Resistindo a Ataques de Personificação no Gerenciamento de Chaves Públicas em Redes Ad Hoc Móveis: *Virtual Public-Key Management System*

Renan Fischer e Silva, Eduardo da Silva, Luiz Carlos Pessoa Albini

[1]NR2 – Departamento de Informática – Universidade Federal do Paraná (UFPR)
Caixa Postal 19.081 – 81.531-980 – Curitiba – PR – Brasil

e-mail: {renan, eduardos, albini}@inf.ufpr.br

Abstract. *Chaining-based key management schemes seams to be the ones that best fit the MANET paradigms. The main chaining-based scheme is the Self-Organized Public Key Management System (PGP-Like). However, it is fully vulnerable to impersonation attacks. In order to reduce such vulnerability, this article introduces a new public-key management system for MANETs, the Virtual Key Management System (VKM). VKM uses a virtual structure to indicate the trust between nodes and the certicate chains formation. VKM is a flexible key management scheme. It can behave in a restrict way, being able to tolerate impersonation attacks to a certain level, or it can behave similarly to the PGP-Like, just by changing a simple parameter. Thus, VKM can suite any user needs with its ability to switch between the two models dynamically, without any network reinitialization or reconfiguration.*

Resumo. *Esquemas de gerenciamento de chaves baseados em cadeias de certificados mostram-se como os melhores para MANETs. O principal esquema baseado em cadeias de certificados é o Self-Organized Public Key Management System (PGP-Like). Entretanto, ele é completamente vulnerável a ataques de personificação. De maneira a reduzir tal vulnerabilidade, este artigo introduz um novo esquema de gerenciamento de chaves para MANETs, o Virtual Key Management System (VKM). O VKM faz uso de uma estrutura virtual para indicar a confiança entre os nós e a formação de cadeias de certificados. O VKM é um esquema flexível podendo se comportar de uma maneira restrita, capaz de suportar ataques de personificação até um certo nível, ou se comportar de forma similar ao PGP-Like, mudando um simples parâmetro. Portanto, o VKM pode suprir a necessidade de qualquer usuário sendo capaz de alterar o seu comportamento dinamicamente, sem qualquer reinicialização ou reconfiguração.*

1. INTRODUÇÃO

Esquemas de gerenciamento de chaves para Redes *Ad Hoc* Móveis (*Mobile Ad Hoc Networks* - MANETs) precisam funcionar em ambientes com topologia dinâmica e serem auto-organizáveis e descentralizados [Hegland et al. 2006, Čapkun et al. 2006, van der Merwe et al. 2007, Silva et al. 2008]. Além disso, devem satisfazer alguns requisitos como [Menezes et al. 1996]: não ter um ponto único de falha; ser tolerante ao comprometimento ds nós; ser capaz de revogar as chaves dos nós comprometidos e atualizar as chaves dos nós não-comprometidos; ser eficiente quanto ao armazenamento, o processamento e a comunicação.

É possível classificar os esquemas de gerenciamento de chaves para MANETs em [van der Merwe et al. 2007]: baseados em identidade [Khalili et al. 2003], baseados em cadeias de certificados [Hubaux et al. 2001, Čapkun et al. 2003a, Čapkun et al. 2006], baseados em *clusters* [Ngai and Lyu 2004, Ngai et al. 2004], baseados em pré-distribuição [Eschenauer and Gligor 2002] e baseados em mobilidade [Čapkun et al. 2003b]. Entre estes, os esquemas baseados em cadeias de certificados parecem ser os melhores para aplicações em ambientes MANETs. O principal esquema de gerenciamento de chaves baseado em cadeias de certificados é o *Self-Organized Public Key Management System* [Hubaux et al. 2001, Čapkun et al. 2003a], chamado neste trabalho de *PGP-Like*.

O *PGP-Like* é um esquema de gerenciamento de chaves auto-organizável baseado nos conceitos do PGP [Zimmermann 1995], no qual todo par de chaves pública e privada é criado pelos próprios nós da rede. Os nós também emitem certificados para outros nós em que confiam. Cada nó tem um repositório local de certificados que é periodicamente trocado com seus vizinhos. As chaves públicas são autenticadas por meio de cadeias de certificados, construídas usando os repositórios locais de certificados.

Como mostrado em [Silva et al. 2008], o *PGP-Like* é altamente vulnerável à ataques de personificação (*Sybil*). Ataques de personificação consistem em um atacante usar identidades falsas para enganar, alterar e/ou prejudicar o funcionamento dos protocolos da rede [Douceur 2001]. A funcionalidade do *PGP-Like* é comprometida mesmo quando apenas 5% de nós maliciosos estão presentes na rede [Silva et al. 2008]. De maneira a reduzir tal vulnerabilidade, este artigo introduz um novo esquema de gerenciamento de chaves públicas para MANETs, o *Virtual Key Management System* (*VKM*). O VKM usa uma estrutura virtual para indicar a confiança entre os nós e a formação das cadeias de certificados. Estruturas virtuais já foram usadas em protocolos de roteamento para MANETs, como no *Virtual Routing Protocol* (VRP) [Albini et al. 2006] e no *Virtual Distance Vector* (VDV) [Robba and Maestrini 2007]. A estrutura virtual é altamente redundante e não possui relação com a localização física das unidades da rede.

O VKM é um esquema de gerenciamento de chaves bastante flexível. Ele pode operar de uma maneira restrita, sendo capaz de suportar ataques de personificação até um certo nível, ou operar de maneira similar ao *PGP-Like* apenas mudando um único parâmetro. Quando o VKM estiver configurado para operar de maneira restrita, ele é capaz de completar quase 80% das requisições de autenticação com 5% de nós comprometidos na rede. Mesmo com 20% de nós comprometidos, VKM ainda é capaz de completar 50% de todas as requisições de autenticação.

