Uma Análise Experimental de Desempenho dos Protocolos de Armazenamento AoE e iSCSI

Pedro Eugênio Rocha[1], **Leonardo Antônio dos Santos**[2]

[1]Departamento de Informática
Universidade Federal do Paraná (UFPR)
Curitiba – PR

[2]Faculdades Integradas do Brasil (UniBrasil)
Curitiba – PR

pedro@inf.ufpr.br, heiligerstein@gmail.com

Abstract. *This paper presents an experimental performance comparison of the storage protocols AoE and iSCSI. Through the execution of a set of microbenchmarks and macrobenchmarks, we analyze the performance and efficiency of both protocols in terms of achieved throughput, cache memory, network and CPU utilization. Based on these results, we state that the protocol must be carefully chosen considering the workloads to be used, as well as any possible memory or CPU restrictions.*

Resumo. *Este artigo apresenta uma comparação experimental de desempenho dos protocolos de armazenamento de dados remoto AoE e iSCSI. Através da execução de um conjunto extensivo de experimentos utilizando microbenchmarks e macrobenchmarks, analisamos o desempenho e a eficiência de ambos os protocolos em termos de vazão de dados, quantidade de memória cache, utilização de rede e CPU. A partir desses dados, mostramos que a escolha do protocolo com melhor desempenho deve ser realizada com base no workload a ser utilizado, bem como sobre possíveis restrições de memória e processamento.*

1. Introdução

A demanda por *data centers* de grande porte e altamente escaláveis em termos de processamento, comunicação e armazenamento de dados vem crescendo nos últimos anos, principalmente impulsionada pela popularização de serviços de computação em nuvem e virtualização. Particularmente na área de armazenamento de dados, a abordagem mais utilizada atualmente é a criação de redes dedicadas para dados, chamadas de *Storage Area Networks* (SANs), com requisitos diferenciados das LANs em termos de confiabilidade, latência, vazão de dados e, principalmente, custo.

SANs operam na camada de bloco, de forma que toda a lógica do sistema de arquivos seja executada nos servidores de aplicação; os servidores de armazenamento, por sua vez, apenas exportam um grande espaço de armazenamento contíguo e confiável. Para atender à necessidade de comunicação entre estes servidores, alguns protocolos como *Fibre Channel* (FC), iSCSI e AoE são utilizados, que, dentre outros, encapsulam os blocos trafegados na rede.

Contudo, o protocolo FC requer hardware e software especializado e de alto custo para sua utilização, como *switches*, interfaces de rede e até cabeamento específico

[Zhou and Chuang 2009]. Diferentemente, os protocolos iSCSI e AoE podem ser executados sobre a mesma infraestrutura de rede local do *data center* (Ethernet). Além disso, caso haja necessidade de uma rede dedicada para a rede de armazenamento de dados, eliminando a competição da largura de banda disponível entre as redes normal e de armazenamento, uma segunda rede Ethernet pode ser utilizada, com custo muito inferior se comparada a uma rede *Fibre Channel*.

Devido a este fato e de que iSCSI e AoE possuem implementações de código livre disponíveis para utilização, neste artigo apresentamos uma comparação destes protocolos de armazenamento de dados em rede. Nosso trabalho difere de trabalhos anteriores nos seguintes aspectos: (a) em trabalhos anteriores a análise é limitada à execução de microbenchmarks, que testam apenas partes muito específicas do protocolo, não considerando os workloads presentes em sistemas reais, (b) analisamos o impacto do uso de caches e da quantidade de memória disponível ao sistema, (c) apresentamos uma comparação da utilização de rede e (d) analisamos a eficiência de utilização de CPU de cada protocolo.

A análise apresentada neste trabalho é baseada estritamente nos resultados obtidos através de um conjunto extensivo de experimentos. Inicialmente, apresentamos o resultado de microbenchmarks sobre o ambiente de testes criado. Em seguida, uma sequência de macrobenchmarks é executada, simulando workloads encontrados em sistemas reais como servidores Web, de e-mails, de arquivos e de banco de dados transacionais. A partir desses experimentos, apresentamos uma análise dos resultados encontrados sobre o ponto de vista da vazão de dados, utilização de rede e de CPU.

Com base nos resultados obtidos, fizemos as seguintes observações: (a) o protocolo iSCSI apresenta melhor vazão em workloads de escrita predominante, cerca de 9%, e desempenho muito próximo ao AoE em workloads de leitura, apenas 1%, mesmo considerando o *overhead* causado pelas camadas de rede adicionais, caso haja memória suficiente para cache, (b) o AoE é a melhor opção para workloads que não se beneficiam do mecanismo de cache do sistema operacional, como bancos de dados transacionais, e (c) o protocolo iSCSI é, em média, 19% menos eficiente em termos de CPU e 3% em utilização de rede sob qualquer workload.

O restante deste artigo está organizado da seguinte forma. A Seção 2 contém uma breve descrição dos protocolos AoE e iSCSI, bem como uma análise de suas principais diferenças. A Seção 3 resume os trabalhos relacionados. Na Seção 4 o ambiente de testes é descrito, bem como a metodologia utilizada para a realização dos experimentos. Os resultados dos microbenchmarks e macrobenchmarks são apresentados nas Seções 5 e 6, respectivamente, enquanto a Seção 7 contém uma discussão sobre os resultados encontrados. Finalmente, a Seção 8 conclui este trabalho.

