Identificação de Canais Encobertos no Skype usando Esboços em SDNs

André Madeira, Diogo Barradas, Nuno Santos, and Luís Rodrigues
{andre.madeira, diogo.barradas, nuno.m.santos, ler}@tecnico.ulisboa.pt

INESC-ID, Instituto Superior Técnico, Universidade de Lisboa

Resumo A caracterização de fluxos de rede é relevante para múltiplas aplicações, em particular para aplicações de segurança, tal como a deteção de canais encobertos em tempo-real. Tipicamente, esta operação é realizada registando todos os pacotes dos fluxos relevantes e analisando as suas características, por exemplo, obtendo a distribuição dos seus tamanhos. No entanto, esta solução consome muitos recursos, afetando o desempenho da rede. Neste trabalho aferimos a possibilidade de explorar os avanços recentes nas redes SDN, nos comutadores programáveis, e nas estruturas de dados probabilísticas (também designadas por *esboços*, do Inglês, *sketches*) para caracterizar os fluxos no próprio comutador, à velocidade da linha, reduzindo assim a quantidade de dados de rede que têm de ser armazenados e analisados para identificar canais encobertos. Apresentamos uma arquitetura de software para comutadores programáveis que permite caracterizar os fluxos utilizando duas camadas de filtragem, cada uma recorrendo a um esboço. A nossa solução permite monitorizar 5K fluxos mantendo uma precisão de $0,95$ na deteção de fluxos encobertos, representando uma capacidade de análise 20 vezes maior para a mesma quantidade de memória no comutador na ausência de um esboço.

Keywords: Canais encobertos · Esboço · SDN

1 Introdução

A caracterização de fluxos de pacotes é necessária para diversas aplicações de segurança, tais como a deteção de canais encobertos [2], ou a criação de perfis de acesso [7]. Neste trabalho, abordamos o problema de detetar, em tempo real, canais encobertos em fluxos multimédia, nomeadamente, em chamadas Skype. Consideramos o cenário em que é possível ter acesso aos pacotes trocados por ligações Skype e pretendemos identificar quais destas são chamadas legítimas e quais são chamadas que transportam um canal encoberto. A informação do canal encoberto é tipicamente codificada no áudio e vídeo trocado pelo Skype através de ferramentas para comunicação resistente a censura como o Facet [9] ou o DeltaShaper [3]. Como o Skype cifra a informação multimédia, a presença do canal encoberto só pode ser detetada analisando algumas características dos pacotes, como o seu tamanho ou frequência. Neste artigo focamo-nos na deteção de canais encobertos com base na distribuição dos tamanhos dos pacotes da

chamada Skype. O nosso trabalho anterior [2] mostrou que a maioria das técnicas usadas para criar o canal encoberto é vulnerável a este tipo de análise.

Tipicamente, para analisar a distribuição dos pacotes de uma dada chamada Skype, criam-se cópias dos pacotes do fluxo correspondente num servidor dedicado que faz a análise das características dos mesmos. Uma vez que o número de chamadas Skype pode ser bastante elevado, este processo pode acarretar um elevado custo, nomeadamente na largura de banda necessária para recolher as cópias dos pacotes. Para além disso, a análise da distribuição dos tamanhos dos pacotes é feita em modo diferido, existindo uma latência significativa na deteção dos fluxos que transportam canais encobertos.

Neste trabalho aferimos a exequibilidade de explorar os avanços recentes nas redes de computadores para detetar canais encobertos com menor custo e menor latência. Em particular, pretendemos tirar partido dos comutadores programáveis, que conseguem realizar operações simples à velocidade da linha, para extrair uma distribuição aproximada do tamanho dos pacotes dos vários fluxos no próprio comutador sem necessitar de duplicar os pacotes. Uma vez que a quantidade de memória disponível nos comutadores é relativamente limitada, utilizamos estruturas de dados probabilísticas (também designadas por *esboços*, do Inglês, *sketches*) [4,5]. Estas estruturas usam a memória disponível de forma bastante eficiente, mas, em contrapartida, não permitem caracterizar a distribuição dos pacotes de forma completamente precisa. Um dos desafios deste trabalho é compreender como é que é possível aumentar o número de ligações Skype monitorizadas em simultâneo, com a memória limitada que existe no comutador, obtendo uma precisão satisfatória na classificação dos fluxos.

A nossa solução baseia-se na utilização de uma variante do esboço *Count-Min* [5]. Uma avaliação experimental da nossa solução, mostra que esta consegue aumentar de forma significativa o número de fluxos Skype que é possível monitorizar. Considere-se, por exemplo, que o comutador possui $0,3$MB de memória disponível. Com esta memória apenas é possível monitorizar cerca de 250 canais sem qualquer perda de precisão. Usando a nossa solução, é possível monitorizar 5K fluxos mantendo uma precisão de $0,95$ na deteção de fluxos encobertos, apesar do erro introduzido pela utilização do esboço. A nossa solução permite a análise de 20 vezes mais fluxos para a mesma quantidade de memória disponível.

