Implementação de uma rede mesh segura utilizando o protocolo OLSR aplicada a Smart Grids

Implemention of a secure mesh network using the protocol OLSR applied to Smart Grids

Alexandre Silva Rodrigues, alexandre.rodrigues@redes.ufsm.br
Yagor Santos Duarte
Bruno da Silva Alves
Tiago Antonio Rizzetti
Universidade Federal de Santa Maria, Santa Maria, Rio Grande do Sul

Submetido em 10/05/2016
Revisado em 20/07/2016
Aprovado em 10/08/2016

Resumo: A implementação de uma rede elétrica inteligente requer uma rede de comunicação segura, confiável e bidirecional para interação entre os diversos dispositivos do sistema elétrico. Nesse contexto, a utilização de redes *mesh* apresenta-se como uma alternativa promissora. A principal característica desse tipo de rede é a autoconfiguração da topologia. Para isso, as tabelas de roteamento são trocadas entre os nós. Dessa forma, é necessário impedir que dispositivos não autorizados acessem a rede. Nesses termos, esse trabalho apresenta uma proposta para garantir a autenticidade dos nós participantes de uma rede mesh, através da assinatura das mensagens de controle do protocolo OLSR, via o *plugin Secure* OLSR e uma arquitetura de gerenciamento de chaves simétricas.
Palavras chave: Rede mesh. Protocolo OLSR. Arquitetura de gerenciamento de chaves simétricas. Redes Elétricas Inteligentes.

Abstract: The implementation of a Smart Grid requires a network secure, reliable and of two-way communication for interaction between the several devices in the electric system. In this context, the use of mesh networks is presented as a promising alternative. The main feature of this type of network is the auto configuration of the topology. For this, the routing tables are exchanged between the nodes. Thus, it is necessary prevent unauthorized devices from accessing the network. In these terms, this work presents a proposal to ensure the authenticity of the participating nodes of a mesh network, through signing the control messages of OLSR protocol, via Secure OLSR plugin and a symmetric key management architecture.
Keywords: Mesh Network. OLSR Protocol. Symmetric Key Management architecture. Smart Grids.

Introdução

As redes elétricas inteligentes (*Smart Grid*) destacam-se por utilizarem tecnologias digitais para monitorar e controlar os dispositivos ativos do sistema elétrico. Para que isso seja possível, há necessidade de estabelecer uma rede de comunicação bidirecional entre os diversos dispositivos presentes na rede de energia com o sistema supervisório utilizado para gerenciá-la.

O sistema elétrico de potência (SEP) apresenta uma diversidade de necessidades de rede. Em alguns sistemas, bem delimitados e geograficamente concisos, é possível utilizar comunicação através de meios cabeados confiáveis, por exemplo, fibra óptica. Sistemas de geração são um exemplo clássico onde isso é possível. No entanto, em sistemas de distribuição, em função da grande quantidade de dispositivos e de sua alta dispersão geográfica, a implementação dessa rede de comunicação torna-se uma tarefa mais complexa (GTREI, 2010).

Nesse contexto, diferentes tecnologias podem ser utilizadas para prover comunicação em determinados segmentos da rede de comunicação. Uma dessas tecnologias consiste nas redes *mesh*, em função da sua natureza dinâmica que permite a inserção, reconfiguração e saída de dispositivos da rede de forma frequente.

No entanto, a segurança da comunicação é um fator que pode trazer um grande impacto para uma rede de comunicação utilizada em uma *Smart Grid* (LOPES, 2012). Em função da criticidade do sistema, deve-se garantir que somente dispositivos autorizados efetuem as transmissões e recepção de dados nessa rede de comunicação (GTREI, 2010). Essa é uma prerrogativa muitas vezes negligenciada na concepção de protocolos para redes *mesh* (JUNIOR, 2003).

Objetivos

O presente trabalho tem como objetivo, apresentar uma topologia, baseada em uma rede *mesh*, para realizar a comunicação segura em uma *Smart Grid*, entre concentradores e uma central de controle. Para possibilitar uma comunicação segura para as mensagens de controle de uma rede *mesh*, nesse trabalho, os seguintes objetivos específicos serão abordados:

- implementar uma rede *mesh*, onde cada nó representa um elemento ativo no sistema elétrico;
- analisar o protocolo e as tabelas de roteamento em cada nó;
- simular a inserção de falsos nós na rede;
- implementar uma forma de comunicação segura para a troca de mensagens de controle na rede *mesh*;

Smart Grids

A energia elétrica é utilizada para os mais diversos fins, seja nas residências quanto nas indústrias. Em situações onde o seu fornecimento é interrompido, evidencia-se o quanto ela é importante e necessita de sistemas capazes de automatizar o processo de restabelecimento da mesma. Além disso, a relação entre as concessionárias de energia elétrica e seus clientes ainda é restrita. Para resolver essas questões, diversas tecnologias têm surgido para facilitar o processo de distribuição de energia elétrica.