2. Protocolos de Armazenamento em Rede

Nesta Seção, apresentamos as principais características dos protocolos de armazenamento de dados em SANs de baixo custo: AoE (*ATA Over Ethernet*) e iSCSI (*Internet Small Computer System Interface*). Consideramos estes protocolos como sendo de baixo custo pois implementações estáveis e de livre uso são disponibilizadas. Além disso, como ambos os protocolos são encapsulados em quadros Ethernet, não há a necessidade de hardware especializado e custoso, como em redes *Fibre Channel*, por exemplo. O restante da Seção está organizado da seguinte forma: os protocolos AoE e iSCSI são detalha-

dos nas Subseções 2.1 e 2.2, respectivamente, enquanto a Subseção 2.3 discorre sobre as principais semelhanças e diferenças entres estes protocolos.

2.1. AoE

O AoE é um protocolo de padrão aberto que permite o acesso remoto a dispositivos de bloco através de uma rede Ethernet. O protocolo AoE é extremamente simples, causando baixo *overhead* em termos de processamento e rede, pois opera diretamente na camada de enlace. Além disso, possui baixo custo quando comparado a outros protocolos, como FC, que exige o uso de interfaces de rede e *switches* especializados.

O AoE encapsula comandos ATA em quadros Ethernet. O ATA (*Advanced Technology Attachment*) é um conjunto padronizado de comandos para acesso a dispositivos de armazenamento em bloco. O chamado *AoE initiator* (cliente) encapsula comandos ATA em quadros *Ethernet* e repassa ao *AoE target* — máquina que exporta o dispositivo de disco. Após lidos, os blocos de disco solicitados são encapsulados pelo *target* em um quadro Ethernet que, por fim, é enviado novamente ao *initiator*. A Figura 1(a) mostra a organização das camadas do protocolo AoE.

2.2. iSCSI

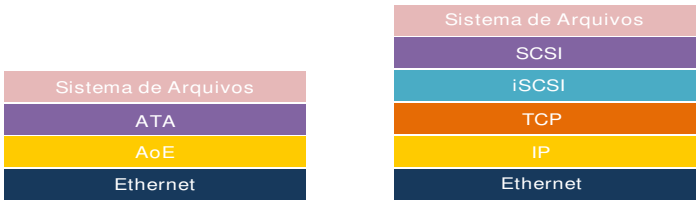
O Internet SCSI, ou iSCSI, é um dos mais conhecidos protocolos para armazenamento remoto em SANs. Semelhante ao AoE, o iSCSI encapsula comandos de dispositivos de armazenamento, mas, no seu caso, através do conjunto de comandos SCSI. Apesar disso, diferentemente do protocolo AoE, o iSCSI opera na camada de aplicação do modelo TCP/IP. Por este motivo, o iSCSI pode ser implantado utilizando-se infraestruturas de rede já existentes. A organização das camadas do protocolo iSCSI é mostrada na Figura 1(b).

SCSI é uma popular família de protocolos que torna possível a comunicação entre o sistema operacional e dispositivos de I/O, operando especialmente dispositivos de armazenamento. O SCSI consiste em protocolos de aplicação *request/response* com um modelo de arquitetura comum, bem como um conjunto de comandos padronizados para diferentes classes de dispositivos, como disco rígidos, discos sólidos e unidades de fita [Satran et al. 2004].

2.3. Comparação entre iSCSI e AoE

A principal diferença entre os protocolos AoE e iSCSI é mostrada na Figura 1. O AoE foi construído para ser simples, eliminando as camadas de rede, transporte e em alguns casos, a criptografia dos dados. Já o iSCSI é um protocolo de camada de aplicação que trabalha sobre a pilha de protocolos TCP/IP. Por operar sobre o protocolo TCP, existe o *overhead* referente às confirmações de entrega de segmentos, ordenação, controle de congestionamento e de fluxo, entre outros. O AoE, por operar sobre a camada de enlace, não trabalha com protocolos de rede com entrega garantida, assim, pode funcionar de forma assíncrona e é capaz tanto de enviar quanto de receber várias solicitações de uma vez e sem respeitar a ordem, ficando o trabalho de verificação dos dados para a camada de aplicação [Zhou and Chuang 2009].

Comparativamente, tanto o padrão ATA quanto o SCSI são padrões de comandos para conectividade com dispositivos de armazenamento em geral. Tradicionalmente, o ATA era considerado mais barato e simples e o SCSI mais caro e robusto, mas esta



(a) AoE (b) iSCSI

Figura 1. Comparação entre os protocolos AoE e iSCSI.

comparação já não é mais verdadeira. Ambos utilizam DMA (*direct memory access*), que elimina o problema de interrupções à CPU durante o processo de leitura ou escrita; ambos podem fazer enfileiramento de comandos fora de ordem para a CPU; e ambos possuem recurso de *hotswap*, que permite o dispositivo ser removido e conectado sem problemas com o sistema em execução [LoBue et al. 2002, Aycock 2006].

