

Entendendo a Compressão de Dados Avançada

Visão Geral Quase todos os appliances de otimização de WAN se aproveitam de rotinas avançadas para melhorar o desempenho de aplicação. Em alto nível, todos eles armazenam e se aproveitam dos dados de rede transferidos anteriormente para atingir altos níveis de compressão. No entanto, quando são mais bem examinados, há grande variação no modo como eles atingem esses ganhos de compressão e nas limitações resultantes.

Pacotes X Sessões

A maioria dos sistemas de compressão de rede vigentes é baseada em pacotes. Sistemas de compressão baseados em pacotes armazenam em buffer pacotes destinados para uma rede remota com um descompressor. Esses pacotes são então comprimidos, um de cada vez ou em grupos, e enviados para o descompressor, pelo qual o processo é revertido. A compressão por pacotes está disponível há muitos anos e pode ser encontrada em roteadores, clientes VPN e nos appliances WX e WXC da Juniper Networks®.

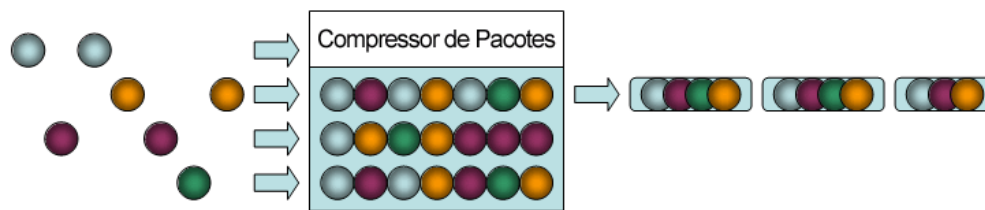


Figura 1: Compressão por pacotes

O principal problema da compressão por pacotes é que ela reúne diferentes tipos de dados no momento da compressão. Ao operar com dados homogêneos, qualquer rotina de compressão atinge níveis maiores de compactação. No caso de dados heterogêneos, como um conjunto de pacotes de vários protocolos diferentes, as taxas de compressão caem drasticamente.

Sistemas de compressão por pacotes também apresentam outros problemas. Ao comprimir pacotes, esses sistemas devem escolher entre gravar pacotes pequenos para a rede e realizar o trabalho adicional de agregar e encapsular vários pacotes. Nenhuma das opções produz os melhores resultados. Gravar pacotes menores na rede aumenta a carga de cabeçalhos de TCP/IP. Agregar e encapsular pacotes também acrescenta cabeçalhos (de encapsulamento) ao fluxo de dados.

Diferentemente das soluções de compressão anteriores, o appliance F5 WANJet opera na camada de sessões. Isso permite a ele aplicar a compressão em um conjunto de dados completamente homogêneo e, ao mesmo tempo, lidar com todos os tipos de aplicativos. O resultado são taxas de compressão maiores do que nos sistemas por pacotes.

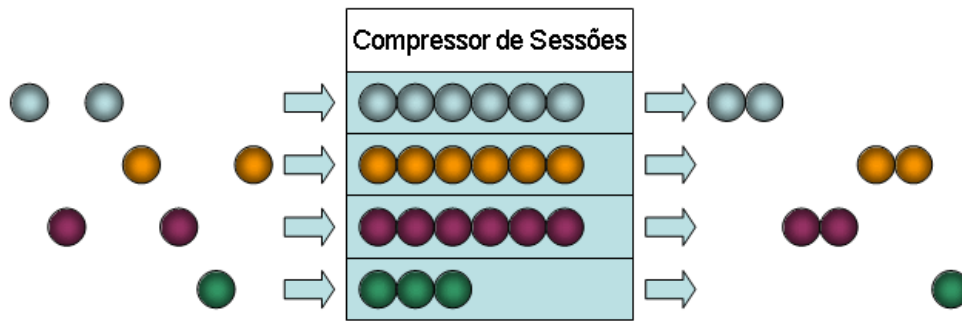


Figura 2: Compressão por sessões

Além disso, quando se opera na camada de sessões, os problemas de fronteiras de pacotes e de reempacotamento são eliminados. Isso permite que o dispositivo WANJet encontre facilmente repetições em fluxos de dados que podem estar a muitos bytes de distância entre si na camada 3, mas que são contíguos na camada 5. A taxa de transferência do sistema também é aumentada quando a compressão é aplicada na camada de sessões, em virtude da eliminação da etapa de encapsulamento.

