Mitigando Ataques DDoS em SGIs por Reorganizações em Agrupamentos de IdP

Ricardo Macedo[1], Leonardo Melniski[1], Aldri Santos[1], Yacine Ghamri-Doudane[2], Michele Nogueira[1]

[1]Universidade Federal do Paraná – Curitiba – PR – Brasil

[2]University of La Rochelle – La Rochelle CEDEX 1 – France

{rtmacedo,lem09,aldri,michele}@inf.ufpr.br, yacine.ghamri@univ-lr.fr

Abstract. *Identity management (IdM) systems employ Identity Providers (IdPs), as guardians of users' critical information. However, Distributed Denial-of-Service (DDoS) attacks can make IdPs operations unavailable, compromising legitimate IdM system users. This work presents SAMOS, a novel schema to mitigate DDoS attacks in IdM systems through a novel approach: Organizations of IdP clustering using optimization techniques. SAMOS is started based on the monitoring of processing and memory resources, differently from solutions in the literature that are started based on DDoS detection through the network traffic analysis. SAMOS minimizes DDoS effects using the system operational IdPs, differentiating from proposes that employ external computer resources. Results considering data from a real IdM systems indicate the scheme viability.*

Resumo. *Os sistemas de gerenciamento de identidades (SGI) utilizam Identity Providers (IdP) como guardiões das informações críticas dos usuários. No entanto, ataques DDoS (Distributed Denial-of-Service) podem indisponibilizar as operações prestadas por IdPs, prejudicando usuários legítimos. Este trabalho apresenta SAMOS, um esquema para mitigar ataques DDoS em SGIs através de uma abordagem inovadora: a reorganização de agrupamentos de IdPs usando técnicas de otimização. SAMOS atua em resposta ao monitoramento dos recursos de memória e processamento dos IdPs, diferentemente das soluções da literatura que agem em resposta à detecção do ataque através da análise do tráfego de rede. SAMOS minimiza os efeitos dos ataques DDoS empregando os IdPs operacionais do sistema, se distinguindo das propostas existentes que utilizam recursos computacionais externos. Resultados considerando dados de um SGI real indicam a viabilidade da proposta.*

1. Introdução

Os Sistemas de Gerenciamento de Identidades (SGI) vêm recebendo atenção da academia e da indústria devido ao seu potencial em integrar diferentes domínios administrativos, preservando tecnologias e políticas locais [Torres et al. 2013]. A principal vantagem destes sistemas consiste em empregar autoridades de autenticação denominadas *Identity Providers* (IdP) como guardiões das informações críticas dos usuários, separando o provimento de recursos (papel desempenhado por SPs - *Services Providers*) do gerenciamento dos dados críticos dos usuários [Barreto et al. 2013]. Através dos SGIs, a autenticação de usuários em um único domínio possibilita o acesso a múltiplos domínios,

reduzindo a complexidade do gerenciamento e incrementando a experiência dos usuários [Arias Cabarcos et al. 2014].

No entanto, IdPs são disponibilizados na Internet, tornando-se propensos à ataques de negação de serviço distribuídos (DDoS, do inglês *Distributed Denial of Service*) [Lonea et al. 2013]. Os IdPs possuem recursos de processamento e memória limitados, podendo ser esgotados frente a uma grande quantidade de requisições. Os ataques DDoS têm como objetivo indisponibilizar as operações de um sistema ao disparar um grande volume de requisições maliciosas, forçando o sistema vítima a consumir todos os recursos de memória e processamento [Carlson 2014]. Casos reais de DDoS foram relatados mesmo em ambientes ricos em recursos computacionais, como por exemplo em nuvens computacionais [Lonea et al. 2013, Shah et al. 2013], enfatizando a periculosidade dos mesmos. Em SGIs, os ataques DDoS podem resultar na indisponibilidade das operações de autenticação de usuários legítimos e congestionar o tráfego de dados para IdPs, impactando indiretamente no provimento de serviços. Neste contexto, as abordagens de mitigação tornam-se promissoras para minimizar os efeitos desses ataques [Fu et al. 2012].

Este trabalho apresenta SAMOS, do inglês **S***cheme for DDoS* **A***ttacks* **M***itigation by the re***O***rganization and optimization of the Identity management* **S***ystem*, a primeira proposta para mitigar os efeitos de ataques DDoS em SGIs através da reorganização de agrupamentos de IdPs. SAMOS atua em resposta ao monitoramento dos recursos de memória e processamento dos IdPs, diferentemente das soluções da literatura que agem em resposta à detecção do ataque através da análise do tráfego de rede. SAMOS minimiza os efeitos dos ataques DDoS empregando os IdPs operacionais do sistema, se distinguindo das propostas existentes que utilizam recursos computacionais externos. As reorganizações são realizadas através de três procedimentos: *Agrupamento*, *Pré-Configuração*, *Otimização*. O primeiro recruta o maior número de agrupamentos de IdPs capazes de suportar a sobrecarga gerada pelo ataque DDoS através de um algoritmo genético. O segundo computa todos os benefícios possíveis em balancear a carga das identidades e SPs entre os IdPs de cada agrupamento. O terceiro utiliza técnicas de otimização para encontrar o benefício máximo de balanceamento de carga. As reorganizações proporcionam a otimização do uso dos recursos computacionais do sistema, minimizando os efeitos do ataque e prolongando o tempo de vida do SGI.

O desempenho do SAMOS é avaliado por simulações baseadas em traços reais. Com base em traços do SGI da Universidade de *Buffalo*, simulações são realizadas sob ataques DDoS em IdPs. Duas análises são conduzidas, a primeira verifica a eficiência do algoritmo genético para formar agrupamentos de IdP e a segunda avalia o tempo alcançado para encontrar uma reorganização ótima dos IdPs do SGI durante um ataque DoS. Os resultados revelam que a taxa de soluções ótimas encontradas pelo algoritmo genético igual a 31,37% com maior tempo de execução inferior a 200 ms e que para um número equivalente a seis IdPs operacionais, o maior tempo de execução para encontrar uma solução para o esquema SAMOS foi inferior a 0,75 segundos, mostrando indícios da viabilidade desta proposta como uma alternativa para mitigar dos efeitos de ataques DDoS em SGIs.