O ATA e o SCSI foram criados para arquiteturas de transferência de dados originalmente paralela, mas atualmente já se utilizam estes padrões em arquiteturas serial com o SAS (Serial Attached SCSI) e SATA (Serial ATA). Quanto à velocidade, é difícil dizer qual padrão é mais rápido do que o outro, pois isto seria como comparar processadores diferentes para julgar o desempenho de um sistema. A velocidade não depende apenas do conector, mas da quantidade de giros do disco, o quão rápido a cabeça de leitura se move entre outros fatores [LoBue et al. 2002, Aycock 2006].

3. Trabalhos Relacionados

A popularização da virtualização e sua aplicação na consolidação de ambientes para *Cloud Computing* aumentou a demanda por protocolos eficientes de armazenamento de dados em rede. Apesar de existirem diferentes alternativas, como FC, iSCSI e AoE, o protocolo a ser utilizado depende fortemente de requisitos como desempenho, confiabilidade, latência e, principalmente, custo.

Grande parte das avaliações de desempenho de protocolos de armazenamento em SANs presentes na literatura restringem-se à análise individual de um protocolo [Tan et al. 2005, Aiken et al. 2003, Zhou and Chuang 2009]. Embora existam comparações de desempenho entre diferentes protocolos, muitos dos trabalhos comparam *Fibre Channel*, por ser amplamente utilizado em sistemas organizacionais de grande porte, com protocolos alternativos [Voruganti and Sarkar 2001, Follett 2001]. Contudo, o protocolo Fibre Channel exige a utilização de hardware especializado e de alto custo, fugindo do escopo deste trabalho.

Uma análise preliminar do desempenho dos protocolos AoE e iSCSI é mostrada em [Chuang and Wenbi 2009]. Neste artigo, microbenchmarks de escrita são executados sobre ambos os protocolos, variando o MTU dos quadros Ethernet. Além disso, a utilização de processamento dos protocolos é comparada. Em [Gerdelan et al. 2007], é apresentada mais uma comparação entre os protocolos iSCSI e AoE, considerados *eficientes em termos de custo* por não empregarem hardware especializado. Gerdelan *et al.* também analisam o desempenho dos protocolos somente através da comparação dos resultados de diferentes microbenchmarks em ambas as arquiteturas.

Argumentamos que, apesar de fornecer uma ideia preliminar do desempenho, mi-

crobenchmarks não refletem o comportamento dos protocolos em cenários reais de uso. Uma medição mais precisa deve considerar os diferentes tipos de workloads que podem existir em sistemas reais e de grande porte, que podem ser simulados com a utilização de macrobenchmarks.

4. Ambiente de Testes e Metodologia

Com o objetivo de testar o desempenho dos protocolos de armazenamento de rede, montamos um ambiente de testes contendo dois servidores. O primeiro servidor, chamado de *target*, contém dois processadores Xeon X5690 *six-core* 3.47 GHz, 64 GB de memória RAM e discos 10.000 rpm SCSI com capacidade de 300 GB. Um disco é utilizado exclusivamente para os testes com os protocolos. A segunda máquina, ou *initiator*, contém dois processadores Xeon *dual-core* 3 GHz, 8 GB de memória RAM e discos SCSI de 146 GB. As duas máquinas possuem mais de uma interface de rede, sendo diretamente interligadas através de interfaces Ethernet 1 Gigabit, dedicadas para os testes. Finalmente, ambas utilizam o sistema operacional Debian 6.0.

Primeiramente, executamos um conjunto de microbenchmarks sobre o ambiente de testes criado. Os microbenchmarks têm o objetivo de estabelecer uma linha de base sobre o desempenho esperado dos protocolos em situações muito específicas de uso. Os resultados obtidos através dos microbenchmarks, embora não reflitam situações próximas das reais de uso, são úteis na interpretação de resultados mais complexos, como os obtidos na execução de workloads reais.

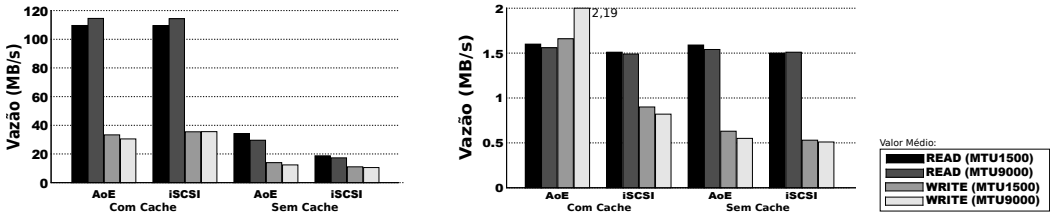
Em seguida, executamos um conjunto de macrobenchmarks sobre o mesmo ambiente de testes. Os macrobenchmarks têm o objetivo de medir o desempenho de diferentes configurações do sistema em workloads muito próximos dos encontrados em ambientes reais. Tais workloads, embora sintéticos, simulam os workloads encontrados em sistemas como servidores Web, servidores de arquivos, servidores de e-mail e bancos de dados transacionais, por exemplo. Além disso, alguns parâmetros como quantidade de memória disponível e MTU da interface de rede são alterados, visando verificar sua influência na vazão de disco alcançada por cada protocolo. Com base nestes resultados, analisaremos a eficiência dos protocolos em cada situação, bem como o impacto que a escolha do protocolo pode apresentar sobre o desempenho geral do sistema.

