PEC: Protocolo Epidémico para Centros de dados*

Miguel Branco, João Leitão, Luís Rodrigues
miguel.branco@ist.utl.pt, jleitao@gsd.inesc-id.pt, ler@ist.utl.pt

INESC-ID, Instituto Superior Técnico, Technical University of Lisbon

Resumo Os protocolos de difusão epidémica, pelas suas características, são excelentes candidatos para suportar a difusão e/ou recolha de informação em centros de dados de grande-escala. Infelizmente, neste contexto particular, soluções alheias à topologia da rede podem facilmente saturar os comutadores nos níveis hierárquicos mais altos da rede. Neste artigo apresentamos um Protocolo Epidémico para Centros de dados (PEC) que assegura uma distribuição adequada da carga pelos comutadores da rede, evitando que o tráfego epidémico seja uma fonte de estrangulamento do sistema. O nosso protocolo combina características importadas de algoritmos propostos anteriormente, as quais são enriquecidas com um protocolo de gestão da filiação e mecanismos de controlo de fluxo cientes da topologia. Os benefícios da nossa solução são ilustrados por uma avaliação experimental que compara o seu desempenho com o desempenho de soluções existentes na literatura.

1 Introdução

Os protocolos de difusão epidémica suportam eficazmente a disseminação de informação em sistemas com grande número de participantes. Duas das razões principais para o seu sucesso são a sua robustez e o facto de proporcionarem uma distribuição de carga uniforme por todos os processos. Esta classe de protocolos tem sido utilizado para diversos fins, incluindo difusão fiável [1,2], agregação de dados [3,4], manutenção de filiação [5,6], entre outros [7].

Outro aspecto importante destes protocolos é que permitem evitar os fenómenos oscilatórios que ocorrem noutras formas de difusão fiável [1]. Sendo baseados em interacções ponto-a-ponto, não necessita que os comutadores suportem difusão IP. Todas estas propriedades fazem com que os protocolos epidémicos sejam apropriados à operação em centros de dados de larga escala.

Infelizmente, neste contexto, abordagens alheias à topologia de rede podem facilmente saturar os comutadores de nível mais alto de um centro de dados. Por essa razão, os protocolos epidémicos utilizados pelos centros de dados têm de se basear numa abordagem ciente da topologia. Alguns protocolos

* Este trabalho foi parcialmente suportado pela FCT – Fundação para a Ciência e a Tecnologia sob os projectos PEst-OE/EEI/LA0021/2011 e HCPI na bolsa (PTDC/EIA-EIA/102212/2008). Os autores gostariam de agradecer a José Orlando Pereira e Miguel Matos pela ajuda na concretização do CLON.

cientes da topologia foram propostos na literatura, como HiScamp [8], Rumor Hierárquico [9] e CLON [10]. Como iremos demonstrar, estes protocolos possuem algumas limitações, nomeadamente: *i*) apesar de tentarem reduzir a carga imposta nos comutadores de alto nível, esta pode ainda assim ser excessiva, *ii*) a utilização de recursos não é eficiente, ou *iii*) a sua latência pode ser alta.

Neste artigo, propomos um protocolo epidémico concebido para operar em centros de dados. O nosso protocolo, PEC, é baseado numa abordagem ciente da topologia, tirando partido de diferentes propriedades dos protocolos referidos anteriormente. Contrariamente ao trabalho prévio, a nossa solução utiliza um mecanismo de manutenção da filiação que cria uma rede sobreposta adaptada à topologia física dos centros de dados. O PEC utiliza esta rede sobreposta para oferecer um mecanismo de difusão que é parcialmente determinista, ciente da topologia e robusto. Além disso, propomos um mecanismo de controlo de fluxo que limita a taxa de transmissão sobre os comutadores de alto nível. O mecanismo de difusão tem ainda em conta a frescura das mensagens durante a sua disseminação. Estas propriedades permitem ao PEC distribuir adequadamente a carga pelas diferentes camadas da rede de comutação do centro de dados.

A combinação destas características permite a PEC suportar um débito agregado superior ao das soluções anteriores, sem sobrecarregar os comutadores. As vantagens da nossa solução são ilustrados por uma avaliação experimental que compara o desempenho do PEC com o desempenho de protocolos concorrentes.

O resto do artigo está organizado da seguinte forma: a Secção 2 discute o trabalho relacionado. A Secção 3 descreve as actuais topologias de centros de dados. A Secção 4 motiva, apresenta e descreve o PEC. Na Secção 5 avaliamos e comparamos o nosso trabalho com as soluções anteriores descritas na literatura e, por fim, a Secção 6 conclui o artigo.