O VKM ainda pode operar de maneira similar ao *PGP-Like*. Para demonstrar o funcionamento do VKM quando operando de maneira similar ao *PGP-Like*, ele foi avaliado em ambientes sem ataques e na presença de ataques de falta de cooperação seguindo os parâmetros usados em [Silva et al. 2008]. Ataques de falta de cooperação consistem em nós egoístas usando os recursos da rede mas não cooperando com alguma ou com todas as operação da mesma [Michiardi and Molva 2003]. Neste caso, o VKM tem o mesmo comportamento do *PGP-Like*. Portanto, o VKM pode suprir a necessidade de qualquer usuário com sua capacidade de se comportar de maneiras distintas dinamicamente sem nenhuma reinicialização ou reconfiguração da rede.

O resto deste artigo é organizado da seguinte forma: a seção 2 descreve brevemente as características do *PGP-Like*; a seção 3 detalha o esquema *Virtual Key Management*; a seção 4 apresenta a avaliação do VKM e a comparação com o *PGP-Like*; finalmente, a seção 5 apresenta as conclusões e os trabalhos futuros.

2. GERENCIAMENTO DE CHAVES PÚBLICAS AUTO-ORGANIZADO

O *Self-Organized Public Key Management System*, chamado neste trabalho de *PGP-Like*, é um esquema de gerenciamento de chaves públicas baseado em cadeias de certificados [Čapkun et al. 2003a, Hubaux et al. 2001]. As chaves públicas e privadas dos nós são criadas pelos própios nós, seguindo os conceitos do PGP [Zimmermann 1995]. Além disso, cada nó emite certificados de chave pública para outros nós nos quais confia.

No PGP-*Like*, se um nó u acredita que uma dada chave pública K_v pertence a um dado nó v, ele pode emitir um certificado associando K_v ao nó v, $(v,K_v)_{prK_u}$, em que prK_u é a chave privada do nó u. Esse certificado é armazenado no repositório local de certificados de u e de v. Adicionalmente, cada nó periodicamente realiza trocas de seu repositório com seus vizinhos físicos.

As chaves públicas e os certificados são representados por um grafo dirigido $G = (V,A)$, no qual V representa as chaves públicas e A representa os certificados. Portanto, uma aresta dirigida entre dois vértices K_u e K_w, $(K_u \rightarrow K_w)$, denota um certificado, assinado por u, associando K_w ao nó w. Já um caminho conectando dois vértices, K_u e K_w, é representado por $(K_u \rightsquigarrow K_w)$. Cada nó u mantém um repositório local de certificados atualizados, G_u, e um outro repositório local de certificados não atualizados, G_u^N [Čapkun et al. 2003a]. O repositório local de certificados não atualizados contém os certificados que expiraram e foram considerados revogados. As figuras 1a e 1b ilustram os repositórios locais de certificados dos nós u e u, G_u e G_v, respectivamente.

Quando o nó u quer autenticar a chave pública K_v do nó v, ele primeiramente tenta achar um caminho de certificados do vértice K_u até o vértice K_v em G_u (Figura 1a). Se $\exists\,(K_u \rightsquigarrow K_v) \in G_u$, o nó u autentica a chave pública do nó v. Se $\nexists\,(K_u \rightsquigarrow K_v) \in G_u$, o nó u une G_u com G_v, criando $G_1 = G_u \cup G_v$ (Figura 1c), e tenta achar $(K_u \rightsquigarrow K_v) \in G_1$. Se tal caminho existe, a autenticação é realizada com sucesso. Se $\nexists\,(K_u \rightsquigarrow Kv) \in G_1$, então o nó u cria $G_2 = G_u \cup G_u^N$ e procura $(K_u \rightsquigarrow K_v) \in G_2$. Se $\exists\,(K_u \rightsquigarrow K_v) \in G_2$, o nó u precisa verificar todos os certificados não validados na cadeia. Por fim, se $\nexists\,(K_u \rightsquigarrow K_v) \in G_2$, então o nó u não consegue autenticar a chave pública K_v.

O caminho encontrado nos repositórios é uma cadeia de certificados. Cadeias de certificados representam a confiança entre os nós e são chamadas de cadeias de confiança. Note que, as cadeias de confianças são consideradas autenticações fracas, pois assume-se transitividade. Por exemplo, se o nó A confia no nó B, e o nó B confia no nó C, então o nó A também confia no nó C. Infelizmente, garantir uma confiança transitiva válida entre mais de dois nós em uma cadeia é muito difícil [Christianson 1996]. Por tal motivo, se qualquer nó da cadeia for comprometido, todos os outros nós pertencentes à cadeia podem obter uma autenticação falsa.

O uso de cadeias de certificados torna o *PGP-Like* altamente vulnerável à ataques de personificação, como mostrado nas figuras 2 e 3 [Silva et al. 2008]. A figura 2 mostra o percentual de nós que contem identidades falsas em seus respositórios locais por tempo,

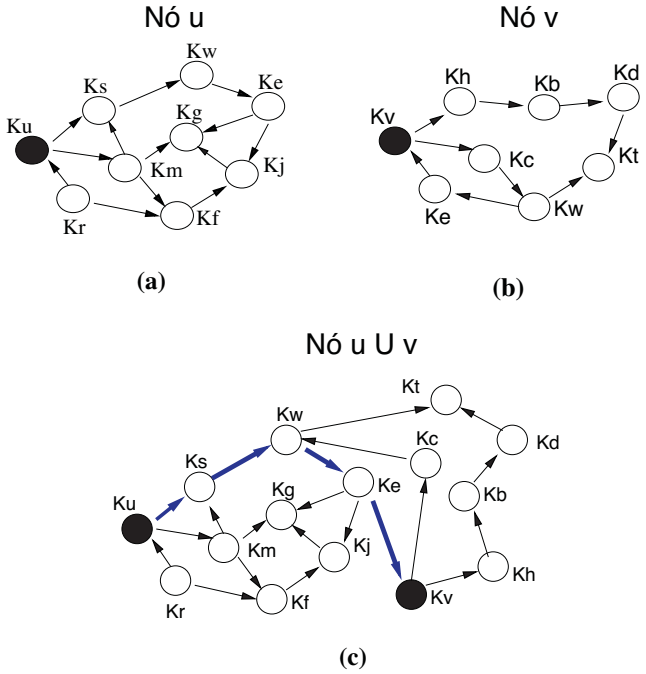

Figura 1. Repositórios de certificados e uma cadeia de certificado

segundos após a inicialização da redes, e para 5%, 10% e 20% de atacantes na rede. A figura 3 mostra que as identidades falsas conseguem ser autenticadas por nós verdadeiros. Note que, um nó atacante x pode criar uma identidade falsa m e emitir certificados amarrando K_m à m. Todos os nós que confiarem em x também confiarão em m. Portanto, se o nó x mantiver um comportamento correto durante um tempo consideravel, várias únidades irão, provavelmente, confiar nele, e a identidade falsa irá se espalhar pela rede devido ao mecanismo de troca de certificados.