2 Contexto e Trabalho Relacionado

Esta secção introduz os conceitos e as tecnologias utilizadas no desenvolvimento do nosso sistema, nomeadamente: redes definidas por software (do Inglês, *software-defined networks*, ou SDNs), comutadores programáveis, deteção de canais encobertos e esboços.

SDN A arquitetura de redes definidas por software introduz uma separação clara entre o que se designa por plano de dados, responsável por transportar os pacotes na rede, e o plano de controlo, responsável por decidir quais os pacotes que devem ser transportados e quais os caminhos que estes devem seguir. O plano de dados é concretizado por equipamentos de rede, tipicamente designados por

comutadores, que se limitam a despachar os pacotes que são recebidos por um porto de entrada para um ou mais portos de saída, através de uma tabela de despacho que toma partido de campos específicos do pacote como o IP de origem, por exemplo. Normalmente, estas tabelas agrupam pacotes em fluxos para os identificar mais facilmente. Sequências de pacotes transmitidos entre os mesmos dois computadores e que possuam os mesmos campos usados pelo comutador para os diferenciar, pertencem ao mesmo fluxo. As tabelas de despacho podem ser configuradas remotamente, através de uma interface normalizada, por um controlador centralizado. Desta forma, as políticas de encaminhamento podem ser expressas, de forma programática, e executadas pelo controlador. Na prática, este controlador é um programa normalmente executado numa máquina dedicada. Os comutadores podem também recolher algumas estatísticas sobre o seu funcionamento que podem ser lidas usando a mesma interface. Esta arquitetura tem-se afirmado por facilitar a configuração e a monitorização das redes [8].

Comutadores Programáveis Numa arquitetura de redes IP clássica, o programa que é executado por um comutador, ao despachar um pacote, é definido pelo fabricante e não pode ser alterado pelo utilizador. Esta restrição deve-se não apenas a políticas comerciais mas sobretudo à necessidade de garantir um elevado débito na comutação de pacotes. A evolução da tecnologia mostrou que é possível desenvolver comutadores que conseguem executar pequenos programas à velocidade da linha, sem comprometer o débito do sistema [1]. Estes comutadores permitem ao utilizador definir programas a executar e permitem que estes programas sejam carregados dinamicamente. Isto possibilita, por exemplo, instalar no comutador programas que executam funções de monitorização dos fluxos desenhadas à medida das necessidades do operador da rede.

Deteção de Canais Encobertos em Chamadas Multimédia Um estudo recente [2] mostrou que as diferentes técnicas que são hoje conhecidas para estabelecer canais encobertos em fluxos multimédia alteram a distribuição do tamanho dos pacotes, em relação à distribuição observada numa chamada legítima. Estas diferenças podem ser detetadas de forma automática recorrendo a técnicas de classificação supervisionada baseadas em árvores de decisão, tais como Florestas Aleatórias ou o algoritmo *XGBoost*. Isto faz com que seja possível executar a deteção dos fluxos que encobrem canais através da recolha da distribuição do tamanho dos pacotes. Caso seja possível capturar esta distribuição à velocidade da linha, usando por exemplo as capacidades oferecidas pelos comutadores programáveis, torna-se possível detetar canais encobertos em tempo real.

Esboços Para grandes volumes de dados, o cálculo dos valores exatos das distribuições de tráfego é computacionalmente dispendioso e requer grandes quantidades de memória. A complexidade desta tarefa motivou o desenvolvimento de esboços, estruturas de dados que permitem o cálculo do valor aproximado de métricas com base num número vasto de amostras. Tipicamente, os esboços recorrem à utilização de funções de dispersão para associar rapidamente os valores lidos a registos, permitindo utilizar um número de registos significativamente inferior aos que seria necessário para obter o valor exato. No entanto, a utilização de funções de dispersão acarreta a possibilidade de ocorrerem colisões, em que

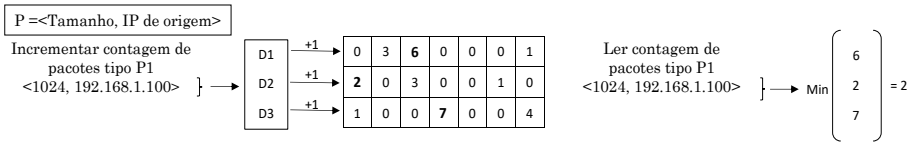


Figura 1. Representação do esboço Count-Min.

entidades diferentes são associadas a um mesmo registo, introduzindo um erro na estimativa obtida. Este erro pode ser controlado variando o número de registos e o número de funções de dispersão usadas na concretização do esboço. A literatura é rica em diferentes tipos de esboços [4,5], sendo que recentemente várias destas estruturas têm sido propostas com o objetivo específico de facilitar a monitorização de redes de computadores [10].