2 Trabalho Relacionado

Vários trabalhos propuseram variantes da propagação epidémica com o objectivo de a tornar ciente da topologia. Nesta secção discutimos os exemplos mais relevantes e comparamo-los com o PEC.

Um ponto importante a ter em conta antes de analisar o trabalho relacionado é que a maioria dos protocolos epidémicos com aplicação prática em sistemas de grande dimensão não exige que cada processo possua uma visão global da filiação do sistema. Um protocolo com filiação total selecciona aleatoriamente, entre toda a população, os alvos para executar trocas epidémicas. No entanto, manter filiação total não é exequível em cenários de larga escala. Os protocolos epidémicos são suportados por um serviço de amostragem da população, que oferece a cada nó uma *vista parcial* da filiação. A união das vistas parciais define uma rede sobreposta sobre a qual os protocolos epidémicos são executados.

Há duas formas distintas de tornar um protocolo epidémico ciente da topologia: uma consiste em alterar o serviço de amostragem da população, enviesando a escolha de vizinhos de acordo com um algoritmo que tome em conta a topologia da rede; outra consiste em actuar ao nível do protocolo epidémico (por exemplo,

influenciando as probabilidades de seleccionar determinados vizinhos da vista parcial). É obviamente possível combinar estas duas técnicas.

Amostragem Ciente da Topologia O protocolo HiScamp [8] é um protocolo de filiação distribuído concebido sobre o serviço de amostragem de pares Scamp [11]. O HiScamp organiza os nós em grupos de acordo com a topologia de rede, mantendo um número limitado de caminhos entre grupos distintos e, consequentemente, uma carga limitada nos comutadores utilizados nesses caminhos. Porém, e contrariamente ao nosso trabalho, a rede sobreposta resultante possui poucos caminhos entre nós em núcleos diferentes, produzidos de forma aleatória. Isto tem um impacto negativo significativo na latência e fiabilidade do protocolo epidémico que executa sobre a rede. Os autores do HiScamp argumentam que a sua solução pode ser estendida a redes com diferentes números de camadas, no entanto a solução proposta aumenta os efeitos negativos descritos anteriormente.

Tanto quanto sabemos, ainda não foi proposto nenhum serviço de amostragem de população com o objectivo de se adequar a centros de dados. Para além disso, e como ficará claro no resto do artigo, é dúbio que uma solução eficaz possa ser concebida actuando exclusivamente ao nível da amostragem.

Protocolos Epidémicos Cientes da Topologia Em contraste com as soluções anteriores, que actuam ao nível do serviço de amostragem, uma abordagem alternativa consiste em manipular directamente os padrões de comunicação exibidos pelo protocolo de difusão epidémica.

Um dos protocolos que segue esta abordagem é o Rumor Hierárquico [9]. Este protocolo faz uma selecção não-uniforme de alvos para trocas epidémicas, de forma a escolher com maior probabilidade nós mais próximos. A solução pode ter em consideração vários níveis de distância, oferecendo a possibilidade de capturar a topologia das redes dos centros de dados. No entanto, o mecanismo de difusão epidémica utilizado pelo Rumor Hierárquico é completamente aleatório por natureza, e contrariamente ao PEC, não tem em conta a frescura das mensagens na escolha dos alvos durante o processo de disseminação. Isto torna esta solução bastante ineficiente na utilização de recursos. Como demonstrado através das nossas experiências, esta estratégia provoca uma carga excessiva nos comutadores mais críticos do centro de dados. Consequentemente, o débito máximo do Rumor Hierárquico é muito menor do que o conseguido pelo PEC.

Abordagens Híbridas Contrariamente a todos os trabalhos discutidos acima, a nossa solução baseia-se numa abordagem integrada, que combina um serviço de filiação ciente da topologia e parcialmente determinista com um mecanismo de difusão epidémica também ciente da topologia. CLON [10] é um trabalho recente que segue uma abordagem semelhante e que partilha o objectivo de suportar protocolos epidémicos eficientes e cientes da topologia em, mas também entre, centros de dados. De uma maneira semelhante ao HiScamp, o CLON utiliza o protocolo Scamp para construir uma rede sobreposta de suporte à disseminação

epidémica. Este serviço de filiação promove a manutenção de vizinhos distantes em cada nó do sistema. Sobre a rede sobreposta resultante, o CLON executa um mecanismo de difusão epidémica que manipula a probabilidade de seleccionar um vizinho de acordo com a frescura da mensagem.