Após apresentar os resultados obtidos pelos macrobenchmarks, uma análise em termos de vazão, utilização de CPU e de rede é mostrada, relacionando-os com os obtidos pelos microbenchmarks. Por fim, uma discussão sobre os resultados encontrados é apresentada, enumerando algumas importantes observações sobre o desempenho dos protocolos, que, até onde sabemos, inexiste em trabalhos anteriores.

5. Microbenchmarks

Nesta Seção, apresentamos os resultados obtidos através da execução de microbenchmarks utilizando os protocolos AoE e iSCSI. Os resultados estão divididos entre operações de leitura e escrita, padrão de acesso sequencial e aleatório, utilização ou não de caches e MTU da interface de rede (1500 e 9000 bytes). Todos os microbenchmarks foram executados através da ferramenta *fio*, que possibilita a emissão de diversos padrões de acesso de I/O e diferentes parâmetros de forma customizada e flexível.

Os resultados dos experimentos são ilustrados na Figura 2. A Figura 2(a) mostra os resultados dos testes de leitura e escrita sequencial, enquanto a Figura 2(b) mostra os resultados obtidos em leitura e escrita aleatória, sob diferentes configurações. Os resultados apresentados no microbenchmark correspondem à vazão média alcançada em três execuções.



(a) Acesso Sequencial (b) Acesso Aleatório

Figura 2. Vazão de disco alcançada com diferentes parâmetros de microbenchmark.

De acordo com os gráficos, é possível observar que o protocolo AoE apresenta melhor desempenho no teste de leitura sequencial sem cache, cerca de 20%, independentemente do MTU. A perda de desempenho do protocolo iSCSI é atribuído ao *overhead* causado pelo processamento das camadas de rede adicionais, quando comparado ao AoE, que opera diretamente na camada de enlace. Todavia, quando o padrão de acesso é leitura aleatória, ainda sem cache, ambos apresentam o mesmo resultado. Neste caso, o *overhead* causado pelo acesso aleatório ao disco (*seek time*) na máquina *target* é predominante, suavizando o *overhead* causado pelas camadas de rede.

Por outro lado, quando os testes de leitura são executados com cache, ambos os protocolos apresentam grande melhora de desempenho quando executados de forma sequencial, devido ao mecanismo de *read-ahead* do sistema operacional. Pelo mesmo motivo, como testes com padrão aleatório não se beneficiam com *read-ahead*, nenhuma alteração no desempenho é obtida pela adição de caches nestes padrões de acesso.

O mesmo comportamento observado na leitura sequencial e sem caches ocorre nas escritas — o protocolo AoE apresenta melhor desempenho devido ao *overhead* das camadas de rede. Contudo, diferentemente das leituras, o protocolo AoE apresenta melhor desempenho mesmo em padrões de acesso aleatórios. Como escritas são, em geral, assíncronas, de forma que os dados são escritos apenas em memória na máquina *target* e posteriormente escritos em disco, o tempo de acesso aleatório ao disco é mitigado, tornando novamente significativo o *overhead* das camadas de rede do iSCSI, diminuindo assim seu desempenho em relação ao protocolo AoE.

Quando as escritas são executadas com cache, por outro lado, o protocolo iSCSI apresenta desempenho superior em todos os experimentos. Isso ocorre pois o subsistema SCSI da máquina *initiator* implementa a política de cache conhecida como *write-back*, onde os dados são escritos no cache da máquina local e enviados à maquina remota em momento oportuno. Assim, o alto *desempenho percebido* pelo protocolo iSCSI em testes de escrita com cache é maior que o *desempenho real*, na medida em que as operações de escrita não atingem imediatamente o disco da máquina destino.

6. Macrobenchmarks

Para analisar o comportamento de ambos os protocolos de armazenamento de redes SAN em situações próximas das encontradas em ambientes reais, uma sequência de macrobenchmarks foi executada. Para tal, utilizamos a ferramenta de benchmarks *filebench* por ser amplamente utilizada e por conter diversos workloads pré-definidos, simulando o padrão real de acesso em servidores com diferentes finalidades. Os workloads pré-definidos utilizados neste experimento são descritos abaixo.

Webserver: Simula o padrão de acesso de um servidor Web. O workload, predominantemente de leitura, executa um conjunto de operações do tipo *open-read-close* em diferentes arquivos de 16 KB, contando com um total de 50.000 arquivos. Além disso, há um fluxo de operações do tipo *append* que simula a escrita em arquivos de log.

Fileserver: Emula o funcionamento de um servidor de arquivos. O workload consiste em uma sequência de operações de criação, exclusão, escrita, leitura e operações sobre meta-dados em um conjunto de 10.000 arquivos, com tamanho de, em média, 128 KB.