Esboço *Count-Min* (CM) O *Count-Min Sketch* [5] é um esboço concebido para realizar contagens utilizando a memória de forma eficiente. O esboço armazena as contagens numa matriz bidimensional de registos, sendo que cada linha da matriz se encontra associada a uma função de dispersão. A Figura 1 ilustra o funcionamento de um esboço CM com três funções de dispersão (D1, D2 e D3) ao processar um pacote identificado pelo tuplo P = <*Tamanho*, *IP de origem*>. Ao receber o pacote P1 = <`1024`, `192.168.1.100`>, o esboço agrupa os valores de P1 e utiliza cada função de dispersão para selecionar os registos a incrementar. Intuitivamente, sabendo o espaço e quantidade dos identificadores dos itens a registar, é possível configurar o número de linhas e colunas da matriz para que a probabilidade de existirem colisões em todas as linhas seja baixa (este valor é um parâmetro de configuração do esboço). Finalmente, é possível ler a contagem de pacotes P1 registados pelo esboço ao agrupar os valores pertencentes ao tuplo, reunindo os valores dos registos correspondentes a cada função de dispersão. O esboço devolve uma aproximação da contagem real de pacotes P1 ao escolher o menor valor obtido a partir de cada linha. O racional para esta decisão centra-se na observação de que o valor de cada linha nunca retorna uma contagem inferior ao valor real do item e que valores superiores resultam da ocorrência de colisões.

3 Arquitetura

O nosso sistema, ilustrado na Figura 2, faz uso de comutadores programáveis e de duas variações do esboço CM para criar uma linha de execução que recolhe várias métricas a partir dos fluxos encontrados na rede, permitindo a sua posterior diferenciação. Para cada fluxo, o sistema armazena uma representação da distribuição do tamanho de pacotes, obtida através da quantificação do tamanho de cada pacote numa escala discreta. Designamos por "banda" cada faixa de valores nesta escala (por exemplo, os pacotes de tamanho compreendido entre 10 e 20 octetos). Quando os pacotes de um fluxo são processados pelos comutadores, estes utilizam o esboço para calcular e armazenar localmente as estimativas da distribuição do fluxo. Posteriormente, o controlador pode requisitar a leitura da

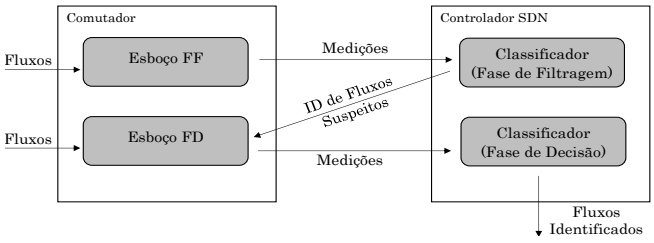


Figura 2. Arquitetura do sistema.

estimativa da distribuição de pacotes de um dado fluxo ao esboço armazenado no comutador, calculando de seguida várias métricas estatísticas com base nesta distribuição. Como analisado na Secção 4, a utilização destas métricas permite a instanciação de um mecanismo de diferenciação de tráfego capaz de detetar a existência de canais encobertos em fluxos multimédia.

Nas próximas secções, explicitamos a intuição que motivou o desenvolvimento de diferentes variações do esboço CM (Secção 3.1), e descrevemos as diferentes etapas envolvidas na análise e diferenciação de fluxos (Secção 3.2).

3.1 Variações do Esboço *Count-Min*

Nesta secção começamos por descrever uma abordagem para a recolha de métricas de fluxos que não toma por base o uso de um esboço e que, portanto, é expectável que consuma uma elevada quantidade de memória do comutador. De seguida, introduzimos diversas variações do esboço CM, descrevendo as vantagens e limitações de cada um na recolha precisa de métricas. O sucesso obtido por cada esboço em caracterizar e diferenciar diferentes classes de fluxos é analisado através do estudo experimental apresentado na Secção 4.

Ausência de Esboço (No-CM) Na ausência de um esboço, o comutador mantém um registo para cada banda de tamanho de pacotes pertencente a um dado fluxo. Para acomodar a recolha de um elevado número de fluxos, é possível representar a distribuição do tamanho de pacotes de cada fluxo de forma sumária ao comprimir esta distribuição através do aumento do intervalo de cada banda. No entanto, é possível que a utilização deste método possa resultar na representação pouco precisa da distribuição do tamanho de pacotes de cada fluxo.