No entanto, e por oposição à nossa solução que pretende suportar um protocolo epidémico eficiente e fiável dentro de um centro de dados, CLON tem como principal objectivo suportar um protocolo epidémico entre múltiplos centros de dados. Isto reflecte-se claramente no serviço de filiação utilizado pelo protocolo, que apenas distingue vizinhos locais e remotos, sendo portanto incapaz de capturar topologias hierárquicas com vários níveis. Isto aumenta significativamente a latência média da disseminação de mensagens. Além disso, contrariamente à nossa solução, o CLON não aplica nenhuma forma de enviesamento determinista na escolha dos alvos para trocas e não concretiza nenhum mecanismo de controlo de fluxo. Isto faz com que consuma bastante largura de banda nos comutadores de topo quando vários nós injectam mensagens na rede simultaneamente.

Como as descrições anteriores indicam, tanto quanto nos é possível afirmar, PEC é o primeiro protocolo epidémico que combina um serviço de filiação ciente da topologia e parcialmente determinista, um mecanismo de difusão que não só é ciente da topologia mas também parcialmente determinista e um mecanismo de controlo de fluxo para suportar uma solução eficiente, fiável e ciente da topologia, especialmente concebida para operar em redes de centros de dados.

3 Arquitectura de Rede

Nesta secção descrevemos a topologia típica das redes dos centros de dados. As topologias de rede de centros de dados seguem tipicamente uma arquitectura de três níveis ou de dois níveis. O nosso protocolo foi concebido para operar eficientemente em topologias similares às que aqui se descrevem. Mais precisamente, temos como objectivo atingir os seguintes objectivos: *i*) minimizar a carga imposta nos comutadores da infraestrutura de rede; *ii*) utilizar os recursos de forma eficiente, maximizando o débito face a um limite máximo de comunicação; e *iii*) minimizar a latência média do processo de difusão epidémica.

Arquitectura de Três Níveis A arquitectura mais comum em redes de centros de dados é a arquitectura de três níveis [12], em que a rede é caracterizada por três níveis hierárquicos de equipamento de rede.[1]

O nível superior, normalmente chamado *nível nuclear*, é composto por um único comutador que possui ligações a múltiplos comutadores de agregação no segundo nível. Estes comutadores ligam-se por sua vez a múltiplos comutadores periféricos, que formam o nível mais baixo da hierarquia. Estes são comutadores "*top-of-rack*" e ligam-se directamente aos servidores. Numa configuração

[1] Iremos utilizar a palavra comutador para nos referirmos a todo o equipamento de rede, sejam eles encaminhadores nível 3 ou comutadores nível 2, sem diferenciação, uma prática comum na bibliografia relacionada [12].

típica, cada comutador periférico pode ligar-se desde a 20 a 80 nós [12]. No resto deste artigo vamo-nos referir ao conjunto de nós ligados directamente ao mesmo comutador periférico como um *agregado*.

Para minimizar a carga imposta no comutador nuclear, algumas configurações utilizam múltiplos comutadores de núcleo. Neste tipo de arquitectura, cada um destes comutadores liga-se a todos os comutadores do nível de agregação, oferecendo caminhos redundantes entre qualquer par de comutadores desse nível.

Em pequenas empresas e universidades, a arquitectura é tipicamente simplificada para ter apenas dois níveis, eliminando o nível de agregação e ligando directamente os comutadores periféricos ao comutador nuclear.

Abstrair a Topologia Física De modo a tornar o nosso protocolo o mais genérico possível, fazemos algumas hipóteses que nos permitem abstrair a topologia física, nomeadamente considerando que o esquema de numeração usado para identificar nós codifica alguma informação sobre a localização dos mesmos. Note-se que hipóteses semelhantes são usadas no trabalho relacionado [9].

Desta forma, assumimos que todos os comutadores no centro de dados estão numerados segundo um esquema de numeração hierárquico. Tratamos o espaço de identificadores de qualquer conjunto de comutadores ligado ao mesmo comutador como um espaço circular. Assumimos ainda que o endereço IP de um nó permite a qualquer nó determinar localmente o identificador do comutador periférico ao qual esse nó está ligado [2].

Considere-se o seguinte exemplo: o nó n está ligado ao terceiro comutador periférico que está por sua vez ligado ao quarto comutador de agregação, o qual está ligado directamente ao comutador nuclear do nível superior. Atribuímos explicitamente o identificador 0 a este comutador para permitir uma adaptação trivial da nossa solução a um cenário com múltiplos centros de dados. Neste cenário, o identificador do comutador de agregação é 0.4. Consequentemente, o identificador do comutador periférico a que n está conectado é 0.4.3.