Varmail: Simula a atividade de I/O de um servidor de e-mails que armazena cada mensagem como um arquivo individual. O workload consiste em uma sequência de operações do tipo *create-append-sync*, *read-append-sync*, leitura e exclusão de arquivos em um mesmo diretório.

Oltp: Emula o padrão de acesso de um banco de dados transacional. Este workload testa o desempenho de múltiplas operações aleatórias de leitura e escrita sensíveis à latência. O padrão de acesso é baseado no banco de dados Oracle 9i.

Em todos os workloads, o número de threads foi ajustado de forma a não sobrecarregar a CPU e influenciar o resultado obtido pelos testes. Nas Subseções seguintes, apresentamos os resultados obtidos através dos experimentos sob três dimensões: vazão no acesso ao disco remoto (Subseção 6.1), utilização de CPU (Subseção 6.2) e utilização de rede (Subseção 6.3).

6.1. Vazão de Disco

A Figura 3 apresenta os resultados obtidos através dos diferentes workloads. Nos gráficos, o eixo horizontal mostra a execução dos experimentos no disco local e utilizando os protocolos de armazenamento remoto. Para cada grupo de barras, são apresentadas as execuções com quantidades diferentes de memória disponível ao sistema (512 MB e 4 GB) e MTU (1500 e 9000 bytes). Para os testes de acesso local, apenas a quantidade de memória é alterada. Limitamos a quantidade de memória disponível ao sistema para amenizar o efeito das caches no resultado obtido, visto que são implementadas em diversos níveis do sistema operacional, como caches de blocos, páginas e mesmo dentro do subsistema SCSI. Mesmo assim, durante todos os experimentos houve memória suficiente às aplicações, não havendo uso de *swap*. O eixo vertical apresenta a vazão obtida.

Em workloads de leitura predominante, como o *webserver*, mostrado na Figura 3(a), o uso de caches atenua a diferença de vazão entre os protocolos. Como esperado, este resultado segue o comportamento encontrado na execução dos microbenchmarks de leitura sequencial. Em todos os casos de teste, invariavelmente, ao fornecer memória

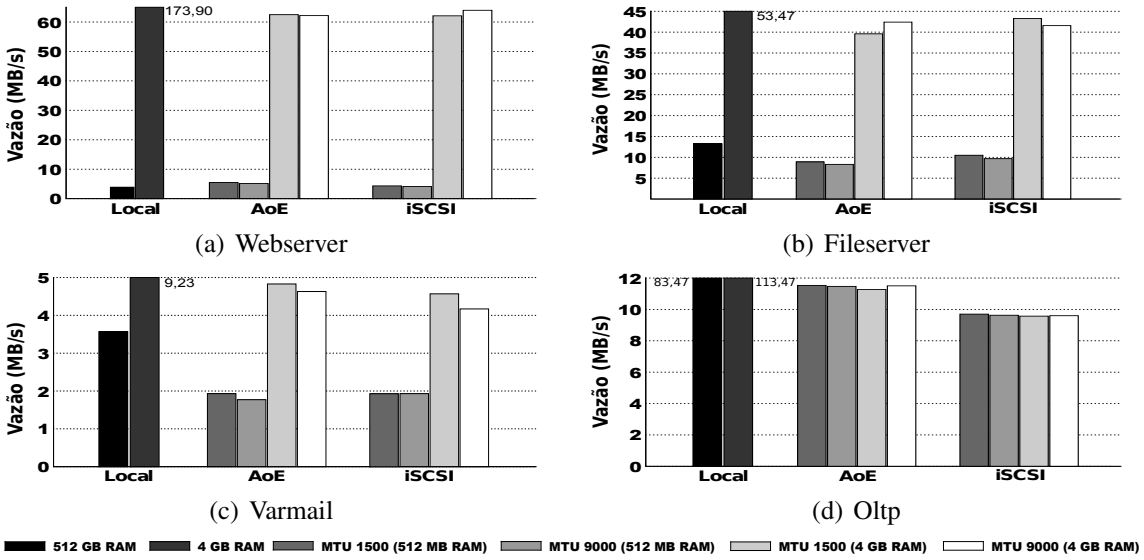


Figura 3. Vazão de disco alcançada por diferentes tipos de workloads em cenários de execução diversos. Onde indicado, o valor correto das barras foi modificado para melhor visualização dos resultados.

RAM suficiente ao sistema (4 GB), a variação dos resultados entre os protocolos é desprezível — inferior a 1%. Caso a quantidade de memória disponível ao sistema seja escassa, simulado pelo caso de teste de 512 MB, é possível observar que o protocolo AoE possui melhor desempenho, em torno de 8%, devido ao *overhead* de processamento das camadas rede. Entretanto, considerando a quantidade e tamanho dos arquivos criados pelo benchmark, e que servidores atuais possuem, em geral, mais do que 4 GB de memória RAM, afirmamos, com base nos experimentos, que *o uso de cache iguala o desempenho dos protocolos em workloads de leitura predominante*.