O trabalho está organizado como segue. A Seção 2 descreve os trabalhos relacionados. A Seção 3 detalha o modelo de sistema e o modelo de falhas. A Seção 4 apresenta

o esquema SAMOS. A Seção 5 descreve a avaliação baseada em traços da proposta. A Seção 6 apresenta as conclusões.

2. Trabalhos Relacionados

Na literatura, existem iniciativas para mitigar ataques DoS no contexto de nuvens computacionais, Content-Centric Networking (CCN) e Software Defined Networking (SDN). No contexto de nuvens computacionais, foi apresentado um mecanismo para migrar os serviços vítimas de ataques DDoS, contando com o auxílio de replicações [Jia et al. 2014]. Este mecanismo usa métodos de otimização para encontrar uma alternativa para separar servidores vulneráveis de atacantes. Um estudo de caso envolvendo a nuvem *Amazon* EC2 demonstra a capacidade de mitigar ataques DDoS em larga escala.

Com o objetivo de detectar e mitigar ataques DoS em CCN foi proposto o framework Poseidon [Compagno et al. 2013]. A detecção baseia-se em anomalias, podendo ser local ou distribuída, e a mitigação ocorre através da limitação do uso dos recursos. A ocorrência de anomalias no tráfego de redes ocasionadas por ataques DoS também impulsionou a proposição de um mecanismo para detectar e mitigar ataques DoS em arquiteturas SDN [Giotis et al. 2014]. Essa solução realiza a detecção e mitigação de ataques DoS com o suporte de três módulos: coleta de dados, detecção de anomalias e mitigação das anomalias através do descarte do tráfego.

Outros trabalhos investigaram a segurança e a resiliência das operações de IdPs em SGIs. Barreto *et al.* apresentaram uma infraestrutura de gerenciamento de identidades tolerante a intrusões [Barreto et al. 2013]. Kreutz *et al.* propuseram uma arquitetura para tornar os IdPs resilientes a falhas arbitrárias através de técnicas de replicação ativa [Kreutz et al. 2014]. Dentre estes, [Kreutz et al. 2014] argumentaram a capacidade dos IdPs de suportar um número pré-determinado de falhas usando técnicas de replicação. Or arcabouços para gerenciamento de identidades tais como *Shibboleth* implementam a funcionalidade de agrupamento de IdPs[1], proporcionando a disponibilidade da sessão e a versão mais atual dos atributos dos usuários pertencentes aos IdPs falhos através da migração de dados entre IdPs.

De modo geral, essas propostas utilizam recursos computacionais externos para replicação e dependem da técnica detecção de ataques através do tráfego de rede. Iniciativas que realizam a mitigação com base na detecção do ataque através da análise do tráfego de rede, tal como em [Barreto et al. 2013], [Compagno et al. 2013] e [Giotis et al. 2014], estão sujeitas a falsos positivos e falsos negativos, dificultando acurar a ocorrência do ataque para então mitigá-lo. As abordagens baseadas em replicação, tais como a descrita em [Jia et al. 2014], são computacionalmente caras e podem tolerar apenas um número pré-determinado de falhas. Devido a essas limitações, mesmo empregando as soluções existentes, um ataque DDoS pode acontecer, motivando a criação de outras abordagens.

Este trabalho apresenta SAMOS, uma proposta que complementa os trabalhos na literatura nos casos onde o número de falhas toleradas pelas abordagens de replicação é ultrapassado ou quando a detecção do ataque DDoS é comprometida. SAMOS assume o uso de ferramentas para detectar os efeitos de ataques DDoS em termos de memória e processamento dos IdPs alvos[2], eliminando a dependência de sistemas de detecção de

[1]https://wiki.shibboleth.net/confluence/display/IDP30/Clustering

[2]https://shib.kuleuven.be/docs/idp/2.x/install-idp-2.1-rhel-monitoring.html

intrusão. Além disso, SAMOS emprega IdPs operacionais do SGI, sem ocupar recursos computacionais externos, minimizando o custo da solução.

3. Modelo do Sistema e de Falha

Esta seção detalha os modelos de sistema e de falhas seguidos neste trabalho. A subseção 3.1 apresenta as principais premissas e os componentes do SGI, além da modelagem do mesmo através da teoria dos grafos. A subseção 3.2 descreve o modelo de falhas.

3.1. Modelo do Sistema de Gerenciamento de Identidades

Os principais componentes de um SGI consistem na entidade, identidade, IdP e SPs. Uma entidade pode ser uma pessoa, um serviço de rede ou um dispositivo. Uma identidade é a representação digital de uma entidade real, tal como uma pessoa ou um dispositivo em interações eletrônicas. Os IdPs são as entidades responsáveis por controlar as identidades e prover serviços de autenticação. As entidades incumbidas de disponibilizar serviços especificamente para as identidades consistem nos SPs.

Para fins de modelagem e sem perda de representatividade, o SGI é considerado como um ambiente cooperativo onde IdPs realizam operações de controle e monitoramento sobre as identidades que utilizam recursos disponibilizados por SPs [Cao and Yang 2010]. Consideramos que os IdPs de um domínio se confiam mutuamente e seguem as mesmas diretrizes de segurança, da mesma forma como ocorrem em organizações com um grande número de filiais. Além disso, assumimos que os IdPs utilizam um canal seguro de comunicação para se comunicarem, tal como o provido pelo protocolo *Transport Layer Security* (TLS). O SGI emprega medidas para impedir que atacantes insiram IdPs falsos no SGI, tal como o uso de certificados de segurança.