4 PEC

Perspectiva Global A nossa solução segue, e estende, as ideias subjacentes à concepção do Rumor Hierárquico [9] e do CLON [10], as quais refinamos com novos mecanismos. O PEC é composto por três componentes principais que se complementam para criar um protocolo de difusão epidémica ciente da topologia, fiável e eficiente. Em primeiro lugar, é concretizado um serviço de filiação que oferece a cada nó um conjunto de vistas parciais. As vistas parciais são geridas de tal forma que os seus conteúdos têm em consideração a topologia da rede, através de uma forma de enviesamento determinista do seu conteúdo. Sobre este serviço de filiação, criámos um esquema de disseminação especialmente construído para tirar partido das características da rede sobreposta resultante.

[2] A lógica da função que permite obter a localização em função do IP está fora do âmbito deste artigo.

Este esquema de disseminação apresenta um grau controlado de determinismo no que toca aos padrões de comunicação. Isto permite reduzir a carga imposta aos comutadores mantendo tolerância a faltas. Tem também em conta a frescura das mensagens quando as reencaminha. Por último, os mecanismos acima descritos são complementados por um mecanismo de controlo de fluxo baseado em regulação da taxa de transmissão.

A nossa solução pode ser facilmente configurada de forma a suportar um número arbitrário de níveis hierárquicos na topologia do centro de dados. No entanto, para simplificar a exposição do protocolo, optámos por descrever a sua operação considerando uma arquitectura de três níveis. Seguidamente, descrevemos a operação dos três componentes principais do PEC.

Serviço de Filiação Apoiamo-nos num serviço de filiação que opera de forma similar ao Cyclon [13]. Na nossa solução, no entanto, cada nó mantém um conjunto de vistas parciais distintas, cada uma encapsulando informação referente a um diferente nível hierárquico. Ao contrário das soluções anteriores, o nosso serviço de filiação utiliza um processo de enviesamento na composição das vistas, promovendo a emergência de estruturas da rede sobreposta que são semelhantes à manutenção de múltiplas árvores redundantes ligando os vários aglomerados.

Tal como no Cyclon, cada nó troca periodicamente amostras dos conteúdos das suas vistas com um alvo aleatório. Quando trocam estas amostras, os nós adicionam o seu identificador à mensagem que enviam para o seu par. Assumimos que os identificadores dos nós que estão guardados nas vistas parciais são enriquecidos com um contador de idade, que é incrementado periodicamente pelos nós de modo a reflectir o tempo que passou desde a criação desse identificador.

Quando um nó recebe uma amostra da filiação do sistema através de um par, ele utiliza essa informação para actualizar os conteúdos das vistas que ele próprio mantém localmente, respeitando um conjunto de restrições impostas a cada vista, as quais descrevemos mais à frente. Adicionalmente, os nós dão preferência a identificadores com menor idade, o que aumenta a probabilidade do nó que produziu o identificador ainda se encontrar activo.

No nosso sistema, cada nó individual mantém L vistas parciais independentes, onde L é o número de níveis na hierarquia da topologia de rede. Chamamos a estas vistas PV_i em que i indica o nível que cada vista codifica (a nossa solução suporta um número arbitrário de níveis hierárquicos). Para o caso de uma arquitectura de 3 níveis, um nó n terá as seguintes vistas parciais:

PV_0 representa o nível hierárquico mais baixo da topologia. Esta vista deve conter o identificador de todos os nós no mesmo aglomerado que n. Para beneficiar o esquema de disseminação, fazemos com que o identificador de n apareça também na vista PV_0 de n. Os conteúdos destas vistas são mantidos ordenados, por ordem crescente de identificadores dos nós do aglomerado. O tamanho desta vista depende da topologia de rede do centro de dados, sendo igual ao número de nós em cada aglomerado.

PV_1 contém identificadores dos nós com que n pode comunicar através de um único comutador de agregação. Os nós tentam manter nesta vista identificadores

oriundos de K_1 agregados diferentes, enviesados de forma determinista. Os agregados alvo do nó n são seleccionados considerando o identificador do comutador periférico ao qual n está ligado. Se n está ligado ao comutador periférico com identificador $c.a.e$, n vai dar preferência a nós ligados a comutadores com identificadores entre $c.a.(e * K_1 + 1)$ e $c.a.((e + 1) * K_1)$ (relembramos que o espaço de identificadores é circular em cada nível hierárquico).