Nesse contexto, destaca-se o conceito de redes elétricas inteligentes (*Smart Grids*), que apresentam uma série de vantagens em relação ao sistema convencional de energia elétrica, como por exemplo: interação entre dispositivos ativos no sistema elétrico de potência em tempo real, capacidade de auto recuperação em casos de falhas no sistema, automatização e melhor gerenciamento dos processos de geração, distribuição e transporte da energia elétrica (WANG, 2011).

De acordo com RAMOS (2012), uma rede elétrica inteligente destaca-se por utilizar as tecnologias da informação, para facilitar a administração e gerenciamento da rede elétrica convencional. As *Smart Grid* visam modernizar a rede elétrica, que ao longo dos anos, apresentou uma discreta evolução. A automação desse sistema possibilitará ações, em tempo real, em equipamentos presentes na geração até a distribuição da energia elétrica.

Uma das primeiras ações para tornar a rede elétrica inteligente é a utilização de *Smart Meters* (medidores inteligentes), que são capazes de comunicar-se com outros equipamentos, como por exemplo, enviar/receber dados para a concessionária de energia (LOPES, 2012). Outro aspecto importante, quando se trata de *Smart Grid*, é contingência em casos de falhas

na rede de distribuição. Nesse contexto, aplica-se o conceito de *Self-Healing*, que pode ser visto como uma reconfiguração automática. Para isso, é realizado o monitoramento e análise da rede, buscando possíveis falhas. Essa funcionalidade permite identificar a falha e o local da ocorrência e assim, facilitar o processo de restabelecimento de energia para os clientes.

Rede de comunicação de dados em uma *Smart Grid*

A rede a ser utilizada em uma *Smart Grid* deve permitir uma comunicação bidirecional entre os consumidores e as empresas que atuam na geração e distribuição da energia elétrica. Além disso, é importante ressaltar que essa comunicação necessita de um elevado índice de disponibilidade e confiabilidade, devido ao alto grau de criticidade das informações que nela podem trafegar. Nesse aspecto, podem ser utilizados diferentes tipos de tecnologias para levar informações de um ponto a outro (WANG, 2011) (EKANAYAKE, 2012).

Nesses termos, a alteração ou falsificação de uma informação pode causar grandes prejuízos ao sistema ou interrupção de importantes serviços oferecidos por ele. Dessa forma, em relação às informações trafegadas, os seguintes critérios necessitam ser observados (EKANAYAKE, 2012):

a) Confidencialidade: relaciona-se com a privacidade de uma informação. Dessa forma, apenas o emissor e o receptor têm acesso à mesma;

b) Autenticidade: refere-se à identidade do emissor de uma mensagem, ou seja, o receptor certifica-se que uma informação recebida não foi enviada por um impostor;

c) Integridade dos dados: garantia que uma informação não sofreu nenhuma modificação, ou seja, a informação que chegou ao receptor é exatamente igual à que foi enviada pelo emissor.

Para prover a comunicação entre os dispositivos ativos em uma *Smart Grid* é necessária a utilização de um padrão de rede que permita a interação entre os diversos dispositivos e protocolos de comunicação. Além disso, é necessário que todos os dispositivos sejam endereçados de forma única na rede. Para isso, a utilização de redes baseadas em IP (*Internet Protocol*) destaca-se, em razão de sua consolidada utilização para as mais diversas aplicações.

Em razão do grande número de dispositivos e sua disposição geográfica, a preocupação com a topologia da rede é essencial. Esse aspecto pode influenciar diretamente em critérios críticos para este tipo de comunicação, tais como, latência, disponibilidade e segurança. Uma alternativa promissora para esse paradigma consiste em dividir esta rede de comunicação em sub-redes menores interligadas por concentradores, agindo como coletores de informações (GTREI, 2010). Devido à alta densidade de dispositivos e o constante crescimento de nós que a rede pode apresentar, as tecnologias de comunicação sem fios, através de redes *mesh*, são apresentados como uma alternativa interessante.