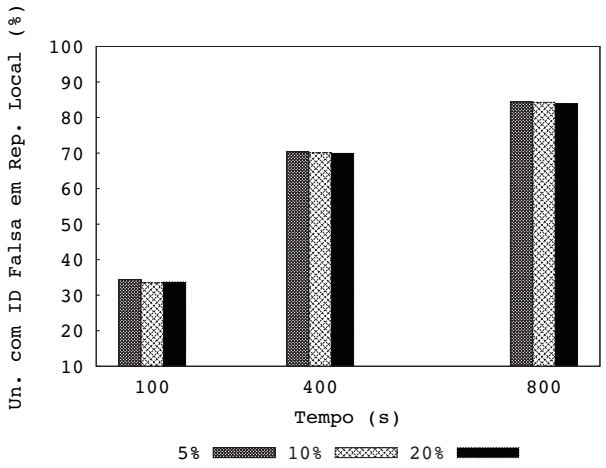

Figura 2. Confiança nas Identidades Falsas

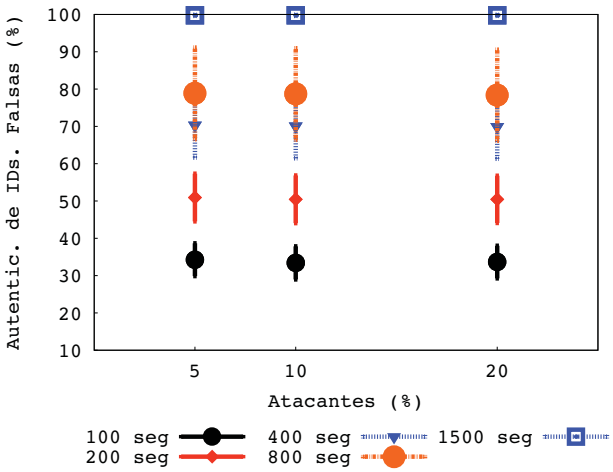

Figura 3. Certificados Falsos nos Reposit órios Locais

3. *VIRTUAL KEY MANAGEMENT*

O esquema *Virtual Key Management* (VKM) usa uma *estrutura virtual* para indicar a confiança entre os nós e a formação de cadeias de certificados. A estrutura virtual é representada por um grafo dirigido $L = (D,E)$, que não está relacionado com a topologia atual da rede. O conjunto de vértices D representa os nós e o conjunto de arestas E representa os enlaces virtuais. Um enlace virtual $(i, j) \in E$, indica que o nó i emite um certificado associando K_j ao nó j. Note que o nó i precisa fazer esse procedimento para cada nó que possuir uma conexão direcionada na estrutura virtual.

É importante mencionar que o VKM é independente da implementação do grafo da estrutura virtual. Entretanto, percebe-se que o grafo deve ser regular para garantir que o número de arestas seja o mesmo para todos os nós. A estrutura virtual mais apropriada deve ser selecionada pelo usuário considerando propriedades como diâmetro, largura da bissecção e escalabilidade. Por exemplo, a estrutura virtual pode ser um *Ring of Rings*, um hipercubo, um CCC ou um Torus, embora os resultados reportados nesse artigo tenham sido obtidos utilizando *Rings of Rings* (RoR), que é detalhada a seguir.

A estrutura *Rings of Rings* (RoR) é baseada em congruências [Vinogradov 1955]. Assumindo dois inteiros, x e y, tal que $x * y = n$, e sendo s um inteiro tal que $1 < s \leq y$. O conjunto D é particionado em x anéis, chamados $D_0, D_1, ..., D_{x-1}$, e para cada $a \in [0,x)$, $D_a = \{i : a * y \leq i < (a+1) * y\}$. O enlace (i, j) pertence à E se, e somente se: $j \; mod \; y = (i+d) \; mod \; y$ para algum $1 \leq d < s$; ou $j = (i+y) \; mod \; n$. Uma característica da estrutura RoR é a redundancia de caminhos virtuais, no qual o grau é determinado por parâmetros x, y, e s.

A figura 4 exemplifica o grafo *Ring of Rings* (RoR) [Albini et al. 2006], com 45 nós, dividido em 3 anéis de 15 nós. Cada nó tem uma conexão direta para outros cinco nós, significando que eles são responsáveis por emitir cinco certificados associando a chave pública desses nós com suas respectivas identidades. Por exemplo, na figura 4, nó w é responsável por emitir certificados associado K_{w_1} à w_1, K_{w_2} à w_2, K_{w_3} à w_3, K_{w_4} à w_4 e K_{w_5} à w_5, e os nós w'_1, w'_2, w'_3, w'_4 e w'_5 são responsáveis por emitir certificados associando K_w à w.