PV_2 codifica o nível mais alto na hierarquia da topologia de rede. Esta vista contém identificadores de nós com que n só consegue comunicar através do comutador nuclear. Como descrito para PV_1, esta vista parcial é construída tentando manter K_2 identificadores diferentes de comutadores de agregação, através de um enviesamento determinista. Se n está ligado através de um comutador de agregação com identificador $c.a$, ele vai dar preferência a nós ligados através de comutadores de agregação com identificadores desde $c.(a*K_2+1)$ a $c.((a+1)*K_2)$, ignorando o identificador do comutador periférico dos mesmos.

Para suportar mais níveis hierárquicos, os nós apenas têm de manter mais vistas parciais de tamanho reduzido, o que não tem um efeito tão negativo sobre a na fiabilidade e a latência das mensagens como o provocado pela a escassez de caminhos remotos em soluções como o HiScamp.

O tamanho das vistas parciais descritas acima (excepto PV_0) deve respeitar $K_i + R - 1 < f$, em que R é um factor de redundância associado ao mecanismo de disseminação aplicado pelo PEC, que explicaremos mais à frente. Determinámos experimentalmente que o valor de $R = 2$ produz resultados adequados para as topologias alvo.

Mecanismo de Difusão Epidémica O algoritmo de disseminação que utilizamos opera de forma semi-determinista sobre a rede sobreposta definida pelo serviço de filiação. O algoritmo baseia-se no princípio sugerido em vários trabalhos anteriores, incluindo o CLON [10], de que para reduzir a latência, as mensagens devem ser disseminadas em primeiro lugar para nós distantes e só depois a nós mais próximos. No entanto, contrariamente à solução de Rumor Hierárquico [9], no nosso algoritmo os nós operam no modelo "*infect and die*" [14], isto é, cada nó só processa uma dada mensagem uma única vez. Quando o nó processa uma mensagem pela primeira (e única) vez, (re)transmite-a para f vizinhos, em que f é um parâmetro designado por "*fanout*".

De modo a aplicar o viés acima referido, cada mensagem enviadas pelo PEC inclui um contador T que indica o número de vezes que ela já foi retransmitida. De forma a ter a tolopogia hierárquica em consideração, a disseminação é controlada por um conjunto de parâmetros $\pi_i, i \in [0, L]$, que limita o número de vezes que uma mensagem é retransmitida a cada nível hierárquico (note-se que π_0 funciona como o típico parâmetro *time to live* de soluções epidémicas uniformes [2]). Para saber a que nível hierárquico deve ser reencaminhada uma determinada mensagem, os nós determinam o i máximo tal que $m.T < \pi_i$

Por último, para assegurar que o protocolo impõe uma carga limitada em cada comutador, utilizamos a vista PV_0 do serviço de filiação para atribuir deterministicamente papeis especializados aos nós de cada aglomerado.

Um factor de redundância, R, é usado na atribuição de papeis aos nós de cada aglomerado da seguinte forma: os primeiros R nós da vista PV_0 são, em cada aglomerado, escolhidos para disseminar informação no nível mais alto da hierarquia (são denominados *nós nucleares*). Os R seguintes nós na vista são responsáveis pela disseminação no nível seguinte (sendo denominados *nós de agregação*). Os restantes nós apenas disseminam informação ao nível mais baixo da hierarquia (*nós periféricos*). Note-se que são mantidas vistas totais de cada aglomerado, e que estas vistas são ordenadas por ordem dos identificadores, pelo que todo o aglomerado possui uma perspectiva coerente dos papeis de cada membro. Como referimos acima, determinámos experimentalmente que o valor de $R = 2$ produz resultados adequados para as topologias alvo.

Quando um nó produz ou recebe uma mensagem pela primeira vez, guarda-a na sua fila local. Periodicamente, cada nó consulta a sua lista de mensagens e processa-as até esgotar a sua quota para esse período (a quota é definida pelo mecanismo de controlo de fluxo que explicaremos mais à frente). Recordamos que cada mensagem que é processada é retransmitida para f outros nós.

Para cada mensagem na fila de um nó, existe um conjunto de regras para a sua transmissão. Considerando o contador de rondas T da mensagem, o nó começa por descobrir qual o nível hierárquico no qual a mensagem deve ser transmitida. Designe-se este nível por h.

– Se o nó não é responsável por transmitir no nível h (considerando a atribuição de papeis acima referida), existem dois cenários possíveis:

• Se a mensagem foi produzida localmente ou oriunda de um um aglomerado remoto, o nó redirecciona a mensagem para os R nós que, no seu aglomerado, são responsáveis por transmitir no nível h (sem incrementar o contador T). Para além disso, reencaminha a mensagem para $f - R$ nós adicionais do seu agregado, incrementando o contador T.

• Se por outro lado a mensagem foi recebida de um elemento do mesmo agregado, o nó envia a mensagem para f vizinhos do seu próprio agregado, incrementando o contador T.