Nesse trabalho um SGI é representado por um grafo $G = (V, E)$, sendo V composto por vértices provenientes do particionamento de três conjuntos de vértices, A, B e C. Estes conjuntos representam respectivamente as identidades, os IdPs e os SPs do sistema, logo o conjunto de vértices de G é a união dos conjuntos A, B e C. As arestas de G são representadas pelos conjuntos $E1$, $E2$, $E3$, $E4$, $E5$, onde $E1 = \{(a, b)|a \in A, b \in B\}$, $E2 = \{(b, c)|b \in B, c \in C\}$, podendo também existir arestas entre os elementos de cada conjunto, $E3 = \{(a_1, a_2)|a_1 \in A, a_2 \in A\}$, $E4 = \{(b_1, b_2)|b_1 \in B, b_2 \in B\}$, $E5 = \{(c_1, c_2)|c_1 \in C, c_2 \in C\}$, logo $E = \bigcup_{i=1}^{5} E_i$.

Cada conjunto de arestas representa um tipo específico de relação. O conjunto de arestas $E1$ representa quais IdPs $b \in B$ uma identidade $a \in A$ pode usar para se autenticar. $E2$ descreve quais SPs $c \in C$ confiam nas autenticações de um IdP $b \in B$. $E3$ retrata a composição de identidades parciais de um usuário. $E4$ expõe os IdPs afetados por ataques DDoS. $E5$ representa a composição de SPs, ou seja, situações onde serviços podem ser consumidos por outros serviços, tal como ocorre em arquiteturas orientadas a serviço. Em outras palavras, em G qualquer tipo de relação pode existir, exceto as associações de identidades com SPs, pois essa relação não é fiel a forma de operação como os recursos são compartilhados em um SGI. A Figura 1 ilustra um exemplo de grafo G.

Na figura, os vértices de cor branca representam identidades (conjunto A), os de cor cinza representam os IdPs do conjunto B e os de cor preta, os SPs do conjunto C. Observa-se que G é conexo, pois existe apenas uma componente conexa, representando

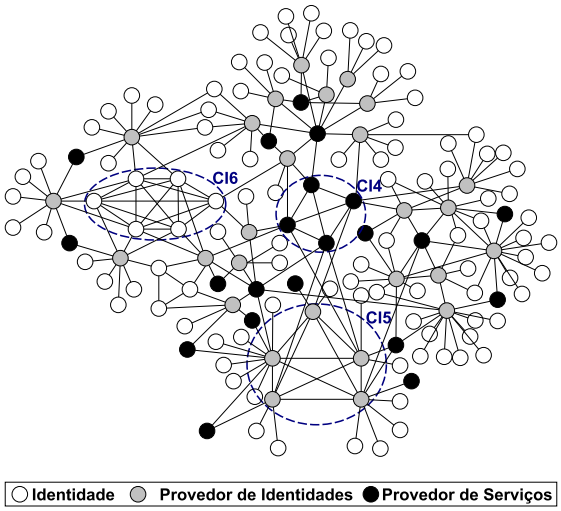

Figura 1. Exemplo de Grafo de um Sistema Gerenciamento de Identidades

o funcionamento normal do sistema. Além disso, nota-se que podem existir ciclos em G entre elementos do mesmo conjunto. Sendo CIj um ciclo onde j vértices são adjacentes, o centro do grafo apresenta um $CI4$ entre elementos de C, representado uma composição de quatro SPs. Logo abaixo do $CI4$, um $CI5$ entre elementos do conjunto B é encontrado, caracterizando cinco ataques DDoS em IdPs com objetivo de indisponibilizar a prestação de suas operações. A esquerda do centro do grafo, um $CI6$ entre os elementos de A, descrevendo a composição de uma identidade através de seis identidades parciais.

3.2. Modelo de Falhas

O modelo de falhas descreve a ocorrência da indisponibilidade em IdPs através da remoção de vértices ($b \in B$), em que as remoções representam IdPs atacados. Logo G se torna desconexo, pois a remoção de IdPs resultará em desconexões entre identidades e SPs (ou seja, a remoção de arestas entre os conjuntos A e C). Assim como em trabalhos existentes [Tan et al. 2011], uma distribuição de probabilidades discreta, mas precisamente a distribuição binomial é utilizada para representar a probabilidade de ocorrência de ataques DoS. A ocorrência de ataques DDoS (ou seja, a remoção de vértices em B) segue uma a distribuição binomial denotada como $Bin(nt, p)$. nt representa o número de tentativas de ataques DDoS e p a probabilidade do atacante obter sucesso, onde cada tentativa resulta apenas em duas possibilidades, sucesso (1) ou fracasso (0), e a probabilidade p de cada tentativa permanece constante. Através desta distribuição são obtidos valores para representar o conjunto de arestas $E4$, as quais indicam IdPs afetados por ataques DDoS.

4. Esquema SAMOS

Esta seção apresenta SAMOS, um esquema para mitigar ataques DDoS em SGI através da reorganizações de agrupamentos de IdPs. As reorganizações proporcionam a otimização do uso dos recursos computacionais do SGI para suprir as sobrecargas causadas por ataques DDoS, minimizando os efeitos do ataque e prolongando o tempo de vida do SGI. A Figura 2 ilustra os procedimentos do esquema SAMOS, sendo eles: *Agrupamento*, *Pré-configuração*, *Otimização*. O *Agrupamento* forma o maior número de agrupamentos para suportar a sobrecarga gerada pelo ataque. A *Pré-Configuração* computa os benefícios possíveis para balancear a carga dos agrupamentos. A *Otimização* emprega técnicas de

otimização para encontrar o benefício máximo de balanceamento de carga ao reorganizar o SGI.