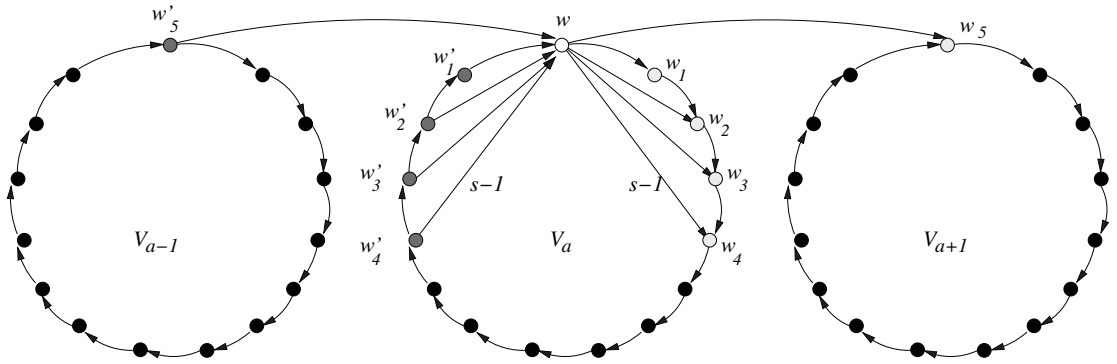

Figura 4. Estrutura virtual RoR com 3 aneis e 15 nós por anel

No VKM, cada nó i cria seu próprio par de chaves pública e privada, K_i e prK_i. Posteriormente, ele precisa emitir certificados seguindo a estrutura virtual. Um par de nós na estrutura virtual precisa trocar suas chaves por um canal seguro como infra-vermelho, *smart cards* ou antes da formação da rede. Todos os certificados são emitidos com um tempo de vida limitado T_v, e após T_v, o certificado é considerado expirado. Antes da expiração do certificado, o nó que o emitiu pode atualizar o certificado, emitindo uma nova versão com um T_v extendido. A revogação de certificados funciona da mesma forma que no *PGP-Like*. Pode ser feita de uma maneira explícita ou implícita. A revogação implícita é baseada no tempo de validade dos certificados: se um emissor não atualiza seu certificado após T_v, o certificado é considerado revogado. Já na revogação explicita, o nó emissor revoga um certificado se suspeita de mal comportamento do outro nó.

Quando um certificado é emitido, o nó emissor armazena esse certificado em seu repositório local e o envia para o nó correspondente, que também armazena o certificado. Assim, no início do tempo de vida da rede, os nós armazenam apenas os certificados que eles emitiram e os certificados que foram emitidos para ele. O uso da estrutura virtual torna o VKM muito flexível, podendo se comportar de uma maneira restrita e sendo capaz de suportar ataques de personificação até um certo nível, ou podendo se comportar de maneira similar ao *PGP-Like*, apenas mudando alguns parâmetros. A principal diferença entre os comportamentos é a maneira que os nós autenticam as chaves públicas. Ambas as formas serão apresentadas a seguir.

3.1. VKM com autenticação reativa

No modo de autenticação reativa, *VKM with Reative Authentication* (VKM-RA), cada nó mantém apenas os seus certificados iniciais, ou seja, os certificados emitidos para ele e os certificados que ele emitiu. Quando um nó i quer autenticar a chave pública de um nó j, ele precisa encontrar um caminho virtual de i até j, uma cadeia de certificados, na estrutura virtual. Note que é possível encontrar vários caminhos virtuais de i até j, uma vez que a estrutura virtual é altamente redundante. Após escolher um caminho virtual, o nó i precisa obter todos os certificados para validar todo o caminho virtual, ou seja, validar toda a cadeia de certificados. Cada certificado precisa ser validade diretamente com o nó que o emitiu.

Todos os nós conhecem a mesma estrutura virtual, mas eles não mantém informações atualizadas sobre os certificados, por exemplo, se eles foram revogados.

Diferentemente do *PGP-Like*, uma unidade precisa manter apenas os certificados emitidos para ela, e os certificados emitidos por ela. Isso reduz a memória necessária para armazenar os certificados. Entretanto, a origem precisa requisistar, reativamente, todos os certificados da cadeia de certificados. Assim, a autenticação é realizada da seguinte maneira:

1. o primeiro certificado pode ser diretamente verificado pelo nó i usando a sua própria chave pública, uma vez que foi ele quem emitiu o certificado;
2. cada certificado remanescente pode ser verificado usando a chave pública contida no certificado anterior;
3. finalmente, o último certificado contém a chave pública do nó j.

Esse comportamento garante que apenas certificados corretos e válidos são utilizados. Entretanto, como os nós precisam requisitar todos os certificados de uma cadeia de certificados antes de autenticar a chave pública, isto implica em uma latência nas autenticações. Por outro lado, o VKM-RA utiliza pouca memória para armazenamento total. Uma unidade precisa manter armazenado localmente apenas os certificados emitidos por ela, os certificados emitidos para ela e uma pequena função para criar e usar a estrutura virtual. Se a rede fizer uso de um protocolo de roteamento que utilize uma estrutura virtual, como VRP [Albini et al. 2006] ou VDV [Robba and Maestrini 2007], o VKM pode usar a essa mesma estrutura virtual.

3.2. VKM com autenticação proativa

O modo de autenticação pró-ativa, *VKM with Proactive Authentication* (VKM-PA), tem um comportamento similar ao *PGP-Like*. Embora os certificados sejam emitidos seguindo uma estrutura virtual, os nós realizam trocas periódicas de seus repositórios de certificados com seus vizinhos físicos. Por simplicidade e sem perda de generalidade, é assumido que todos os nós possuem o mesmo intervalo de troca T_{ex} e que as trocas não são simétricas. Isto é, se o nó i está enviando um certificado para o nó j, não quer dizer que o nó j precisa mandar os seus certificados para o nó i.

Como no *PGP-Like*, os nós armazenarão a informação adquirida no repositório local de certificados. Então, quando um nó i quer autenticar uma chave pública K_j do nó j, ele procura uma cadeia de certificados no seu repositório local, $(K_i \rightsquigarrow K_j) \in G_i$. Se $\exists\ (K_i \rightsquigarrow K_j) \in G_i$, ele realiza a autenticação. Se $\nexists\ (K_i \rightsquigarrow K_j) \in G_i$, nó i cria $G_\alpha = G_i \cup G_i^N$ e procura $(K_i \rightsquigarrow K_j) \in G_\alpha$. Se $\exists\ (K_i \rightsquigarrow K_j) \in G_\alpha$ o nó i precisa validar todos os certificados expirados antes de usá-los. Se $\nexists\ (K_i \rightsquigarrow K_j) \in G_\alpha$ então o nó i invoca o uso do VKM-RA. Esta característica faz a autenticação do VKM mais eficiente que a do *PGP-Like*, uma vez que é possível atingir todos os nós da rede usando VKM-RA.