Esboço CM Simples (CM-S) O esboço CM-S é constituído por um único esboço CM, cujo funcionamento foi descrito na Secção 2. Quando um pacote é recebido pelo comutador, este incrementa o registo no esboço correspondente à aplicação da função de dispersão sobre um tuplo que inclui: a) o valor da banda a que o tamanho do pacote corresponde e b) o identificador do fluxo. Recordamos o leitor de que o identificador de um fluxo é composto por um conjunto de campos comuns a todos os pacotes pertencentes ao fluxo, tais como o IP de origem/destino e o porto utilizado. No esboço CM-S, a contabilização da frequência de pacotes numa dada banda que pertencem a um fluxo pode

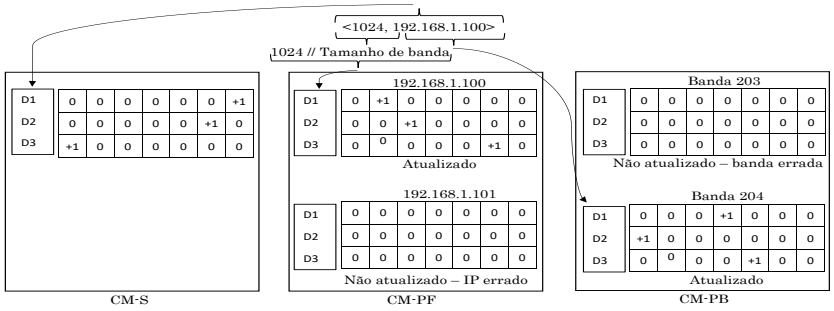


Figura 3. Exemplo da atualização de cada esboço aquando da receção de um pacote.

ser contaminada por pacotes (nessa ou noutra banda) pertencentes a qualquer outro fluxo. Como cada fluxo possui um identificador diferente, o tuplo resultante usado como entrada para a função de dispersão é sempre diferente, o que pode reduzir as colisões para valores que permitam ainda diferenciar os fluxos a partir da estimativa da distribuição que resulta da utilização do esboço. A Figura 3 ilustra a contabilização de um pacote identificado pelo tuplo P1 = <`1024`, `192.168.1.100`> no esboço CM-S (à esquerda).

Esboço CM por Fluxo (CM-PF) O esboço CM-PF é composto por um esboço CM para cada fluxo identificado pelo comutador. Quando um pacote é recebido pelo comutador, este verifica qual o fluxo correspondente, associando-o a uma entrada no esboço determinada por uma função de dispersão. Se esta entrada já se encontrar associada a outro fluxo, o novo fluxo não é registado. O registo a incrementar é selecionado com base na aplicação da função de dispersão ao identificador da banda correspondente ao tamanho do pacote. O esboço CM-PF assenta na observação de que as colisões nos registos de cada banda ocorrerão de forma igual para cada fluxo existente (uma vez que todos os esboços CM são configurados com as mesmas funções de dispersão). Assim, ao comprimir uma distribuição de pacotes num conjunto limitado de bandas, é expectável que diferentes classes de fluxos façam emergir uma assinatura com informação suficiente para diferenciar as várias classes. A Figura 3 ilustra a contabilização de um pacote no esboço CM-PF (ao centro). Apenas a primeira entrada do esboço é atualizada pois esta corresponde ao fluxo a que o pacote recebido pertence.

Esboço CM por Banda (CM-PB) O esboço CM-PB é composto por um esboço CM para cada banda de tamanho de pacotes. Cada esboço ocupa a mesma memória. Quando um pacote é recebido pelo comutador, o esboço CM-PB incrementa os registos correspondente à aplicação da função de dispersão sobre o identificador do fluxo no esboço CM que diz respeito à banda do tamanho do pacote em causa. No esboço CM-PB, todos os fluxos que colidem num dos esboços irão colidir em todos os esboços. No entanto, se os fluxos que colidem tiverem distribuições semelhantes, estas colisões não afetam a distribuição resultante. Pelo contrário, se um dos fluxos que colidir apresentar uma distribuição

distinta, este poderá ainda ser detetado, o que é desejável no contexto deste trabalho. A Figura 3 ilustra a contabilização de um pacote recebido no esboço CM-PB (à direita). Apenas a segunda entrada do esboço é atualizada uma vez que a banda a que este esboço pertence corresponde ao tamanho do pacote.