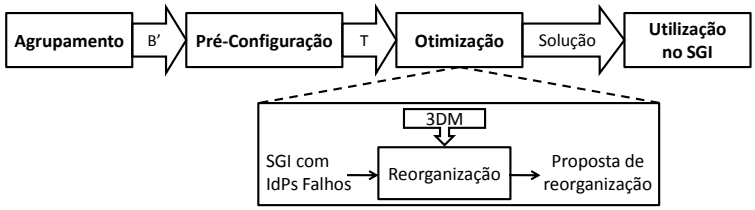

Figura 2. Procedimentos do Esquema SAMOS

4.1. Agrupamento

Este procedimento busca o número máximo de agrupamentos capazes de suportar a sobrecarga gerada pelo ataque DDoS. A solução mais trivial para este problema consiste em formar um único agrupamento com todos os IdPs. Esta alternativa é recomendável quando os membros de um agrupamento estão conectados por uma rede local de alta velocidade, todavia, quando os membros do agrupamento estão geograficamente distribuídos e se comunicam através da Internet, a criação de sub-agrupamentos com poucos nós facilita o gerenciamento da comunicação entre os membros [Aron et al. 2000]. Encontrar o número máximo de agrupamentos capazes de suportar uma sobrecarga consiste em um problema de otimização multiobjetivo: minimizar o número de membros de um agrupamento e maximizar o número total de agrupamentos. Problemas desta natureza são normalmente difíceis de resolver computacionalmente, pois necessitam comparar todas as combinações possíveis para encontrar uma solução ótima e frente a um grande número de elementos pode consumir um tempo de execução não polinomial. Os algoritmos genéticos (AGs) surgem como uma solução heurística capaz de prover boas soluções para problemas de otimização em tempo aceitável ao imitar o processo de seleção natural [Goldberg 1989]. Os experimentos conduzidos por Handl e Knowles revelam benefícios ao utilizar a computação evolucionária para encontrar agrupamentos considerando multiobjetivos [Handl and Knowles 2007]. Seguindo esta linha de pesquisa, este procedimento utiliza o algoritmo genético apresentado no Algoritmo 1.

A entrada para este algoritmo consiste em dados de monitoramento coletados dos IdPs e parâmetros definidos pelos usuários. Cada elemento do vetor v agrega a capacidade disponível dos IdPs operacionais e $capacity$ expressa o dano causado pelo ataque DDoS. O tamanho da população (TP) e número da geração (NG) são parâmetros definidos que ajudam o algoritmo na busca de uma solução, impactando no tempo de execução. A saída do AG consiste em um conjunto de agrupamentos de IdPs capazes de suprir os efeitos do ataque DDoS representado através de um conjunto de subconjuntos.

Na linha 3, *inicie* $P(g)$ gera soluções aleatórias seguindo como restrição a não repetição de elementos em um mesmo agrupamento, ajudando o algoritmo a convergir mais rapidamente para uma solução ótima. A Figura 3 ilustra uma instância simples do problema, uma possível solução e a representação desta solução através de um cromossomo binário. A instância do problema representa uma situação com cinco IdPs com suas respectivas capacidades $(1, 1, 2, 3, 2)$ através do vetor v. Considerando os índices de v, uma solução ótima para este problema é combinar $(0, 2)(1, 4)(3)$, pois as somas dos elementos destes índices é igual a 3 e todos os elementos de v estão representados em S.

Algoritmo 1 Encontra o número máximo de Agrupamentos de IdPs

Entrada: Vetor v, $capacity$, TPS, NG
Saída: Matriz S

1: Seja $P(g)$ a população de soluções na geração g
2: $g \leftarrow 0$
3: inicie $P(g)$
4: avalie $P(g)$
5: classifique $P(g)$ pela função de aptidão
6: **enquanto** solução não encontrada **and** $NG < maxGen$ **faça**
7: $g \leftarrow g + 1$
8: mantenha $P(1)$
9: realize operadores genéticos em $P(g)$
10: avalie $P(g)$
11: classifique $P(g)$ pela função de aptidão
12: **fim enquanto**

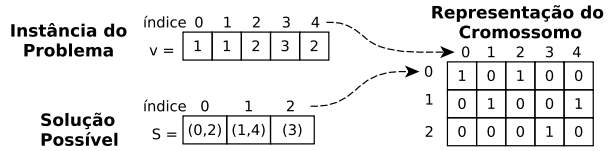

Figura 3. Instância do Problema, Solução e Representação do Cromossomo

Nas linhas 4 e 10, as soluções são avaliadas através da função de aptidão $f(x) = cr1(x) + cr2(x) + cr3(x)$, onde $cr1$ avalia a eficiência dos agrupamentos encontrados na solução x suportarem os efeitos do ataque. $cr2$ verifica se os agrupamentos podem ser fragmentados, favorecendo a criação de um número máximo de soluções e $cr3$ promove soluções com o maior número de agrupamentos capazes de suportar os efeitos do ataque DDoS. O resultado desta função consiste em um valor para classificar as soluções, possibilitando o descarte das soluções menos aptas (linhas 5 e 11). O AG evita que a melhor solução encontrada se perca através das gerações. Para isto, o indivíduo mais apto da geração anterior é mantido na primeira posição da nova geração (ver linha 8), esta técnica é referenciada na literatura de AG como elitismo.

Uma matriz de incidência representa as soluções através de cromossomos binários, possibilitando a execução de operadores genéticos (linha 9). Esta matriz tem o número de colunas igual ao tamanho do vetor v e o número de linhas igual ao tamanho do vetor de uma solução. Cada linha representa um agrupamento de IdPs, quando o elemento está presente no agrupamento, a coluna equivalente ao índice da sua posição é igual a 1, caso contrário 0. A representação da solução através de cromossomos possibilita a execução das operações genéticas de cruzamento e mutação, possibilitando o processo evolutivo das soluções.

Para realizar o cruzamento três soluções aleatórias da população são escolhidas para realização de um torneio. O resultado deste torneio consiste na classificação das duas soluções com maior valor da função de aptidão, as quais são selecionadas para a operação de reprodução. A Figura 4 ilustra a operação de cruzamento. Primeiro, um valor aleatório entre 0 e o número de IdPs é selecionado através de uma taxa de cruzamento.