Redes Mesh

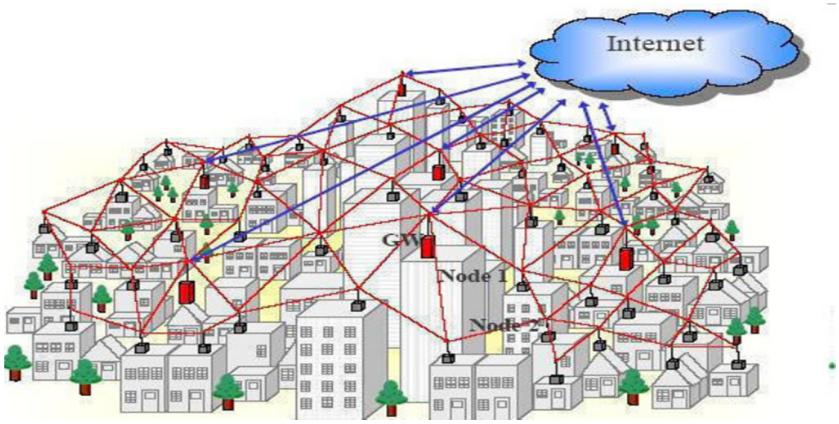
Segundo ABELÉM (2007), uma rede *mesh* pode ser vista como uma rede com topologia dinâmica, variável e que pode ser facilmente expandida. Essa é constituída por diversos equipamentos (nós) interligados. A comunicação entre esses é realizada através do padrão IEEE 802.11s, formando uma grande rede sem fio (malha), utilizando múltiplos saltos para transmitir dados. Nesse caso, cada nó pode atuar como roteador e prover acesso a uma rede externa (nó funcionará como gateway para os demais nós) (CARDOSO, 2012).

Dessa forma, o sinal de um nó é replicado pelos demais nós pertencentes a uma rede, possibilitando a escolha do melhor caminho ou rotas alternativas para encaminhar um pacote (MACHADO, 2013). Assim, uma rede mesh pode ser utilizada em regiões geográficas extensas ou que apresentam determinada dificuldade de acesso. Um exemplo disso, é em regiões que apresentam montanhas ou prédios que dificultam a propagação do sinal sem fio de uma rede estruturada. Nesse caso, é possível criar rotas alternativas entre um nó emissor e o seu destino (ZHANG, 2006).

Outro aspecto importante, em relação a esse tipo de rede, é autoconfiguração da topologia e descoberta dos nós ativos na rede. Dessa forma, um nó consegue ingressar na rede automaticamente, sem a necessidade de uma configuração manual. Além disso, se um nó apresentar uma falha e ficar indisponível, a rede cria rotas alternativas automaticamente sem afetar a disponibilidade da mesma. A Figura 1 apresenta a topologia de uma rede *mesh*.

Nela podemos visualizar a forma como os nós são interligados, possibilitando que diferentes rotas possam ser utilizadas para que um determinado nó possa acessar a rede externa (Internet), através de múltiplos *gateways*, que aumentam a disponibilidade da rede.

Figura 1. Topologia de uma rede *mesh*.



Fonte: (ZUCCHI, 2006).

Protocolos de roteamento para redes *mesh* e segurança nativa

Em razão do crescente interesse de estudo e implementação de redes mesh, diversos protocolos de roteamento surgiram. Esses protocolos podem ser divididos em três grupos (BOWITZ, 2011):

a) Protocolos pró-ativos: realizam atualização constante da rede, através da troca mensagens de controle. Dessa forma, esse protocolo reconhece uma possível modificação na topologia da rede, como por exemplo, a inclusão ou a indisponibilidade de um nó;

b) Protocolos reativos: criam rotas apenas quando um nó emissor precisar realizar a comunicação com um nó destino. Dessa forma, as mudanças na topologia da rede demoram um maior tempo para ser detectada;

c) Protocolos híbridos: mesclam as características dos protocolos descritos anteriormente.

Tipos de ataques em redes *mesh*

Em razão da utilização de meios de comunicação sem fio e da falta de mecanismos de segurança nos protocolos de roteamento, a segurança é um dos

principais problemas para tornar viável a implementação de uma rede *mesh*. Essa pode ser exposta a uma série de ataques. Esses ataques podem ser praticados por atacantes externos, para vasculhar e modificar as informações trafegadas na rede. Além disso, um nó participante da rede pode agir de forma maliciosa, podendo ocasionar consequências mais graves (FERNANDES, 2006).

Esses ataques podem ser inativos, quando o objetivo é apenas vasculhar o trafego de dados de uma rede ou ativos, quando o objetivo é injetar, modificar ou descartar os dados trafegados na rede (SEN, 2013). Para ter acesso a esses dados, o atacante pode explorar vulnerabilidades nos protocolos de roteamento utilizados na rede. Em relação às tabelas de roteamento, o atacante pode utilizar a ausência de mecanismos de autenticação em uma rede para divulgar falsas informações sobre a topologia da rede.

Nesse contexto, a inserção de falsos pacotes, nos quais são inclusas mensagens de controle, na rede pode ocasionar um estouro na tabela de roteamento de um dispositivo ou alteração nas rotas verdadeiras, para um dispositivo intermediário, entre o emissor e o receptor de um dado. Dessa forma, o atacante pode afetar a confidencialidade da rede e obter informações privadas. Além disso, esses dados podem ser modificados antes de chegarem ao seu destino (FERNANDES, 2006).