3.2 Recolha e Diferenciação de Fluxos

Propomos uma arquitetura com duas fases, usando dois esboços em sequência, para identificar fluxos que transportam canais encobertos. A primeira fase é denominada *Fase de Filtragem*, sendo o respetivo esboço denominado *FF*. A segunda fase é denominada *Fase de Decisão* e o esboço respetivo denominado *FD*. Na secção de avaliação, iremos aferir a eficácia desta arquitetura e qual o tipo de esboço mais indicado para cada fase.

Nesta arquitetura, um conjunto de fluxos a classificar é processado em primeiro lugar pelo esboço FF, de forma a obter uma estimativa (possivelmente pouco precisa) da distribuição dos pacotes de cada fluxo. Periodicamente, as distribuições obtidas são enviadas para um classificador, que identifica o subconjunto destes fluxos que são considerados suspeitos. Os identificadores dos fluxos suspeitos são então reportados ao comutador, que coleciona tráfego adicional dos mesmos através do esboço FD. Adicionalmente, a primeira fase de filtragem contribuí para a suavização da proporção inicialmente enviesada entre fluxos legítimos e fluxos que transportam canais encobertos. Uma vez que o número de fluxos a observar na segunda fase é significativamente menor que os fluxos iniciais, espera-se que a utilização do esboço FD obtenha estimativas mais precisas da distribuição de cada um destes fluxos. As estimativas obtidas são de novo enviadas para classificação, que produz o resultado final da filtragem.

4 Avaliação

A avaliação do nosso sistema evidencia a necessidade da existência de uma arquitetura que contém duas fases de filtragem para executar a distinção de fluxos, respeitando duas principais dimensões: o número de fluxos que é possível caracterizar e a precisão da diferenciação associada a esses fluxos.

A Secção 4.1 descreve a configuração da bancada experimental, incluindo as métricas e conjuntos de dados utilizados no decorrer das nossas experiências. Na Secção 4.2 é analisado o sucesso obtido por cada esboço anteriormente descrito na diferenciação de fluxos. Na Secção 4.3, avaliamos o sucesso do processo de diferenciação de fluxos ao utilizar uma combinação de esboços para o efeito.

4.1 Configuração das Experiências

Configuração dos esboços Simulamos a utilização de um comutador programável para a instanciação de esboços conducentes à recolha de uma representação da distribuição do tamanho de pacotes respeitantes a um número limitado de fluxos. Cada esboço utiliza três funções de dispersão que correspondem à aplicação de uma função de dispersão com um valor de inicialização distinto.

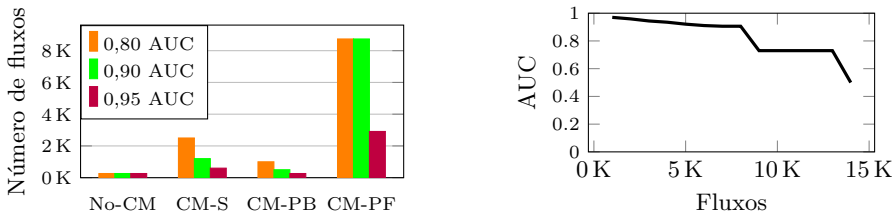
A memória útil do comutador é fixada num total de 0,3MB, escolhida tendo em conta a base teórica subjacente ao desenho do esboço CM. Com 0,3MB de memória, é garantido com 95% de probabilidade que o valor da contagem lido a partir de cada registo, numa dada banda do esboço CM, incorre num erro inferior a 10% do total de pacotes da verdadeira distribuição de um fluxo. No decorrer da nossa avaliação experimental, os tamanhos dos pacotes de cada fluxo são quantificados em bandas de $k = 5$ octetos. Considerando a possível existência de pacotes com um tamanho compreendido entre 0 a 1500 octetos, utilizamos um total de 300 bandas. Adicionalmente, utilizamos o IP de origem (atribuído aleatoriamente) para atribuir um identificador a cada fluxo.

Conjunto de dados As nossas experiências contemplam a transmissão de fluxos multimédia numa proporção de 95% de fluxos Skype legítimos e 5% de fluxos Skype que transportam um canal encoberto. Este rácio contempla a simulação de circunstâncias reais, onde é expectável que a maioria dos fluxos observados na rede não atuem como veículo de um canal encoberto. Relegamos testes com diferentes proporções do tráfego para trabalho futuro. Os fluxos que transportam um canal encoberto foram produzidos pelo sistema Facet [9], que substitui uma região das tramas de vídeo produzidas por uma video-chamada legítima por outros conteúdos visuais, tais como vídeos YouTube. Cada fluxo tem a duração total de 60s. Todos os fluxos utilizados resultam do estudo já mencionado [2].