Independentemente do caso, se o nível pretendido para a mensagem é superior ao papel do nó, o nó mantém a mensagem na sua fila local e configura o valor T da mensagem para um número apropriado ao papel que ele desempenha, de forma a garantir uma disseminação futura a esse nível.

– Se o nó é responsável por fazer a disseminação no nível h, existem também dois cenários possíveis:

• Se a mensagem já chegou ao nível periférico (0), o nó envia apenas a mensagem para f vizinhos do seu agregado, incrementando o contador T.

• Se a mensagem deve ser disseminada num nível superior ao periférico ($h > 0$, isto é, o nó em causa é um nó nuclear ou de agregação), esta é enviada para todos os K_h vizinhos da vista correspondente, assim como para os outros $R - 1$ nós do aglomerado também responsáveis pelo nível h. Para além disso, a mensagem é reencaminhada para $f - K_h - (R - 1)$ nós adicionais do seu agregado, começando pelos responsáveis do nível $h - 1$, e assim sucessivamente, configurando o contador T com valores apropriados a cada nível.

No entanto, todo este passo só é feito por um dos R nós do aglomerado responsáveis pelo nível h por ronda, usando um critério determinista com base no número da ronda. Isto evita a transmissão redundante de mensagens nos comutadores. Uma mensagem que é processada por uma das réplicas do nível h, pode ser descartada pelas restantes réplicas desse nível.

Este processo permite propagar uma mensagem por todos os nós do centro de dados de uma forma eficiente, promovendo um número controlado de mensagens redundantes, mascarando omissões de mensagens e falhas de nós.

Controlo de Fluxo De modo a assegurar um limite de tráfego gerado pelo PEC, criámos um mecanismo distribuído simples, mas eficaz, de controlo de fluxo. Relembramos que cada nó no sistema mantém uma fila local que contém as mensagens que este precisa de encaminhar para os seus pares. Para limitar o número de mensagens encaminhadas por ronda, usamos valores de *quota* para cada nó. A cada ronda, os nós extraem da sua fila um número m de mensagens tal que $m * f \leq quota$. Isto pressupõe evidentemente que $quota \geq f$. A quota de cada nó depende do papel que desempenha no aglomerado, uma vez que são atribuídas quotas diferentes para a transmissão em cada nível hierárquico. A cada nível i, a quota configurada depende da carga alvo ca_i da seguinte forma: $q_i = \frac{ca_i}{Naglomerados} \frac{f}{|PV_i|}$.

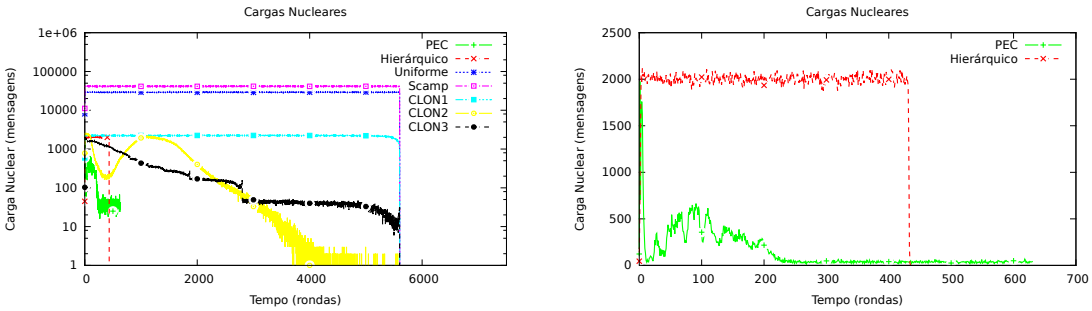
Tolerância a Faltas Os nós de um mesmo aglomerado mantêm entre si ligações TCP, como um mecanismo simples de detecção de falhas. Quando a ligação para um nó falha, os restantes membros do grupo retiram-no da sua vista e reconfiguram os papéis, de modo a manter R réplicas para cada nível.

5 Avaliação

Para poder avaliar o desempenho da nossa solução, simulámos o PEC, considerando uma topologia de rede para o centro de dados com 1 comutador nuclear, ligado a 8 comutadores de agregação, que ramificavam em 10 comutadores periféricos com agregados de 32 nós. O total de nós no sistema é 2.560. Todas as experiências foram conduzidas utilizando o simulador PeerSim [15], usando o seu motor de simulação estimulado por eventos discretos.

