembro de 2002.

[11] LABOVITZ, C.; MALAN, G. R.; JAHANIAN, F. Origins of Internet routing instability. In: INFOCOM'99, 1999, New York, NY. **Proceedings...** Institute of Electrical and Electronic Engineers, 1999, v.1, p.218–226.

[12] LABOVITZ, C. et al. The Impact of Internet Policy and Topology on Delayed Routing Convergence, In: INFOCOM'01, 2001, Anchorage, AK. **Proceedings...** Institute of Electrical and Electronic Engineers, 2001, v.1, p.537–546.

[13] VOHRA, Q.; CHEN, E. **BGP support for four-octet AS number space.** [S.l.]: Internet Engineering Task Force, 2002. Internet-Draft (draft-ietf-idr-as4bytes-05.txt).

[14] HAIN, T. **Architectural Implications of NAT.** [S.l.]: Internet Engineering Task Force, 2000. RFC 2993.