Este número é usado para selecionar partes dos cromossomos das soluções pais, gerando duas cabeças e duas caldas. Onde, Filho 1, consiste na combinação da cabeça de Pai 1 e calda de Pai 2 e Filho 2 é o resultado da combinação inversa. Após o cruzamento, a taxa de mutação é observada para decidir se as soluções filhos sofrerão mutações, o que garante a geração de novas soluções. Quando uma solução sofre uma mutação, dois IdPs de diferentes agrupamentos são trocados, gerando uma nova solução.

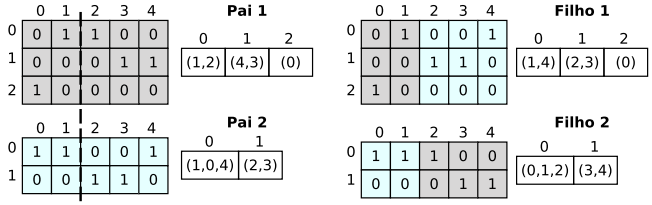

Figura 4. Exemplo da Operação de Cruzamento

Duas são as condições de parada do AG, quando o processo evolutivo encontra uma solução ótima, ou o número máximo de gerações é alcançado. A situação ideal consiste quando o AG encontra uma solução ótima, mas quando esta não for encontrada ou não existir, a solução classificada com o maior valor em sua função de aptidão é retornada.

4.2. Pré-Configuração

A *Pré-Configuração* computa os benefícios possíveis para balancear a carga dos agrupamentos. Cada membro do agrupamento pode apresentar diferentes capacidades de processamento, memória ou fluxo de rede para os outros membros. O ataque DDoS interfere diretamente nas condições normais destas variáveis, congestionando o tráfego da rede do sistema vítima ao transportar um grande volume de requisições maliciosas e exaurindo recursos de processamento e memória. Alguns domínios podem considerar essas variáveis igualmente importantes, mas flexibilizar a computação do benefício pode atender situações onde algumas destas variáveis são mais importantes em detrimento a outras.

Esse procedimento possibilita a atribuição de pesos de importância aos requisitos de rede, processamento de carga ou memória dos IdPs, permitindo aos administradores do SGI empregarem sua expertise para guiar a reorganização. Ambas as funcionalidades utilizam informações providas por ferramentas de monitoramento capazes de informar a taxa de entrega de pacotes entre IdPs (N), taxa de utilização de processamento (P) e taxa de utilização de memória dos IdPs (M). A ferramenta *ping* fornece a taxa de perda de pacotes entre dois computadores, neste caso a T pode ser obtido como o valor inverso desta métrica. A ferramenta *Monit* proporciona M e P, a Universidade Católica de Leuven provê um guia especial para configuração do *Monit* em IdPs [Leuven 2015]. A identificação de IdPs operacionais dá-se através da verificação de um valor limite para P e M, de modo que IdPs com valores inferiores a esse limite deverão ser incluídos no grafo que representa as identidades, os IdPs operacionais e os SPs disponíveis para a reorganização.

Todas as relações possíveis entre identidades, IdPs e SPs são expressas através triplas (a', b', c') associadas com um peso $ben_{a'b'c'} \in \mathbb{R}^+$, representando o benefício que cada associação pode proporcionar. O elemento a' no conjunto de subconjuntos de identidades que poderão ser migrados com um conjunto de subconjuntos de IdPs. O elemento

b' representa o conjunto de subconjuntos de IdPs, e o elemento c' expressa o conjunto de subconjuntos de SPs. A Equação 1 define como um IdP calcula os benefícios de agrupar uma tripla (a', b', c') considerando importâncias relativas aos critérios N, P, M.

$$ben_{a'b'c'} = \frac{(W_N \times N) + (W_M \times M) + (W_P \times P)}{W_N + W_M + W_P} \quad (1)$$

A Equação 1 define o benefício da relação entre os elementos de uma tupla através da média ponderada usando W_N, W_P, W_M como pesos. M representa a taxa de memória ociosa de b'. P denota a taxa de processamento ocioso de b_i. W_N consiste no peso de importância atribuído ao congestionamento da rede. W_P define o peso de importância associado ao processamento. W_M representa o peso de importância para ao critério de memória. Onde N é denotada por $N = \frac{N(b')+N(c')}{2}$, onde $N(b')$ consiste na taxa de entrega de entrega de pacotes para o IdP b' e $N(c')$ representa a taxa de entrega de entrega de pacotes para o SP c'. $M = \frac{M(b')+M(c')}{2}$ é a média da taxa de utilização de memória de b' e c'. $P = \frac{P(b')+P(c')}{2}$ é a média da taxa de utilização do processamento de b' e c'. A atribuição de pesos de importância possibilita guiar as reorganizações para priorizar um melhor fluxo da rede, processamento, ou memória dos IdPs, permitindo aos administradores do SGI utilizarem sua expertise sobre os requisitos do sistema para melhor atender as necessidades dos serviços prestados pelos SPs. Como resultado este procedimento gera o conjunto de triplas T, contendo todas as possibilidades de associação entre identidades, IdPs e SPs em um agrupamento e seu respectivo benefício.

4.3. Otimização

Este procedimento emprega técnicas de otimização para reorganizar a carga do SGI dentro de um agrupamento, tendo como base os benefícios atribuídos pelo procedimento de *Pré-organização*. O objetivo é encontrar uma solução para reorganizar o SGI com o maior benefício existente de modo que todos os nós do agrupamento de IdPs sejam utilizados para atender todas as identidades e SPs. Pela similaridade com o problema de emparelhamento tridimencional (*Three-Dimensional Matching* - 3DM) [Karp 1972], em que uma configuração ótima precisa ser encontrada para associar os elementos de três conjuntos de forma balanceada e sem repetição nas associações, este procedimento de otimização segue os passos para a resolução do 3DM aplicado a um SGI.