Já em workloads que executam massivas operações de escrita, como é o caso do *fileserver*, o protocolo iSCSI apresenta melhor desempenho, em média 9%, independentemente do MTU empregado. A utilização da política de cache *write-back* nas escritas, onde as operações são efetuadas localmente na memória da máquina *initiator* e depois executadas remotamente em momento oportuno, garante que o desempenho percebido seja superior ao do protocolo AoE. Nos casos testados, quanto maior a quantidade de memória, e consequentemente o espaço disponível para cache, maior é o aumento na vazão. Assim, concluímos que *o protocolo iSCSI apresenta melhor desempenho em workloads de escrita predominante, caso haja espaço suficiente para cache.*

Quando o workload consiste em um número balanceado de operações de escrita e leitura, como no workload *varmail*, o protocolo AoE mostra desempenho $11,2\%$ superior. Por um lado, a quantidade de operações de escrita não é suficientemente grande para que o iSCSI beneficie-se de seu melhor desempenho em escritas; por outro, as operações de leitura não são exclusivamente sequenciais (já que as operações ocorrem sobre múltiplos arquivos). Assim, o *overhead* do protocolo iSCSI torna-se determinante na vazão alcançada por este experimento.

Por fim, no caso do workload *oltp*, onde o padrão predominante é a escrita aleatória, o protocolo AoE apresenta melhor desempenho em todos os resultados, em

média 19%. Novamente, este resultado é semelhante ao encontrado nos testes com microbenchmarks. Todavia, é importante ressaltar que, apesar do aumento na quantidade de memória disponível para caches, não houve aumento significativo na vazão. Isso ocorre porque parte das operações executados pelo workload são realizadas sem cache (utilizando-se a flag O_DIRECT). Assim, notamos que em todas as configurações deste workload o protocolo AoE obteve melhor desempenho.

6.2. Utilização de CPU

A Figura 4 apresenta a utilização de CPU em cada caso de teste. O eixo horizontal apresenta os mesmos grupos de barras do teste anterior; o eixo vertical, a média do tempo de CPU gasto por cada operação de I/O emitida pelo benchmark. Quanto menor o tempo de processamento, mais eficiente em termos de CPU o protocolo é considerado.

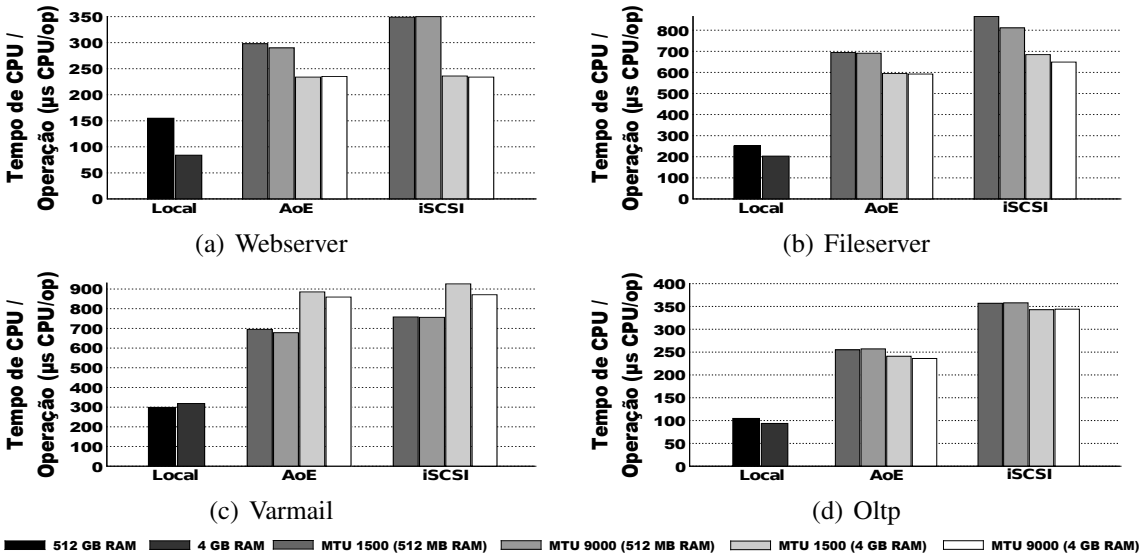


Figura 4. Tempo de CPU por operação de I/O em diferentes workloads.

Através dos resultados, é possível observar que o protocolo iSCSI apresenta maior utilização de CPU em todos os testes realizados. Este comportamento é esperado devido à camada em o que o protocolo opera, pois acrescenta o *overhead* de processamento das camadas de transporte e rede, quando comparado ao protocolo AoE, que opera diretamente na camada de enlace. Este fato pode ser percebido em todos os casos, e independe do resultado da vazão alcançada pelos protocolos. Entretanto, apesar do protocolo iSCSI apresentar maior utilização de CPU em todos os casos, o uso de caches diminui significativamente a diferença entre a utilização de CPU dos protocolos, de $22,2\%$ sem cache para 15% com cache, na medida em que distribui o *overhead* das operações custosas realizadas no servidor remoto entre as operações que acertaram a cache local. Assim, o uso de caches diminui significativamente a quantidade de processamento necessária por operação, aumentando a eficiência de CPU de ambos os protocolos.