Para permitir a comparação com sistemas anteriores, executámos outras soluções existentes na literatura sobre a mesma topologia. Em particular, testámos: *i*) um protocolo epidémico uniforme com filiação total; *ii*) um protocolo epidémico sobre o serviço de filiação *Scamp* [11]; *iii*) o protocolo *Rumor Hierárquico* [9] com filiação total; e finalmente *iv*) o sistema CLON [10]. Todas as concretizações dos vários protocolos foram validadas experimentalmente.

Configurámos cada protocolo de forma a atingir 100% de fiabilidade. Estabelecemos o parâmetro f do protocolo epidémico uniforme a 13 e o factor C de redundância do Scamp de forma a gerar um grau o mais aproximado possível. Devido ao número de parâmetros do CLON, optámos por realizar os testes com 3



(a) Carga para todos os protocolos

(b) Detalhe dos protocolos RH e PEC

Figura 1. Cargas no comutador nuclear

configurações distintas: $CLON1$ adiciona os nós ao sistema de forma aleatória, atingindo um número de conexões remotas aproximado de um terço de todas as conexões; $CLON2$ usa um valor de redundância C grande o suficiente para assegurar um grau elevado para todos os nós, de forma a podermos limitar o "fanout" e número de rondas da mensagem no nível core; $CLON3$ utiliza um método externo para adicionar os nós ao sistema de forma a que o contacto usado esteja no nível hierárquico mais próximo possível, permitindo um número mais reduzido de conexões remotas. Todas as implementações do CLON foram realizadas em modo "eager push", assim como os restantes protocolos. Para além disso, o mecanismo de "lazy push" do CLON iria apenas reduzir a carga média nos comutadores à custa da latência das mensagens (a qual já é penalizada), e não a carga máxima, a qual queríamos limitar. Para o Rumor Hierárquico, alterámos o valor K de geração de probabilidades para 6, a fim de aumentar as probabilidades o suficiente para atingir a fiabilidade objectivo. Mantivemos o "fanout" de 13 para o nosso protocolo, adoptando valores de 3 e 4 para K_2 e K_1, respectivamente. Testámos todos os protocolos no modelo *infect-and-die*, adicionando limites de quota de forma a podermos limitar a carga no nível nuclear de igual forma. Para tal recorremos ao motor cíclico do simulador para despoletar operações periódicas nos nós, e usamos o motor de eventos discreto do simulator para modelar o envio de mensagens com latência associada.

Para a nossa primeira experiência, injectámos 800 mensagens no sistema a cada 10 rondas, para um total de 5.600 mensagens. Medimos a carga nos comutadores a cada ronda. A Figura 1(a) mostra a carga no nível nuclear.

Podemos observar que mesmo a carga mínima produzida pela solução epidémica uniforme e pelo Scamp ultrapassam em muito o limite configurado para os outros protocolos. O baixo número de mensagens que enviam por ronda prejudica-os gravemente na latência do processo de disseminação, fazendo com que as mensagens fiquem em fila de espera em cada nó demasiado tempo, atingindo latências na ordem das 1.400 rondas contra as cerca de 330 do Rumor Hierárquico e do PEC. A configuração $CLON1$ não pode tirar partido do número de rondas

limitado para o nível nuclear (caso contrário perderia fiabilidade devido ao reduzido número de caminhos remotos utilizados para cada mensagem) e utiliza então a carga máxima nesse nível durante toda a simulação. No entanto, obtém latências na ordem das 3.756 rondas, superiores às melhores por um factor superior a 10. As restantes configurações do mesmo protocolo atingem uma carga inferior à máxima durante parte da simulação, enquanto melhoram relativamente a latência para cerca das 3.100 rondas.

O Rumor Hierárquico manteve a carga máxima no comutador nuclear durante toda a simulação, gastando mais recursos totais que o PEC, que apenas transmite mensagens recentes pelos mesmos caminhos. É ainda visível que a carga máxima no PEC é apenas atingida no início da simulação, quando todas as mensagens são recentes e necessitam de ser transmitidas pelos caminhos de nível superior. Quando mensagens recentes e antigas coexistem, a carga no núcleo é diluída pelas rondas, sem penalização na latência. Uma visão mais pormenorizada da carga destes dois protocolos pode ser observada na Figura 1(b).

Para a segunda experiência, limitámos a carga máxima total no comutador nuclear. Para cada protocolo, observámos o débito que é possível atingir sem exceder a carga alvo neste comutador. A Figura 2 ilustra os resultados.

É visível que o PEC oferece o melhor débito quando limitamos a carga máxima total, por um factor superior a 10 sobre o segundo melhor protocolo. Previsivelmente, dado os resultados acima obtidos, o protocolo epidémico uniforme, o Scamp e as duas primeiras configurações do *CLON* não conseguem transmitir nenhuma mensagem impondo uma carga tão reduzida.