O 3DM é resolvido em três passos, a geração de uma matriz tridimensional, a redução para o problema linear e sua resolução. A matriz tridimensional possui tamanho T^3 e representa as relações e benefícios na reorganização. Cada dimensão equivale a um dos conjuntos A', B' e C' no 3DM e T descreve o tamanho de cada conjunto. Os valores dessa matriz representam as relações existentes no SGI no momento do ataque DDoS. A redução da matriz para o problema linear é formalizado como:

$$\begin{array}{lll} \text{Encontrar:} & x_{a'b'c'} \in \{0,1\} & a', b', c' = 0, \ldots, T-1 \\ \text{Maximizar} & \sum_{a'b'c'} ben_{a'b'c'} x_{a'b'c'} & \\ \text{sujeito a} & \sum_{a'b'} x_{a'b'c'} \leq 1 & \forall c' = 0, \ldots, T-1 \\ & \sum_{a'c'} x_{a'b'c'} \leq 1 & \forall b' = 0, \ldots, T-1 \\ & \sum_{b'c'} x_{a'b'c'} \leq 1 & \forall a' = 0, \ldots, T-1 \end{array}$$

Onde, $\sum_{a'b'c'} ben_{a'b'c'} x_{a'b'c'}$ representa a função objetivo pra extrair as melhores relações. $x_{a'b'c'}$ consiste na variável de decisão, de modo que $x_{a'b'c'} = 1$ descreve a situação

onde a tupla (a', b', c') é selecionada no emparelhamento, e caso contrário $x_{a'b'c'} = 0$. $\sum_{a'b'} x_{a'b'c'} \leq 1 \ \forall c' = 0, \ldots, N-1$, $\sum_{a'c'} x_{a'b'c'} \leq 1 \ \forall b' = 0, \ldots, N-1$ e $\sum_{b'c'} x_{a'b'c'} \leq 1 \ \forall a' = 0, \ldots, N-1$ consistem nos limites para restringir a repetição dos elementos nas tuplas. A solução para o problema linear consiste em encontrar uma combinação ótima para as relações considerando o benefício ben e de modo que nenhum elemento apareça em mais de uma tupla, representando uma proposta para reorganização dos componentes do SGI.

5. Avaliação de Desempenho Baseada em Traços

Esta seção apresenta a avaliação de desempenho da proposta. A mesma se baseia em traços do sistema de gerenciamento de identidades da Universidade de Buffalo, EUA. Este sistema foi implementado usando o Shibboleth, que é o arcabouço para gerenciamento de identidades federado mais utilizado no meio acadêmico [Watt et al. 2011]. A subseção 5.1 descreve os traços utilizados e a representação do SGI através da abordagem apresentada na Seção 3. A Subseção 5.2 apresenta a análise do procedimento de agrupamento. A Subseção 5.3 detalha a análise do procedimento otimização.

5.1. Descrição dos Traços e Representação do Sistema *Shibboleth* Analisado

O framework *Shibboleth* possibilita o agrupamento de IdPs (normalmente referenciado como *IdP Clustering*), o qual implementa a operação de migração de atributos das identidades e dados de sessão entre IdPs do mesmo agrupamento. Estas funcionalidades são utilizadas como suporte para implementação do esquema SAMOS. O período de coleta dos dados compreendeu os meses de Abril de 2009 até Setembro de 2013 e estão disponíveis no site da Universidade de *Buffalo* [UB 2013]. Denotamos como H o grafo que representa o sistema modelado, mantendo as propriedades do modelo do sistema descritas na Seção 3. A Figura 5 ilustra o grafo H. Os vértices de cor branca, cinza e preta, representam respectivamente os conjuntos A, B e C. Todas as identidades do SGI (elementos do conjunto A) são representadas pelo vértice branco da figura.

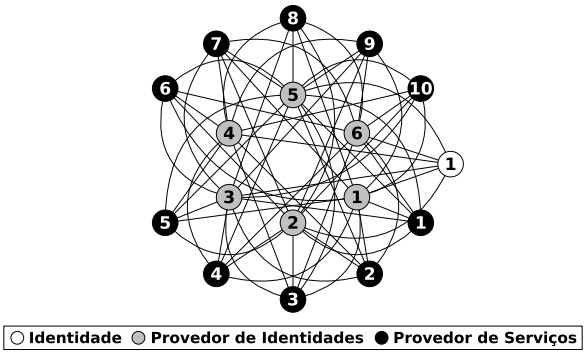

Figura 5. Grafo do Sistema de Gerenciamento de Identidades *Shibboleth*

5.2. Análise do Procedimento de Agrupamento

Esta análise verifica a eficiência do AG para formar agrupamentos de IdP. Três métricas são consideradas na análise, a média da função de aptidão por geração, o tempo de execução e a taxa de soluções ótimas. A média da função de aptidão por geração mostra a

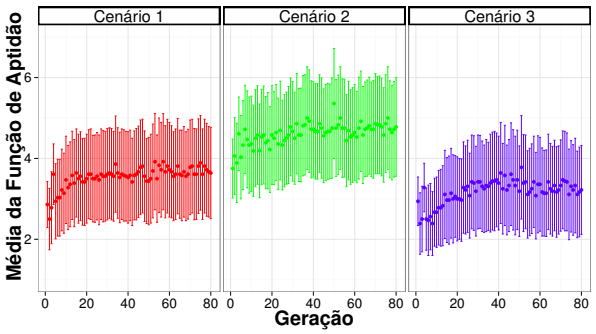

Figura 6. Processo de Evolução

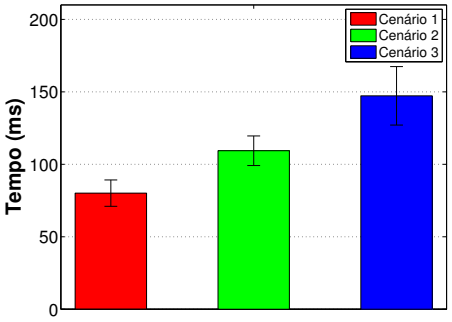

Figura 7. Tempo de Agrupamento

evolução das soluções através do processo evolucionário. O tempo de execução fornece indícios da aplicabilidade do algoritmo em cenários reais. A taxa de soluções ótimas expressa a eficiência do AG em encontrar as melhores soluções.