Além disso, pode-se observar que em grande parte dos testes, como esperado, aumentar o MTU da interface de rede pode diminuir o tempo de CPU gasto por operação pelos dois protocolos. Isso ocorre pois, como o número de quadros Ethernet enviados é menor, menor é o tempo de processamento necessário para interpretá-lo.

6.3. Utilização de Rede

A Figura 5 apresenta a quantidade total de bytes transmitidos e recebidos divididos pelo número total de operações realizadas em cada caso de teste. O eixo horizontal apresenta os mesmos grupos de barras dos testes anteriores; o eixo vertical, o número médio de bytes por operação realizada pelo benchmark. É importante notar que, como parte das operações é executada em cache sem utilizar a interface de rede, o valor apresentado não reflete a quantidade real de rede utilizada por operação; entretanto, o valor é utilizado como métrica para comparação, na medida em que reflete a eficiência em termos de rede dos protocolos sob diferentes configurações.

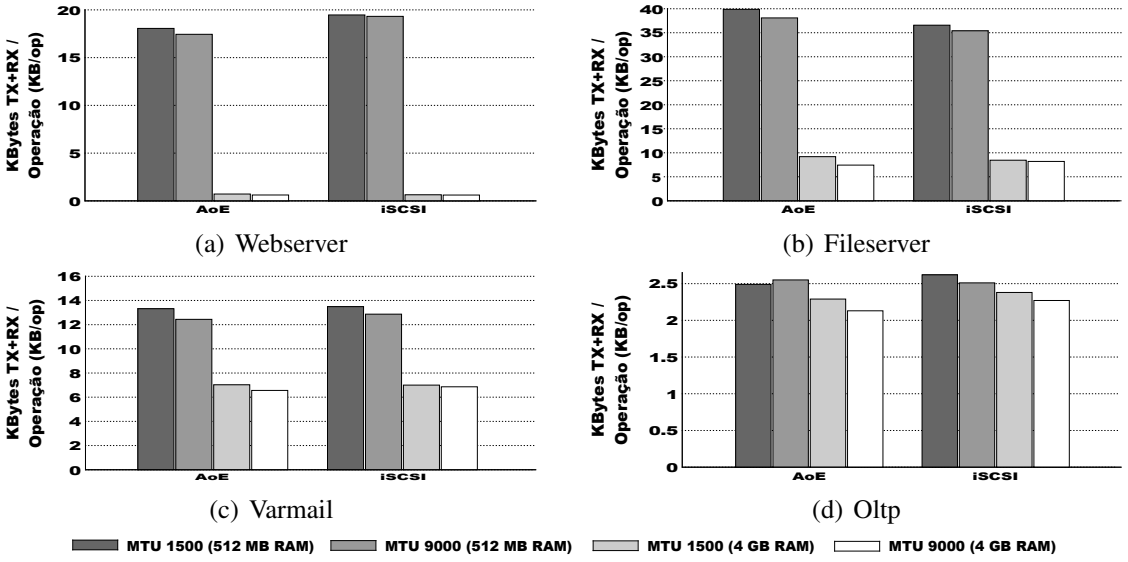


Figura 5. Quantidade de dados trafegado na rede (transmitido + recebido) por operação.

Como esperado, em todos os casos, quanto maior a quantidade de memória disponível ao sistema, menor é a quantidade de dados trafegados na rede, devido à maior quantidade de operações realizadas em cache local. Da mesma forma, na maioria dos casos testados, o AoE possui melhor eficiência de uso de rede (média de 3% para Webserver, Varmail e Oltp), devido à ausência do *overhead* introduzido pelas camadas adicionais no iSCSI, com uma exceção: no workload *fileserver*, como o iSCSI utiliza o cache com mais eficiência na escrita, consequentemente sua utilização de rede é menor ($3,26\%$).

Além disso, na maioria dos casos, aumentar o MTU na interface de rede melhora a eficiência na utilização de rede, pois mais dados são enviados em um mesmo quadro Ethernet, diminuindo assim o *overhead* introduzido pelos cabeçalhos. Nos poucos casos em que ocorre o contrário, memória RAM limitada é utilizada (512 MB); nesses, acreditamos que a alocação de buffers maiores para os quadros Ethernet diminua a quantidade de memória disponível para a cache de disco, diminuindo a taxa de acerto em cache e portanto aumentando a utilização de rede.

7. Discussão

Através do conjunto de experimentos executados, observamos que o protocolo AoE apresenta melhor desempenho global, de cerca de 9%. Contudo, ao fornecer quantidade suficiente de memória ao sistema, o protocolo iSCSI alcança melhor resultado em workloads

de escrita predominante, devido a mecanismos mais eficientes de cache, e resultado muito próximo ao AoE, mas ainda inferior, em workloads de leitura. Ainda assim, é possível afirmar que o protocolo iSCSI é uma boa alternativa para a maioria dos workloads, considerando que servidores atuais possuem quantidade considerável de memória RAM e que o protocolo iSCSI possui outras vantagens, como o fato de ser roteável e possibilidade de utilização de criptografia, através de IPSec.