Observámos também experimentalmente que o desenho do PEC inclui redundância suficiente de forma a tolerar falhas sequenciais de nós, sem impacto significativo na sua capacidade de disseminação. Neste teste verificámos a robustez da solução com falhas até 30% do nós.

6 Conclusão

Neste artigo apresentámos PEC, um protocolo epidémico para centros de dados. Mostrámos os benefícios de adicionar determinismo à difusão epidémica,

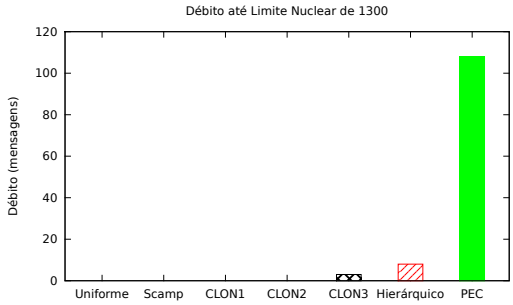


Figura 2. Débito até o limite no núcleo ser atingido

criando um protocolo que assenta em três componentes cientes da topologia: um serviço de filiação, um mecanismo de disseminação e um mecanismo de controlo de fluxo. Avaliámos ainda a nossa solução face ao trabalho relacionado, tendo observado um aumento do débito suportado, sem sobrecarga dos comutadores nucleares nem perda de fiabilidade.

Referências

1. Birman, K., Hayden, M., Ozkasap, O., Xiao, Z., Budiu, M., Minsky, Y.: Bimodal multicast. ACM Trans. Comput. Syst. **17** (1999) 41–88
2. Leitão, J., Pereira, J., Rodrigues, L.: HyParView: A membership protocol for reliable gossip-based broadcast. In: Proc. of the DSN'07, Edimburgh, UK (2007) 419–429
3. Gupta, I., Renesse, R.v., Birman, K.: Scalable fault-tolerant aggregation in large process groups. In: Proc. of the DSN'01, Goteborg, Sweden (2001) 433–442
4. Renesse, R.v., Birman, K., Vogels, W.: Astrolabe: A robust and scalable technology for distributed system monitoring, management, and data mining. ACM Trans. Comput. Syst. **21** (May 2003) 164–206
5. DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Sivasubramanian, S., Vosshall, P., Vogels, W.: Dynamo: Amazon's highly available key-value store. SIGOPS Oper. Syst. Rev. **41** (October 2007) 205–220
6. Lakshman, A., Malik, P.: Cassandra: A decentralized structured storage system. SIGOPS Oper. Syst. Rev. **44** (April 2010) 35–40
7. Renesse, R.v., Minsky, Y., Hayden, M.: A gossip-style failure detection service. Technical report, Ithaca, NY, USA (1998)
8. Ganesh, A.J., Kermarrec, A.M., Massoulié, L.: HiScamp: self-organizing hierarchical membership protocol. In: Proc. of the 10th ACM SIGOPS EW'10, Saint-Emilion, France (2002) 133–139
9. Gupta, I., Kermarrec, A.M., Ganesh, A.J.: Efficient and adaptive epidemic-style protocols for reliable and scalable multicast. IEEE Trans. Parallel Distrib. Syst. **17**(7) (July 2006) 593–605
10. Matos, M., Sousa, A., Pereira, J., Oliveira, R., Deliot, E., Murray, P.: CLON: Overlay networks and gossip protocols for cloud environments. In: Proc. of the DOA'09, Springer Verlag (2009) 549–566
11. Ganesh, A.J., Kermarrec, A.M., Massoulié, L.: SCAMP: Peer-to-peer lightweight membership service for large-scale group communication. In: Proc. of the NGC'01, Springer-Verlag (2001) 44–55
12. Benson, T., Akella, A., Maltz, D.A.: Network traffic characteristics of data centers in the wild. In: Proc. of the IMC'10, Melbourne, Australia (2010) 267–280
13. Voulgaris, S., Gavidia, D., Steen, M.v.: CYCLON: Inexpensive membership management for unstructured p2p overlays. Journal of Network and Systems Management **13** (2005) 2005
14. Eugster, P., Guerraoui, R., Kermarrec, A.M., Massoulie, L.: From epidemics to distributed computing. IEEE Computer **37**(5) (May 2004) 60 – 67
15. Montresor, A., Jelasity, M.: PeerSim: A scalable P2P simulator. In: Proc. of the P2P'09, Seattle, WA (2009) 99–100