Diferentes combinações de parâmetros foram escolhidas, formando os cenários da Tabela 1. O tamanho do vetor v foi igual ao seis, representando o número máximo de IdPs do sistema analisado. Cada cenário possui diferentes combinações para o tamanho da população (TP) e número de gerações (NG). A Tabela 2 mostra os parâmetros usados nos cenários. Os valores adotados para a taxa de cruzamento (TC) e taxa de mutação (TM) são recomendados pela literatura de algoritmos genéticos. Valores aleatórios são usados para representar o dano do ataque DDoS e preencher os valores do vetor v.

Tabela 1. Cenários

Cenário 1		Cenário 2		Cenário 3	
TP	NG	TP	NG	TP	NG
15	80	30	80	60	80

Tabela 2. Parâmetros

Parâmetro	Valor
TC	0.9
TM	0.01
$Capacity$	aleatório
Entrada do vetor	aleatório

Para cada cenário 35 simulações foram conduzidas, permitindo o cálculo do intervalo de confiança de 95%. As Figuras 6 e 7 ilustram os resultados obtidos. Observa-se que a média da função de aptidão apresentou pouca variação. Entretanto, a taxa das soluções ótimas encontradas foi igual a 31,37%, indicando a eficiência do algoritmo. O maior tempo de execução observado foi inferior a 200 ms (ver cenário 3), fornecendo indícios da aplicabilidade do AG em situações reais. Os experimentos foram conduzidos em computador Dell PowerEdge T410 equipado com o sistema operacional Debian GNU/Linux versão 6, processador Intel Xeon E5620 (CPU 2.40GHz) e 3GB de memória RAM.

5.3. Análise da Mitigação de Ataques DDoS em SGIs

A análise é realizada em três etapas. A primeira mapeia o subgrafo das relações entre IdPs através de uma matriz de adjacência. A segunda emprega variações nas probabilidades da distribuição binominal para gerar diferentes cenários de ataques DDoS. A terceira mensura o tempo para reorganizar os componentes operacionais do SGI com objetivo de mitigar os efeitos do ataque DDoS.

Denotamos como H' o subgrafo com seis vértices que descreve as relações entre os IdPs do grafo H e M como a matriz de adjacência para representá-lo, logo $M(H') = M_{6,6}$. A diagonal principal da matriz M é preenchida com zeros, indicando a

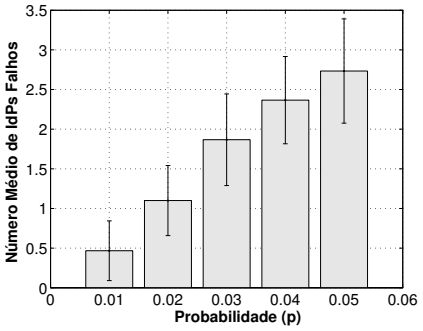

Figura 8. Média de IdPs Falhos

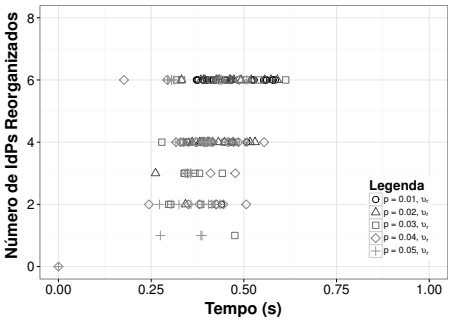

Figura 9. Dispersão

inexistência de arestas de um vértice para si mesmo. Os valores do triângulo superior e inferior variam entre zero e um de forma simétrica ao longo da diagonal principal para compor o conjunto de arestas $E4$. A garantia da simetria da matriz ao longo da diagonal principal é garantida através da soma da matriz com a diagonal principal e o triângulo inferior mais sua transposta, ou seja, $M = T(M)$.

A distribuição binomial é aplicada com diferentes probabilidades para obtenção dos valores da matriz de adjacência, gerando um espaço amostral da ocorrência de ataques DDoS. Para representar as arestas entre os seis vértices são necessários quinze valores distribuídos de forma simétrica entre o triângulo superior e inferior, variando entre zero e um. Os valores utilizados para representar a probabilidade p de ocorrência de ataques entre dois IdPs foram 0,01, 0,02, 0,03, 0,04 e 0,05. Para cada uma das variações de probabilidades, 30 amostras são colhidas. O número de tentativas de ataques DDoS nt utilizado foi igual a 450, como resultado da multiplicação do número de amostras (30) pelo número de elementos necessários para completar o triângulo superior de M, neste caso 15. Como resultado da aplicação destes valores em M, cada amostra do subgrafo H' apresenta IdPs com grau zero, ou seja, não terão nenhum vizinho, esses vértices representam os IdPs operacionais durante o ataque DDoS.

Duas métricas são usadas, a velocidade de reorganização e o número médio de IdPs falhos. A velocidade de reorganização denotada pela equação $\upsilon_r = \frac{\Delta_n}{\Delta_t}$ é definida. Onde υ_r é a velocidade de reorganização. Δ_n é o número de IdPs operacionais durante o ataque DDoS, obtido pela subtração do Número de Provedores de Identidades Falhos (NPIF) do Número Total de IdPs (NTPI), ($\Delta_n = NTPI - NPIF$). Δ_t é o tempo de máquina para encontrar uma solução com a instância n. O número médio de IdPs falhos é denotado como: $\eta = \frac{1}{k}\sum_{i=1}^{k} NPIF_i$, onde k é o número de amostras para uma probabilidade p e $NPIF_i$ é o número de IdPs falhos na amostra i.

A análise utiliza valores aleatórios para os benefícios das tuplas. Além disso, consideramos que ao organizar esses conjuntos, o conjunto de subconjuntos B' apresenta sempre o mesmo tamanho do número de IdPs operacionais durante o ataque. Dessa forma, os IdPs operacionais extraídos de cada amostra da matriz M foram passados como entrada para o protótipo encontrar uma solução ótima do 3DM. A Figura 8 mostra o número médio de IdPs falhos usando 95% de intervalo de confiança e a Figura 9 ilustra os resultados obtidos pela velocidade de reorganização.