Tamanho de Dicionário

Uma limitação que todas as rotinas de compressão têm em comum é o espaço de armazenamento limitado. Algumas rotinas, como a Gzio, armazenam apenas 64kb de dados. Outras técnicas, como sistemas de compressão baseados em discos, podem armazenar até um terabyte de dados. Para entender o efeito do tamanho de dicionário, é necessário um entendimento básico do gerenciamento de cache.

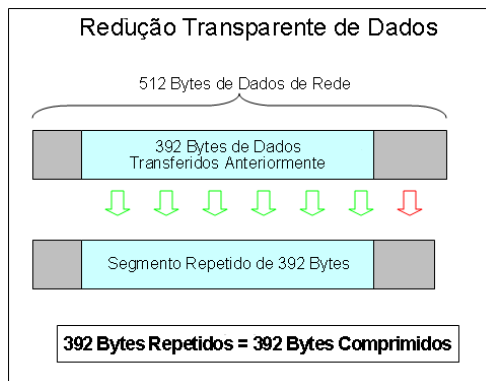
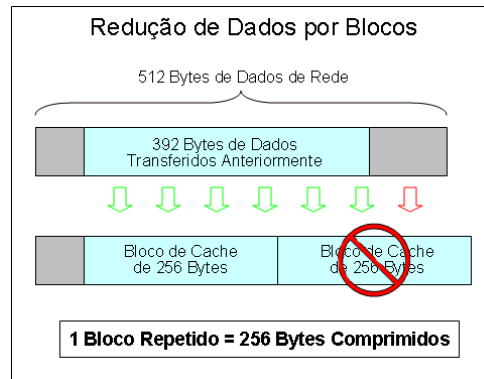
De forma similar às requisições para um website, nem todos os bytes transferidos na rede se repetem com a mesma frequência. Alguns padrões de bytes ocorrem com grande frequência por serem parte de um documento popular ou de um protocolo de rede comum. Outros padrões de bytes ocorrem uma vez e nunca mais se repetem. O relacionamento entre seqüências de bytes usualmente repetidas e seqüências de menor frequência são vista nas leis de Zipf e de Heaps.

Todos os sistemas de compressão modernos baseados em dicionários se aproveitam dessa distribuição desigual, armazenando os dados de acesso mais freqüente e descartando os de menor frequência. Por meio desse tipo de otimização, um dicionário que armazene menos de 10% de todos os padrões de bytes pode atingir uma taxa de compressão acima de 50%. O efeito dessa distribuição desigual de padrões de bytes é evidente na eficiência de programas de compressão comuns. O Gzip armazena somente 64kb de histórico e, mesmo assim, atinge uma compressão de aproximadamente de 64%. O Bzip2, que armazena entre 100kb e 900kb de histórico, comprime a 66% em média. A razão pela qual o Gzip e o Bzip2 operam tão bem apesar da falta de uma armazenagem substancial de dados é que as seqüências de bytes de maior frequência representam a maioria dos bytes em uma rede.

Blocos X Bytes

Sistemas baseados em blocos, como os appliances WXC da Juniper Networks e Steelhead® da Riverbed® Technology, armazenam segmentos de dados anteriormente transferidos que trafegam pela WAN. Quando esses blocos são encontrados uma segunda vez, as referências dos blocos são transmitidas para o appliance remoto, que reconstrói os dados originais.

Uma desvantagem crítica dos sistemas baseados em blocos é que os dados repetitivos quase nunca têm o comprimento exato de um ou mais blocos. Como resultado, as repetições são sempre parciais, deixando parte dos dados repetidos sem compressão. O diagrama à direita demonstra o que acontece quando um sistema que usa um tamanho de bloco de 256 bytes tenta comprimir 512 bytes de dados.