3.3. Avaliando o VKM-RA e o VKM-PA

Como no VKM-RA os nós não realizam trocas de seus repositórios de certificados, a sobrecarga das trocas de repositórios é eliminada. Porém isso inclui um atraso necessário para validar cada certificado da cadeia. Esse atraso depende do protocolo de roteamento usado pela rede e do tamanho das cadeias de certificados. São necessárias duas mensagens para cada nó da cadeia de certificados. Por exemplo, na Figura 5 o nó i precisa mandar

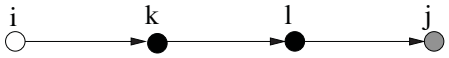

Figura 5. Cadeia de certificados

uma mensagem para o nó k para validar $(l, K_l)_{prK_k}$ e uma mensagem para o nó l para validar $(j, K_j)_{prK_j}$. Após receber ambas as respostas, i pode autenticar K_j.

Por outro lado, o VKM-PA pode eliminar o atraso na autenticação se uma cadeia de certificados válida e atualizada for encontrada no repositório local. Caso contrário, é necessário invocar o VKM-RA. Portanto, o atraso do VKM-PA pode ser menor ou igual ao do VKM-RA, dependendo da completude e validade do repositório local de certificados. O VKM-PA possuí uma sobrecarga para realizar as trocas de certificados, e dependendo da completude do repositório local de certificados, pode ser possível ter um atraso igual ao VKM-RA. Outro problema é a memória de armazenamento, pois enquanto no VKM-RA os nós só mantém cópias de alguns certificados, no VKM-PA e no *PGP-Like*, os nós irão eventualmente manter cópias de todos os certificados da rede.

Outra característica importante do VKM é a habilidade de alternar entre os dois modelos apresentados dinamicamente, sem reinicialização da rede ou mesmo sem qualquer reconfiguração. Então, quando energia ou memória for uma restrição para a rede, os nós podem operar usando VKM-RA, enquanto em outras situações, eles podem usar o VKM-PA.

4. AVALIAÇÕES E RESULTADOS DAS SIMULAÇÕES

O simulador *Network Simulator* 2 (NS-2) [NS-2 2007], em sua versão 2.30, foi usado para verificar a eficácia do VKM quando submetido a ataques de personificação. O VKM também foi comparado com o *PGP-Like* em ambientes sem ataques e sofrendo ataques de falta de cooperação. Os resultados aqui apresentados para o *PGP-Like* são os mesmos exibidos em [Silva et al. 2008].

Os parâmetros usados nas simulações estão na Tabela 1. Os resultados são as médias de 35 simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR, com 4 anéis e 25 nós por anel. Cada nó emite 5 certificados e tem 5 certificados emitidos para ele. No VKM-PA e no *PGP-Like* o intervalo para trocas dos repositórios locais de certificados é de 60 segundos e o mesmo não é simétrico. Note que apenas os resultados para o pior caso, de acordo com [Silva et al. 2008], estão reportados.

4.1. Ataques de personificação

Para avaliar o VKM-RA sob o ataque de personificação, uma nova métrica é proposta: *Chains with Sybil Nodes* (*CSN*). *CSN* é a porção de caminhos virtuais, ou seja, cadeias de certificados, que contém ao menos uma identidade falsa (m). Ela representa quantas autenticações são feitas usando cadeias de certificados comprometidas, isto é, que tenham pelo menos um nó personificado. *CSN* pode ser definida como:

$$CSN = \frac{\sum CSN_i}{|NA|} \quad \forall\, i \in NA \qquad \text{onde} \tag{1}$$

$$CSN_i = \begin{cases} 1 & \text{se} \quad \exists\, m \in (K_i \rightsquigarrow K_j) \\ 0 & \text{caso contrário} \end{cases} \tag{2}$$

Tabela 1. Cenários da simulação

Parâmetros	Valores
Dimensão da rede	1000 x 1000 metros
Raio da transmissão	120 metros
Quantidade de nós	100
Modelo de mobilidade	*random waypoint*
Velocidade máxima	20 m/s
Tempo de pausa máximo	20 segundos
Intervalo de troca de certificados	60 segundos
Tempo de Simulação	1500 segundos
Modelo de Propagação	*two-ray ground reflection*
Protocolo de Acesso ao Meio	IEEE 802.11

Como o VKM-RA precisa seguir a definição da estrutura virtual em todas as autenticações, o ataque de personificação é o único que pode ser efetivo contra a rede. Ataques que os nós maliciosos criam identidades falsas para obter algum tipo de vantagem ou mesmo prejudicar o funcionamento da rede não seriam eficazes, uma vez que os nós devem pertencer à estrutura virtual. Portanto, se um atacante utilizar uma falsa identidade que não faça parte da estrutura virtual ou mesmo se ele criar identidades falsas para atacar a rede, o ataque será completamente sem sentido uma vez que essa identidade falsa nem chegará a ser autenticada.

Os cenários das simulações consideram 5%, 10%, 15% e 20% de nós atacantes. Sendo S o número de certificados emitidos para e por cada nó, os cenários também consideram $S = 5$, $S = 10$, $S = 15$ e $S = 20$. Como mostrado na Figura 6, mesmo com 20% de atacantes na rede, o VKM é capaz de autenticar mais de 40% de cadeias de certificados não comprometidas. Na presença de 5% de atacantes, o VKM-RA é capaz de autenticar corretamente aproximadamente 80% das cadeias de certificados, enquanto PGP-*Like* é completamente vulnerável mesmo com apenas 5% de atacantes [Silva et al. 2008].

O VKM-RA pode tolerar ataques de personificação melhor que o PGP-*Like* devido à estrutura virtual, uma vez que a mesma é altamente redundante e estabelece várias cadeias fixas para autenticação. Se o número de nós comprometidos é pequeno, é possível evitar os nós comprometidos apenas escolhendo a cadeia de certificados de maneira aleatória. Além disso, se for possível implementar um mecanismo de detecção de mau comportamento, então os nós podem explicitamente evitar cadeias de certificados (caminhos virtuais) que contenham nós comprometidos.