Métricas Trabalhos anteriores [2,3,9] utilizam a métrica ROC AUC [6] para medir o sucesso obtido na diferenciação de fluxos. De forma breve, a AUC sumariza a relação entre a taxa de verdadeiros positivos e falsos positivos de um classificador. Um classificador com a capacidade de emitir um palpite aleatório sobre a classe de um fluxo exibe uma AUC = 0,5, ao passo que um classificador perfeito exibe uma AUC = 1. Tal como proposto no nosso trabalho anterior [2], utilizamos a AUC e o algoritmo de classificação supervisionada *XGBoost* para avaliar a precisão dos diferentes tipos de esboços. O classificador é treinado utilizando um conjunto de fluxos, recolhido em rondas. Em cada ronda, selecionamos uma sub-amostragem de fluxos na proporção 95%/5%. Depois, é obtida uma representação aproximada da distribuição destes fluxos através do esboço sob análise. O classificador é treinado com um conjunto de amostras balanceado, isto é, 50%/50%, obtidas a partir das representações das distribuições geradas pelo esboço. Este processo é repetido 200 vezes.

4.2 Usando um Único Esboço

Começamos por comparar o desempenho das diferentes variantes do esboço CM em relação a uma solução que não recorre a esboços para o caso em que se usa toda a memória disponível com um único esboço. Neste caso, foram recolhidos pelo esboço os primeiros 30s de tráfego correspondente a cada fluxo. Nesta experiência, pretendemos compreender qual o número de fluxos que podem ser analisados pelos diferentes tipos de esboços ao passo que se obtém uma precisão de classificação acima de três diferentes patamares de AUC (0,80, 0,90, e 0,95).



(a) Capacidade de análise de fluxos por esboço - Memória = 0,3MB.

(b) Relação entre o número de fluxos e a AUC obtida pelo esboço CM-PF.

Figura 4. Capacidade de análise de fluxos por cada variação do esboço CM.

Os resultados apresentados na Figura 4a mostram que não recorrendo ao uso de esboços, e usando um número fixo de registos por cada fluxo (neste caso, 300, um por cada banda, cada um dos quais ocupando 4 octetos de memória) apenas é possível medir 262 fluxos. O esboço CM-S mede mais fluxos para todos os patamares de AUC, indicando que apesar das diversas colisões entre fluxos e bandas ainda é possível distinguir os dois tipos de tráfego com precisão elevada.

O esboço CM-PB exibe valores inferiores ao CM-S, o que se deve ao facto do CM-PB não fazer uso eficiente da memória disponibilizada, uma vez que as diferentes bandas não são utilizadas de forma uniforme pelos diferentes fluxos. Este fenómeno pode ser observado na Figura 5, que ilustra o mapa de calor das bandas 13 e 100. Cada linha corresponde à aplicação de uma função de dispersão distinta (como previamente ilustrado na Figura 3) e a divisão da memória disponibiliza 87 registos por cada função de dispersão para cada banda. De acordo com o mapa de calor, este número de registos não é suficiente para evitar um elevado número de colisões à medida que o número de fluxos contabilizado pelo esboço é aumentado. Constata-se então que, para os fluxos sob análise, existem bandas com uma elevada contagem de pacotes ao passo que outras apresentam uma baixa contagem. A redistribuição do espaço reservado para bandas pouco utilizadas para reduzir o número de colisões nas restantes bandas revela-se um processo moroso que necessita de uma afinação manual do esboço.

Finalmente, o esboço CM-PF obtém os melhores resultados. Isto deve-se ao facto de cada fluxo ser mantido separadamente de todos os outros, prevenindo a colisão entre fluxos Skype legítimos e fluxos que transportam um canal encoberto. Adicionalmente, os dois tipos de fluxos são facilmente distinguíveis com base na distribuição do tamanho de pacotes, e estas diferenças continuam visíveis mesmo quando várias bandas colidem no mesmo registo. É de notar que o CM-PF suporta o mesmo número de fluxos para AUC 0,8 e 0,9 uma vez que a compressão efetuada a nível dos registos não é significativa.

Limite do CM-PF Ao usar o CM-PF para medir um número elevado de fluxos é necessário reduzir o tamanho de cada esboço CM, levando a que, a partir de um certo número de fluxos, o número de colisões entre bandas distintas dita a perda da capacidade de distinguir o tráfego. Com efeito, os resultados representados na Figura 4b mostram que o esboço CM-PF atinge uma AUC compreendida entre 1

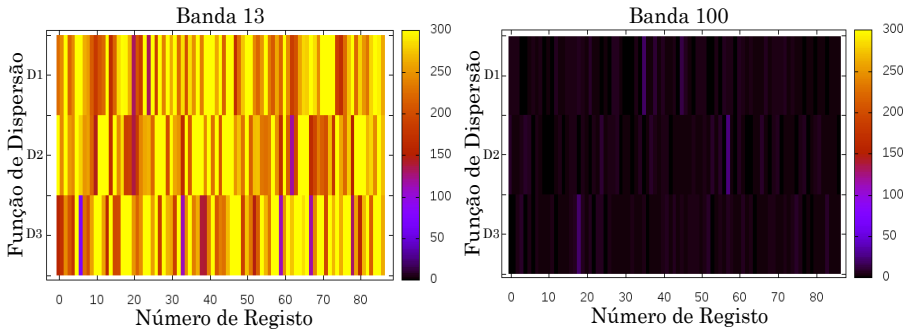


Figura 5. Mapa de calor das bandas 13 e 100 do esboço CM-PB.