Apesar disso, em workloads que não utilizam exaustivamente o mecanismo de cache do sistema operacional, como o Oltp, em que as caches são geralmente gerenciadas pela própria aplicação, o iSCSI também apresenta desempenho inferior. Nestes casos, sob todas as dimensões (vazão, CPU e rede) e protocolo AoE apresenta melhor desempenho, sendo, definitivamente, a melhor solução.

Outro fato a ser considerado é que o protocolo iSCSI apresenta, invariavelmente, *overhead* de CPU de cerca de 19% quando comparado ao AoE. Assim, caso haja restrições de processamento, a utilização do protocolo iSCSI deve ser evitada. Quanto à utilização de rede, a eficiência do protocolo iSCSI é cerca de 3% menor se comparado ao AoE; entretanto, considerando os fatos observados e a capacidade das redes atuais (1, 10 e até 40 Gbps), em poucos casos a capacidade da rede pode limitar a vazão no acesso ao disco. Finalmente, o aumento do MTU dos quadros Ethernet melhora a vazão em todos os casos, além de diminuir a quantidade de processamento e a utilização de rede. O aumento do MTU deve ser evitado somente quando a quantidade de memória for limitada, onde os buffers alocados para os quadros maiores possam concorrer com o mecanismo de cache de disco.

8. Conclusão

Neste trabalho, apresentamos uma análise experimental do desempenho dos protocolos de armazenamento remoto AoE e iSCSI. Estes protocolos, além de amplamente utilizados em SANs de ambientes de computação em nuvem e virtualização, não necessitam de hardware especializado e possuem implementações de código livre, diferentemente do protocolo FC. Através de um conjunto extensivo de experimentos, baseados tanto em microbenchmarks quanto em macrobenchmarks, analisamos as principais características desses protocolos em termos de vazão de dados, uso de memória cache, utilização de CPU e rede.

Observamos a partir de nossos experimentos que o protocolo iSCSI apresenta cerca de 9% de aumento na vazão em workloads de escrita predominante e desempenho muito próximo ao AoE em workloads de leitura (próximo a 1% no caso do webserver), caso haja memória suficiente para cache. Apesar disso, o AoE é a melhor opção para workloads que evitam o mecanismo de cache do sistema operacional, como bancos de dados Oltp. Ademais, o protocolo iSCSI é menos eficiente em termos de CPU e utilização de rede sob qualquer workload.

Dando continuidade a este trabalho, pretendemos analisar outros aspectos dos protocolos, como sua escalabilidade, confiabilidade na presença de falhas e interações com sistemas de arquivos. Acreditamos que nosso trabalho possa auxiliar administradores de infraestrutura a melhor entender o funcionamento destes protocolos sobre diferentes workloads, bem como suas implicações sobre utilização de memória, CPU e rede, ajudando-os a escolher o melhor protocolo para cada caso, além de dimensionar corretamente a

quantidade necessária de recursos.

Referências

Aiken, S., Grunwald, D., Pleszkun, A., and J.Willeke (2003). A performance analysis of the iscsi protocol. In *IEEE/11th Conference on Mass Storage Systems and Technologies*, pages 123–134.

Aycock, C. (2006). What is the difference between scsi and ata? `http://insidehpc.com/2006/04/07/what-is-the-difference-between-scsi-and-ata/`.

Chuang, H. and Wenbi, R. (2009). Modeling and performance evaluation of the aoe protocol. In *International Conference on Multimedia Networking and Security*, pages 609–6012. IEEE Press.

Follett, D. (2001). Distributed storage networking architectures: fibre channel, iscsi, ethernet, infiniband, hyper transport, rapid io and 3gio collide in the data center. In *IEEE International Symposium on Network Computing and Applications, 2001. NCA 2001.*, page 159.

Gerdelan, A., Johnson, M., and Messom, C. (2007). Performance analysis of virtualised head nodes utilising cost-effective network attached storage. In *Proceedings of the Asian Particle Accelerator Conference*.

Hopkins, S. and Coile, B. (2001). Ata over ethernet specification. `http://support.coraid.com/documents/AoEr11.txt`.

LoBue, M. T., Mason, H., Hammond-Doel, T., Anderson, E., Alexenko, M., and Clark, T. (2002). Surveying today's most popular storage interfaces. *Computer*, 35(12):48–55.

Satran, J., Meth, K., Sapuntzakis, C., Chadalapaka, M., and Zeidner, E. (2004). Internet Small Computer Systems Interface (iSCSI). RFC 3720 (Proposed Standard). Updated by RFCs 3980, 4850, 5048.

Tan, Y., Jin, J., Cao, Y., and Zhu, L. (2005). A high-throughput fibre channel data communication service. In *Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies.*, pages 975–978.

Voruganti, K. and Sarkar, P. (2001). An analysis of three gigabit networking protocols for storage area networks. In *IEEE International Conference on Performance, Computing, and Communications, 2001.*, pages 259–265.

Zhou, C. and Chuang, H. (2009). A performance analysis of the aoe protocol. In *Proceedings of the 5th International Conference on Wireless communications, networking and mobile computing*, pages 3938–3941. IEEE Press.