A figura 6 também mostra que com 40% de atacantes e 5 certificados emitidos por nó, a possibilidade de escolher uma cadeia de certificados comprometida chega a mais de 80%. Entretanto esse número é reduzido para aproximadamente 60% se o número de certificados emitidos for aumentado para 20. Isso demonstra, portanto, que com um aumento na conectividade da estrutura virtual é possível reduzir ainda mais os efeitos de ataques de personificação.

Os resultados mostram que o VKM-RA é efetivo contra ataques de personificação, desde que a quantidade de atacantes seja pequena e aleatória. Se a quantidade de atacantes for grande ou se os atacantes organizarem um ataque cooperativo à um nó (ou à uma

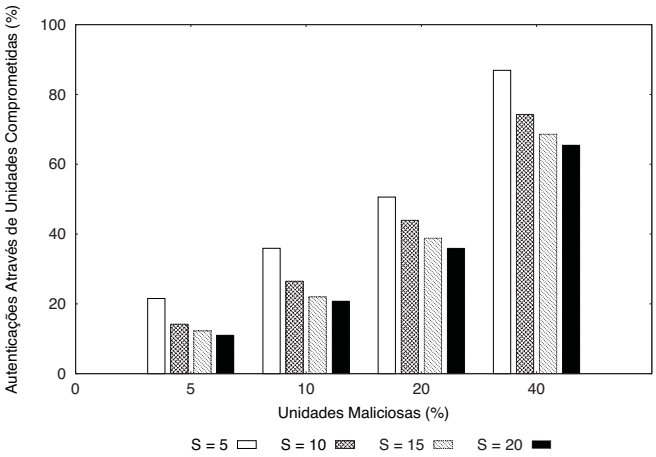

Figura 6. Autenticações feitas através de nós comprometidos.

região) da estrutura virtual, separarando tal nó (ou região) do resto da rede, o ataque pode ser eficaz. Entretanto, isso somente é possível se os atacantes conhecerem a estrutura virtual e ainda realizarem um ataque a vários nós de maneira simultânea, causando a desconexão da estrutura virtual.

4.2. Ataques de falta de cooperação

Para demonstrar que o funcionamento do VKM-PA é similar ao *PGP-Like*, ele foi avaliado em ambientes sem ataques e sob ataques de falta de cooperação e seus resultados comparados com aqueles publicados em [Silva et al. 2008]. O VKM-PA não foi avaliado sob ataques de personificação, pois, igualmente ao *PGP-Like*, ele é completamente vulnerável devido as trocas dos repositórios locais de certificados dos nós.

As simulações consideram 5%, 20%, 40%, 60% e 80% de nós egoístas. Tais nós realizam todas as funções básicas da rede, inclusive emitem certificados. Entretanto, eles não cooperam no mecanismo de troca de certificados, não respondendo aos pedidos de certificados feito por seus vizinhos. As simulações consideraram um tempo de vida de 1500 segundos e os certificados foram emitidos na inicialização da rede.

Seguindo os resultados apresentados em [Čapkun et al. 2003a] e [Silva et al. 2008] para avaliar o *PGP-Like*, duas métricas são usadas nas avaliações: Convergência das Trocas de Certificados (*CE - Certificate Exchange Convergence*) e Alcançabilidade dos Nós (*UR - User Reacheability*). CE e UR medem a completude dos repositórios locais de certificados e a utilidade das trocas de certificados, respectivamente. De acordo com [Silva et al. 2008], CE e UR podem ser definidos como:

$$CE(t) = \frac{\sum CE_i(t)}{|S|} \quad \forall i \in S \quad \text{onde} \tag{3}$$

$$CE_i = \frac{\sum |(K_a \rightsquigarrow K_b) G_i^N \cup G_i|}{\sum |(K_x \rightsquigarrow K_y) \in G|} \quad \forall\, a,b,x,y \in S \tag{4}$$

$$UR(t) = \frac{\sum UR_i(t)}{|S|} \quad \forall i \in S \quad \text{onde} \tag{5}$$

$$UR_i = \frac{\sum |(K_i \rightsquigarrow K_a) \in G_i^N \cup G_i|}{\sum |(K_i \rightsquigarrow K_x) \in G|} \quad \forall a, x \in S \tag{6}$$

As Figuras 7 e 8 ilustram o comportamento do VKM-RA em ambientes sem ataques e sob ataques de falta de cooperação. Em ambos, os resultados do VKM-PA são comparados aos do PGP-*Like*. Como esperado, o aumento do número de atacantes, diminui o valor de CE (Figura 7). Note que em cenários sem atacantes ou com 5% à 60% de atacantes, VKM-PA apresenta um comportamento idêntico ao PGP-*Like*. Apenas em cenários com 80% de atacantes, o desempenho do VKM-PA é cerca de 8% menor que a do PGP-*Like*. Isso se deve à desconexão da estrutura virtual.