Os resultados obtidos revelam que o esquema SAMOS é capaz de reorganizar rapidamente um SGI com poucos IdPs. Os resultados da Figura 8 mostram que o aumento

da probabilidade de sucesso de ataques DDoS implica diretamente no crescimento de IdPs falhos, gerando diferentes cenários de ataques. A Figura 9 ilustra a velocidade de reorganização considerando os diferentes cenários de ataques. A velocidade de reorganização para o maior número de IdPs foi inferior a 0,75 segundos. Esse resultado advoga que para um número pequeno de IdPs, a solução executa em tempo aceitável, apresentando potencial para mitigar ataques DDoS como uma abordagem *bottom-up*, onde a mitigação dos efeitos do ataque em um SGI ocorre de baixo para cima.

6. Conclusão

Este trabalho apresentou o esquema SAMOS, a primeira proposta para mitigar os efeitos de ataques DDoS em SGIs através de reorganizações de reagrupamentos de IdPs. SAMOS utiliza um algoritmo genético para encontrar melhores soluções de agrupamentos de IdPs e emprega técnicas de otimização para balancear a carga do sistema entre os IdPs operacionais, empregando de forma mais eficiente os recursos computacionais do SGI. O esquema proposto também possibilita aos administradores do SGI utilizarem sua expertise sobre os requisitos do sistema para guiar as reorganizações para priorizar o melhor fluxo da rede, processamento, ou memória dos IdPs. Duas avaliações foram conduzidas envolvendo dados reais de um SGI para verificar a eficiência do algoritmo para encontrar soluções de agrupamentos de IdPs e reorganizar o sistema. Os resultados obtidos mostram indícios da viabilidade desta proposta, possibilitando a mitigação dos efeitos do ataque primeiramente nos agrupamentos e abordando o SGI de modo recursivo. Como trabalhos futuros pretende-se *(i)* implementar um protótipo da solução através do framework *Shibboleth* e *(ii)* realizar análises de desempenho do protótipo em um laboratório de gerenciamento de identidades.

Referências

Arias Cabarcos, P., Almenárez, F., Gómez Mármol, F., and Marín, A. (2014). To federate or not to federate: A reputation-based mechanism to dynamize cooperation in identity management. *Wirel. Pers. Commun.*, 75(3):1769–1786.

Aron, M., Druschel, P., and Zwaenepoel, W. (2000). Cluster reserves: A mechanism for resource management in cluster-based network servers. In *Proceedings of the 2000 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pages 90–101, New York, NY, USA. ACM.

Barreto, L., Siqueira, F., Fraga, J., and Feitosa, E. (2013). An intrusion tolerant identity management infrastructure for cloud computing services. In *IEEE International Conference on Web Services*, pages 155–162.

Cao, Y. and Yang, L. (2010). A survey of identity management technology. In *IEEE International Conference on Information Theory and Information Security*, pages 287 –293.

Carlson, F. R. (2014). Security analysis of cloud computing. *CoRR*, abs/1404.6849.

Compagno, A., Conti, M., Gasti, P., and Tsudik, G. (2013). Poseidon: Mitigating interest flooding ddos attacks in named data networking. In *IEEE Conference on Local Computer Networks*, pages 630–638.

Fu, Z., Papatriantafilou, M., and Tsigas, P. (2012). Mitigating distributed denial of service attacks in multiparty applications in the presence of clock drifts. *IEEE Transactions on Dependable and Secure Computing*, 9(3):401–413.

Giotis, K., Argyropoulos, C., Androulidakis, G., Kalogeras, D., and Maglaris, V. (2014). Combining openflow and sflow for an effective and scalable anomaly detection and mitigation mechanism on sdn environments. *Computer Networks*, 62(0):122 – 136.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition.

Handl, J. and Knowles, J. (2007). An evolutionary approach to multiobjective clustering. *Evolutionary Computation, IEEE Transactions on*, 11(1):56–76.

Jia, Q., Wang, H., Fleck, D., Li, F., Stavrou, A., and Powell, W. (2014). Catch me if you can: A cloud-enabled ddos defense. In *IEEE/IFIP DSN*, pages 264–275.

Karp, R. (1972). Reducibility among combinatorial problems. In Miller, R. and Thatcher, J., editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press.

Kreutz, D., Feitosa, E., and Cunha, H. (2014). Provedores de identidade resilientes e confiáveis. *Anais do XV Workshop de Testes e Tolerância a Falhas*.

Leuven (2015). Guide: Local monitoring of a shibboleth identity provider. `https://shib.kuleuven.be/docs/idp/2.x/install-idp-2.1-rhel-monitoring.html`. Último Acesso: Junho de 2015.

Lonea, A., Tianfield, H., and Popescu, D. (2013). Identity management for cloud computing. In Balas, V. E., Fodor, J., and Várkonyi-Kóczy, A. R., editors, *New Concepts and Applications in Soft Computing*, volume 417 of *Studies in Computational Intelligence*, pages 175–199. Springer Berlin Heidelberg.

Shah, H., Anandane, S. S., and Shrikanth (2013). Security issues on cloud computing. *CoRR*, abs/1308.5996.

Tan, Y., Sengupta, S., and Subbalakshmi, K. (2011). Analysis of coordinated denial-of-service attacks in ieee 802.22 networks. *IEEE Journal on Selected Areas in Communications*, 29(4):890–902.

Torres, J., Nogueira, M., and Pujolle, G. (2013). A survey on identity management for the future network. *IEEE Communications and Surveys Tutorials*, 15(2):787–802.

UB (2013). UB Identity Management and Authentication Metrics. `https://ubidm.buffalo.edu/stats/`. Último Acesso em Outubro de 2013.

Watt, J., Sinnott, R., Inman, G., and Chadwick, D. (2011). Federated authentication and authorisation in the social science domain. In *International Conference on Availability, Reliability and Security*, pages 541 –548.