De forma similar à abordagem da Riverbed e da Juniper de usar dados transferidos anteriormente para reduzir a utilização de rede, o F5 WANJet monta um dicionário de bytes transferidos anteriormente, usando a tecnologia TDR (Transparent Data Reduction – Redução Transparente de Dados). No entanto, ao contrário dos appliances WXC e Steelhead, o WANJet combina e envia referências com granularidade ao nível de bytes. O diagrama à esquerda ilustra

como o WANJet lida com os mesmos 512 bytes de dados.

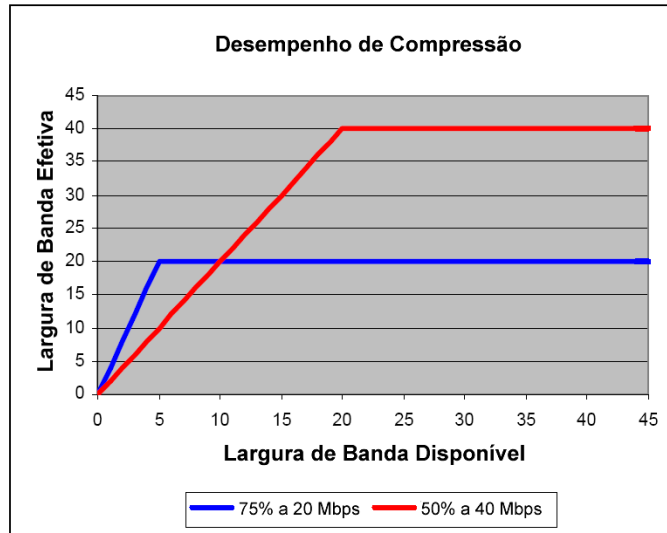
Ao contrário dos sistemas por blocos, todo o padrão repetido é encontrado e comprimido pelo WANJet. Nos exemplos anteriores, ao invés de comprimir apenas 256 bytes de dados, o WANJet consegue encontrar e reduzir todos os 392 bytes de dados repetidos. Esse nível de granularidade permite que o WANJet atinja níveis maiores de compressão do que os sistemas concorrentes baseados em blocos, não apenas em documentos como também em cabeçalhos de protocolos de camada de aplicação.

A Taxa de Transferência Faz a Diferença?

Embora atingir uma alta taxa de compressão seja vital para melhorar o desempenho de aplicativos em redes de largura de banda limitada, a taxa de transferência do sistema também tem um papel importante. Os ganhos de desempenho de uma determinada tecnologia de compressão podem ser avaliados levando em consideração a taxa de compressão esperada das tecnologias, a taxa máxima de transferência de compressão dos dispositivos e a largura de banda da rede. Se a taxa de compressão for muito baixa, a rede permanecerá saturada e os ganhos de desempenho serão mínimos. Da mesma forma, se a velocidade de compressão for baixa, o próprio compressor será o gargalo.



No gráfico abaixo, o compressor com a taxa de compressão um pouco melhor (75%) só supera a rotina mais rápida para velocidades de conexão abaixo de 10 Mbps. A razão para isso é que, em velocidades acima de 5 Mbps, bem abaixo da marca de 20 Mbps, a rotina de compressão mais lenta não consegue processar dados suficientes para continuar a utilizar totalmente a rede.



A Redução Transparente de Dados, da forma implementada no F5 WANJet, foi otimizada para manter uma alta taxa de transferência. Enquanto o Juniper WXC 500 atinge um máximo de 20 Mbps e o Riverbed Steelhead 5010 atinge 45 Mbps, o WANJet pode manter velocidades de até 400 Mbps com um único appliance. Esse nível de desempenho permite que o WANJet ofereça ganhos de desempenhos de 8 vezes em redes T3 – um feito impossível para os appliances WXC e Steelhead.

Resumo

Atingir ganhos substanciais de desempenho de aplicativo através de compressão exige não só um bom algoritmo de compressão como também toda uma arquitetura de sistema projetada para o desempenho. O sistema de compressão deve encontrar padrões repetidos com precisão para atingir altas taxas de compressão. Ele deve gerenciar tanto os dados armazenados quanto o tráfego de aplicativo de entrada para maximizar a eficiência. E deve fazer tudo isso rapidamente para minimizar a latência e continuar a alimentar a rede. O WANJet e a TDR (Redução Transparente de Dados) da F5 Networks foram projetados desde a concepção para atender a essas necessidades.