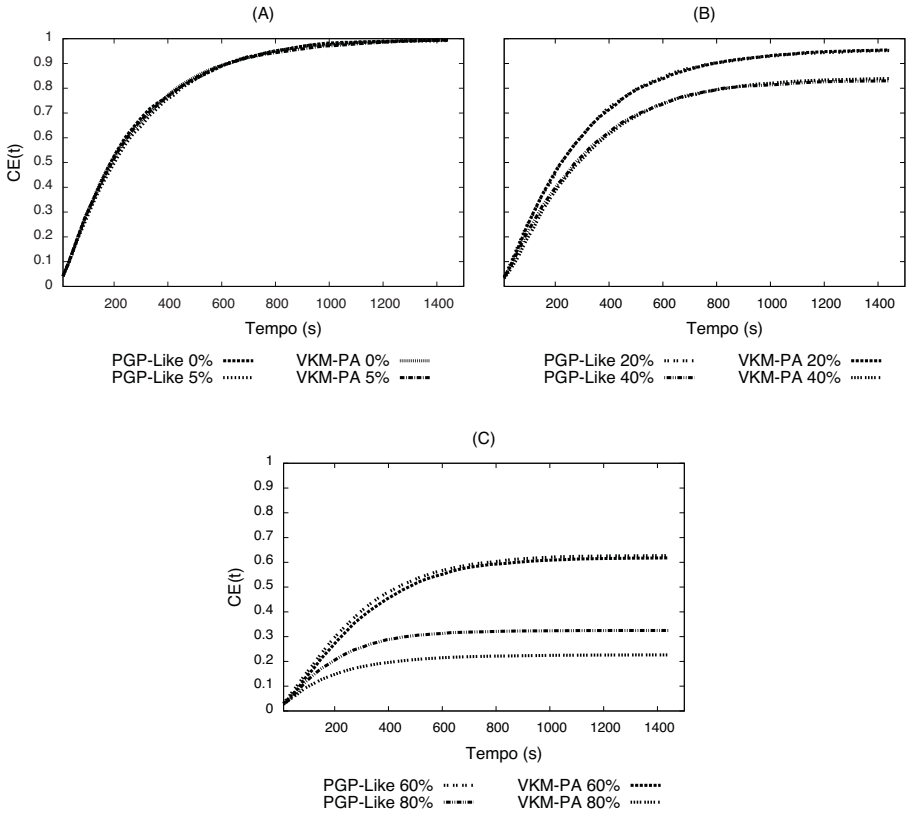

Figura 7. Convergência de trocas de certificados sob ataques de falta de cooperaç ão.

Os resultados do UR para o VKM-PA com até 60% de nós egoístas são também muito similares aos do *PGP-Like* (Figura 8). O UR é quase 100% mesmo na presença de 60% de nós egoístas. Entretanto, quando o número de atacantes sobe para 80%, o desempenho do VKM-PA diminui drasticamente, ficando abaixo de 10%. Novamente, neste caso, a estrutura virtual fica desconexa e o VKM-PA não consegue encontrar cadeias de certificados para realizar as autenticações. Uma maneira de superar este problema é aumentando a conectividade da estrutura virtual. A Figura 9 mostra simulações com 80% de nós egoístas e a conectividade aumentando de 5 até 20 certificados emitidos por nó

(S). Já com $|S = 10|$, VKM-PA tem resultados melhores que o PGP-*Like*, com $|S = 15|$ e $|S = 20|$ a alcançabilidade é maior que 95%. Entretanto, cenários com 80% de nós egoístas não são realistas, eles representam sistemas completamente comprometidos, e podem, inclusive, ser desconsiderados.

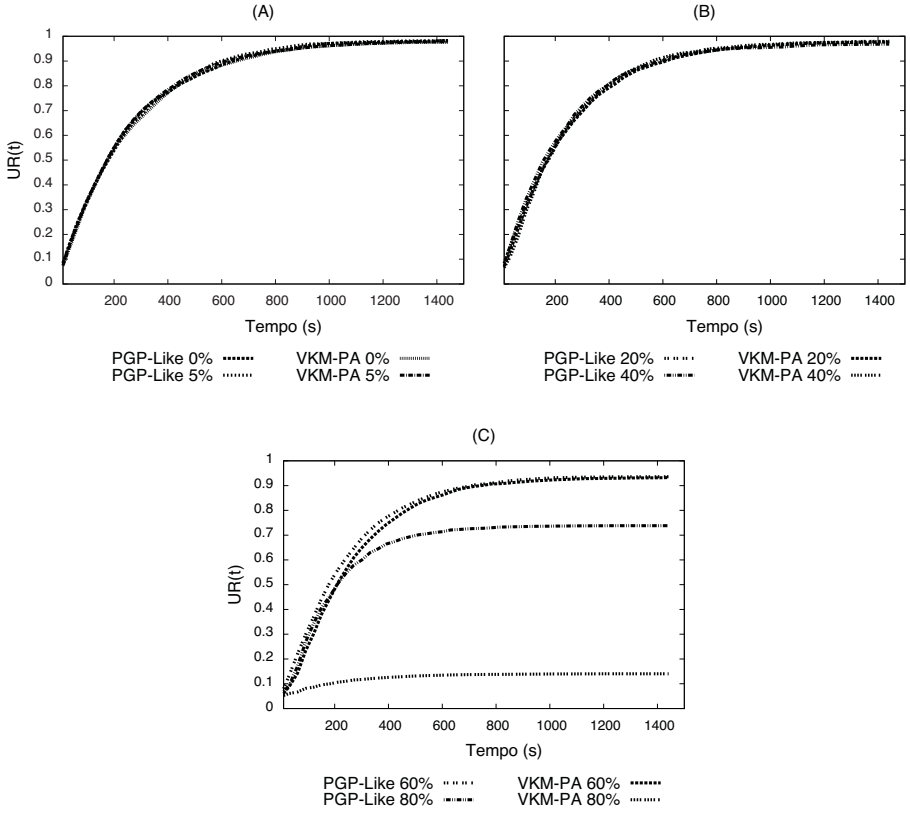

Figura 8. Alcançabilidade dos nós sob ataques de falta de cooperação

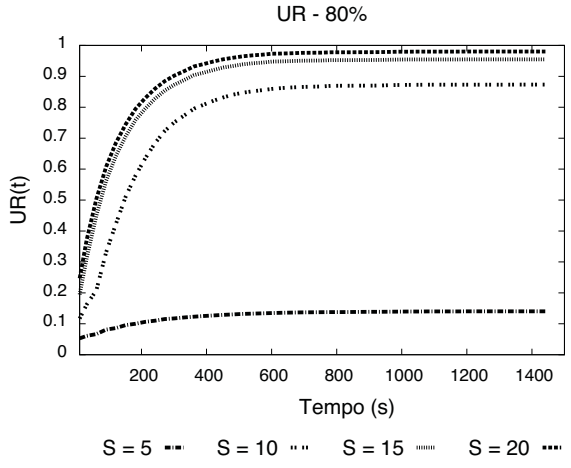

Figura 9. Alcançabilidade dos nós sob 80% de nós egoístas

5. CONCLUSÕES E TRABALHOS FUTUROS

Esquemas de gerenciamento de chaves baseados em cadeias de certificados parecem ser os que melhores se encaixam nos paradigmas das MANETs. O principal esquema baseado em cadeias de certificados é o *Self-Organized Public Key Management System* (*PGP-Like*). Entretanto, como apresentado por [Silva et al. 2008], o *PGP-Like* é altamente vulnerável a ataques de personificação. A funcionalidade do *PGP-Like* é comprometida mesmo com apenas 5% de nós personificados na rede.