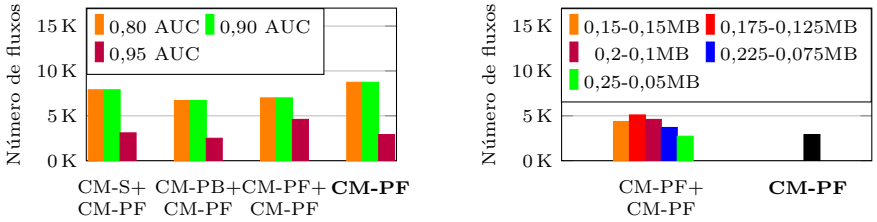
e 0,9 para um número de fluxos inferior a 8K, decaindo para 0,7 até 13K fluxos. Neste ponto, é observado um decréscimo acentuado da AUC, aproximando-se esta do valor 0,5, e, portanto, reduzindo a decisão da classificação a um palpite aleatório para um número de fluxos superior a 14K. Em suma, a figura evidencia que à medida que o tamanho de cada esboço CM diminui, cada contador que o compõe vai ser partilhado por um maior número bandas. Desta forma, os valores correspondentes à leitura destas bandas tenderão a ser partilhados. Naturalmente, quantas mais bandas partilharem os mesmos valores, maior será o grau de sobreposição das aproximações das distribuições de diferentes fluxos.

4.3 Combinando Dois Esboços

Nesta secção aferimos se a arquitetura proposta, que se baseia na utilização de dois esboços em sequência, oferece vantagens em relação à utilização de um único esboço. Ao invés de experimentar todas as combinações possíveis de diferentes esboços, optámos por usar sempre o esboço que apresentou melhores resultados na Fase de Decisão (isto é, usar o CM-PF como esboço FD) e variar apenas o esboço usado na Fase de Filtragem. Para simular a diferenciação de fluxos em tempo real, a Fase de Filtragem é alimentada com os primeiros 30s de tráfego de cada fluxo, enquanto que a Fase de Decisão é alimentada com os restantes 30s.

Na Figura 6a apresentamos os resultados para uma configuração em que 0,2MB são reservados para o esboço FF e 0,1MB são usados para o esboço FD (apresentaremos resultados adicionais variando esta proporção na Secção 4.4.). Note-se que, nestes testes, só o esboço FD mantém a AUC indicada; a primeira fase pode admitir um AUC menor visto que os fluxos selecionados para a segunda fase serão alvo de uma nova análise.

Como vimos anteriormente, o CM-PF consegue recolher cerca de 9K fluxos enquanto mantém as AUCs alvo de 0,8 e 0,9, usando 0,3MB de memória. Isto significa que o esboço CM-PF deverá conseguir classificar 3K fluxos quando usado com 0,1MB na Fase de Decisão. Desta forma, para ser competitiva, a arquitetura em duas fases terá que filtrar mais do que 2/3 dos fluxos na Fase de Filtragem. Os valores mostram que as variantes do esboço CM não oferecem uma



(a) Memória = 0,2MB (FF) + 0,1MB (FD). (b) Memória Variável (FF + FD).

Figura 6. Capacidade da arquitetura de duas camadas na análise de fluxos.

capacidade de filtragem competitiva para estes valores de AUC. Por exemplo, o número máximo de fluxos da arquitetura em duas camadas é, para AUC = 0,8, sempre inferior ao de uma arquitetura usando um único esboço CM-PF. Por outro lado, quando se pretendem atingir valores de 0,95 de AUC, o CM-PF só consegue suportar a análise de 3K fluxos (1K quando usado com 0,1MB na Fase de Decisão). Neste caso, a arquitetura em duas camadas oferece vantagens tangíveis. De facto, usando uma combinação do esboço CM-PF em duas camadas é possível aumentar o número de fluxos que se conseguem monitorizar de 3K para cerca de 4.5K, o que representa um aumento na ordem de 50%.

4.4 Variando a Memória do Filtro

Nesta secção, analisamos de que forma a memória disponível para as diferentes fases de análise afeta a capacidade que o sistema exibe na diferenciação de fluxos. Mais concretamente, comparamos diferentes configurações da arquitetura em duas fases, fazendo variar a proporção de memória reservada para cada um dos esboços, nomeadamente, fixando a memória disponível para FF, entre 0,15MB e 0,25MB, em aumentos sucessivos de 25KB. Nesta experiência, utilizamos a melhor combinação de esboços previamente identificada (CM-PF + CM-PF).

Como se pode observar na Figura 6b, diferentes proporções apresentam resultados distintos, sendo que a memória reservada para a FD necessita de ser suficiente para classificar com precisão todos os fluxos que não são filtrados na FF. Existe pois um equilíbrio entre a capacidade de filtragem da primeira fase e a precisão da segunda fase. Para as configurações em estudo, a proporção que oferece melhores resultados consiste em reservar 0,175MB para a FF e 0,125MB para a FD. Com esta configuração é possível monitorizar cerca de 5K fluxos, um ganho de cerca de 66% em relação à utilização de um único esboço.

5 Conclusões

Neste trabalho estudámos a possibilidade de classificar canais encobertos em tempo real e de forma eficiente, tomando partido da utilização de comutadores programáveis e redes definidas por software para capturar uma representação

aproximada da distribuição dos pacotes dos fluxos a monitorizar. De forma a capturar uma aproximação da distribuição de forma eficiente, recorremos a estruturas de dados probabilísticas conhecidas por esboços. Neste contexto, propomos uma arquitetura com duas camadas de filtragem, em que numa primeira fase um esboço permite filtrar uma fração significativa dos fluxos, sendo os restantes fluxos classificados recorrendo a um esboço configurado para obter melhor precisão. Esta arquitetura apresenta ganhos superiores a 66% em relação à utilização de um único esboço, sendo capaz de monitorizar cerca de 5K fluxos simultaneamente e oferecer uma AUC de 0,95 na identificação de canais encobertos, usando para este efeito apenas 0,3MB de memória no comutador. Em trabalho futuro, planeamos expandir a análise quantitativa da nossa solução através da exploração de diferentes proporções entre o tráfego legítimo e encoberto existente na rede.

Agradecimentos: Este trabalho foi suportado pela FCT – Fundação para a Ciência e a Tecnologia, através dos projectos UID/CEC/50021/2019 e COSMOS (financiado pelo OE com a ref. PTDC/EEI-COM/29271/2017 e pelo Programa Operacional Regional de Lisboa na sua componente FEDER com a ref. Lisboa-01-0145-FEDER-029271).

Referências

1. Barefoot: https://www.barefootnetworks.com/products/brief-tofino/, accessed: 2019-06-12
2. Barradas, D., Santos, N., Rodrigues, L.: Effective detection of multimedia protocol tunneling using machine learning. Proceedings of the 27th USENIX Security Symposium pp. 169–185 (August 2018)
3. Barradas, D., Santos, N., Rodrigues, L.: Deltashaper: Enabling unobservable censorship-resistant tcp tunneling over videoconferencing streams. In: Proceedings on Privacy Enhancing Technologies. vol. 2017(4), pp. 5–22. Minneapolis, MN, USA (2017)
4. Charikar, M., Chen, K., Farach-Colton, M.: Finding frequent items in data streams. ICALP '02 Proceedings of the 29th International Colloquium on Automata, Languages and Programming pp. 693–703 (July 2002)
5. Cormode, G., Muthukrishnan, S.: An improved data stream summary: The count-min sketch and its applications. Journal of Algorithms **55**(1), 58–75 (April 2005)
6. Fawcett, T.: Roc graphs: Notes and practical considerations for data mining researchers. ReCALL **31**, 1–38 (01 2004)
7. Hayes, J., Danezis, G.: k-fingerprinting: A robust scalable website fingerprinting technique. In: 25th USENIX Security Symposium. pp. 1187–1203. Austin, Texas, USA (August 2016)
8. Kreutz, D., Ramos, F.M.V., Veríssimo, P.E., Rothenberg, C.E., S., A., Uhlig, S.: Software-defined networking: A comprehensive survey. Proceedings of the IEEE **103**(1), 14–76 (Jan 2015)
9. Li, S., Schliep, M., Hopper, N.: Facet: Streaming over videoconferencing for censorship circumvention. In: Proceedings of the 13th Workshop on Privacy in the Electronic Society. pp. 163–172. Scottsdale, AZ, USA (2014)
10. Yang, T., Jiang, J., Liu, P., Huang, Q., Gong, J., Zhou, Y., Miao, R., Li, X., Uhlig, S.: Elastic sketch: Adaptive and fast network-wide measurements. SIGCOMM '18 Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication pp. 561–575 (August 2018)