Este artigo apresentou um novo esquema de gerenciamento de chaves públicas para MANETs que é resistente a ataques de personificação, o *Virtual Key Management System* (VKM). O VKM faz uso de uma estrutura virtual para indicar a confiança entre os nós e a formação de cadeias de certificados. O VKM é um esquema de gerenciamento muito flexível, podendo ser configurado de duas maneiras diferentes: VKM-RA e VKM-PA. O VKM-RA possui um comportamento restrito. Usando o VKM-RA, os nós seguem as regras da estrutura virtual para emitir certificados e autenticar as chaves públicas. Como mostrado nas simulações, apenas alguns nós atacantes de uma maneira desorganizada não conseguem comprometer o funcionamento da rede. O VKM-RA é capaz de completar corretamente 80% de todas as requisições de autenticação com 5% de nós comprometidos na rede. Mesmo com 20% de atacantes na rede, o VKM-RA consegue autenticar em torno de 50% das cadeias de certificados.

Além disso, o VKM pode se comportar de forma similar ao PGP-*Like*, como VKM-PA, apenas alterando um parâmetro. Para provar isto, o VKM também foi avaliado em ambientes sem qualquer tipo de ataque e em ambientes com ataque de falta de cooperação. Os resultados mostram que sob ataques de falta de cooperação, o VKM-PA tem um desempenho similar ao *PGP-Like*, sendo que os valores de convergência das trocas de certificados (CE) e alcançabilidade dos nós (UR) são praticamente iguais. Os resultados apresentam apenas uma razoável diferença com 80% de nós comprometidos, devido à desconexão da estrutura virtual. Portanto, o VKM pode suprir a necessidade de qualquer usuário com sua capacidade de se comportar de maneiras distintas dinamicamente sem nenhuma reinicialização ou reconfiguração da rede. Os trabalhos futuros incluem o teste do VKM sob diferentes tipos de ataques. Também incluem o desenvolvimento de uma versão segura do protocolo VRP usando o VKM como esquema de gerenciamento de chaves.

Referências

Albini, L., Caruso, A., Chessa, S., and Maestrini, P. (2006). Reliable routing in wireless ad hoc networks: The virtual routing protocol. *Journal of Network and Systems Management*, 14(3):335–358.

Čapkun, S., Buttyán, L., and Hubaux, J.-P. (2003a). Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(1):52–64.

Čapkun, S., Hubaux, J.-P., and Buttyán, L. (2003b). Mobility helps security in ad hoc networks. In *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 46–56, New York, NY, USA.

Čapkun, S., Hubaux, J.-P., and Buttyán, L. (2006). Mobility helps peer-to-peer security. *IEEE Transactions on Mobile Computing*, 5(1):43–51.

Christianson, B. (1996). Why isn't trust transitive. In *Proceedings of the International Workshop on Security Protocols (WSP 1996)*.

Douceur, J. R. (2001). The sybil attack. In *Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS 01)*, pages 251–260.

Eschenauer, L. and Gligor, V. D. (2002). A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM conference on Computer and communications security (CCS 2002)*, pages 41–47, New York, NY, USA.

Hegland, A. M., Winjum, E., Mjolsnes, S. F., Rong, C., Kure, O., and Spilling, P. (2006). A survey of key management in ad hoc networks. *IEEE Communications Surveys*, 08(03):48–66.

Hubaux, J.-P., Buttyán, L., and Čapkun, S. (2001). The quest for security in mobile ad hoc networks. In *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & computing (MobiHoc 2001)*, pages 146–155.

Khalili, A., Katz, J., and Arbaugh, W. A. (2003). Toward secure key distribution in truly ad-hoc networks. In *Proceedings of the 2003 Symposium on Applications and the Internet Workshops (SAINT 2003 Workshops)*, page 342, Washington, DC, USA.

Menezes, A. J., Oorschot, P. C. V., and Vanstone, S. A. (1996). *Handbook of Applied Cryptography*. CRC Press.

Michiardi, P. and Molva, R. (2003). Ad hoc networks security. *ST Journal of System Research*, 4(1).

Ngai, E. C. H. and Lyu, M. R. (2004). Trust- and clustering-based authentication services in mobile ad hoc networks. In *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW 2004)*, pages 582–587, Washington, DC, USA.

Ngai, E. C. H., Lyu, M. R., and Chin, R. T. (2004). An authentication service against dishonest users in mobile ad hoc networks. In *Aerospace Conference 2004*, volume 02, pages 1275–1285, Big Sky, MT.

NS-2 (2007). The network simulator - ns-2.

Robba, A. and Maestrini, P. (2007). Routing in mobile ad-hoc networks: The virtual distance vector protocol. In *Proceedings of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2007)*, pages 1–9.

Silva, E., dos Santos, A. L., Albini, L. C. P., and Lima, M. N. (2008). Quantify misbehavior attacks against the self-organized public key management on manets. In *Proceedings of the International Conference on Security and Cryptography (SECRYPT 2008)*, pages 128–135.

van der Merwe, J., Dawoud, D., and McDonald, S. (2007). A survey on peer-to-peer key management for mobile ad hoc networks. *ACM Computing Survey*, 39(1):1.

Vinogradov, I. M. (1955). *An Introduction to the Theory of Numbers*. Pergamon Press, London & New York.

Zimmermann, P. R. (1995). *The official PGP user's guide*. MIT Press, Cambridge, MA, USA.