Outro aspecto importante é a disponibilidade da rede que pode ser afetada, através da criação de buracos negros na rede. Nesse aspecto, uma falsa rota é divulgada na rede. A origem dessa rota pode ser um nó malicioso que descarta todos os pacotes recebidos (SEN, 2013).

Além disso, o atacante pode utilizar complexos sistemas computacionais para indisponibilizar os serviços ou recursos de um dispositivo ou rede. Nesse contexto, uma técnica denominada DoS é utilizada para realizar essa ação. Esse tipo de ataque pode ocasionar sérias consequências em uma rede que necessita de um nível alto de disponibilidade.

Protocolo OLSR

De acordo com SILVA (2011), o protocolo OLSR (*Optimized Link State Routing Protocol*) foi criado para ser utilizado em grandes redes Ad Hoc,

calculando e mantendo rotas para todos os dispositivos de uma rede construída sob uma topologia em malha.

Por ser um protocolo pró-ativo, o OLSR tem como característica principal, a atualização constante das tabelas de roteamento de todos os nós participantes da rede. Dessa forma, qualquer modificação na topologia da rede é detectada automaticamente. Nesse contexto, o protocolo OLSR utiliza mensagens de controle para realizar a descoberta de nós na rede e atualização da topologia (SILVA, 2011). Essas mensagens são enviadas em broadcast através da porta UDP 698 (CLAUSEN, 2003).

De acordo com CLAUSEN (2003) e FERNANDES (2006), as seguintes mensagens são essenciais para o funcionamento do protocolo OLSR:

a) HELLO: utilizada para realizar a escolha de um MPR e a descoberta de enlaces e vizinhos. Essa mensagem é enviada apenas para os vizinhos e não deve ser encaminhada para os vizinhos distantes a mais de um salto e apresenta os nós com os quais um nó possui uma comunicação.

b) TC: é utilizada para realizar o controle da topologia da rede. Essa mensagem é composta por uma lista de nós que selecionaram o nó emissor dessa mensagem como MPR e enviada para toda a rede. Dessa forma, é possível que um nó não envie essa mensagem, em razão não ter sido escolhido como MPR. As informações recebidas através dessa mensagem são armazenadas na tabela de roteamento de cada nó, a qual é utilizada para realizar o cálculo de rota para enviar um pacote a um determinado destino.

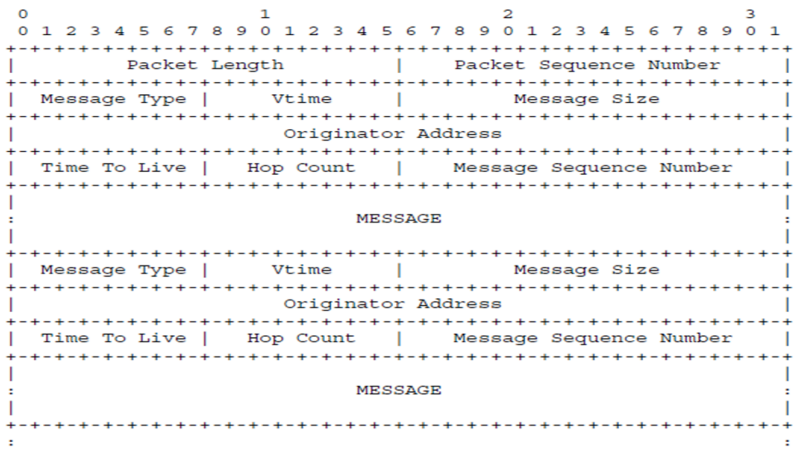
c) MID: essa mensagem é enviada para toda a rede e utilizada para declarar múltiplas interfaces em um nó. Através dessa mensagem, é possível associar diversas interfaces de um dispositivo no cálculo de rotas.

d) HNA: contém informações sobre os anúncios das redes de um nó, ou seja, um nó apresenta-se como gateway para o acesso de uma determinada rede.

Conforme CLAUSEN (2003), o protocolo OLSR utiliza um formato de pacote unificado para todas as informações relacionadas ao protocolo. Dessa forma, cada pacote pode encapsular mais de uma mensagem. Essas mensagens compartilham um formato de cabeçalho comum. A Figura 2

apresenta o formato de um pacote OLSR. Nela podemos visualizar o cabeçalho do pacote e o encapsulamento de mensagens.

Figura 2. Formato dos Pacotes OLSR.



Fonte: (CLAUSEN, 2003).

Secure OLSR

O protocolo OLSR possui um *plugin* de segurança denominado Secure OLSR (SOLSR), que utiliza uma chave simétrica de 128 bits para assinar as mensagens de controle do protocolo. Essa chave deve ser conhecida por todos os nós da rede. As mensagens são assinadas a cada salto, ou seja, não é possível estabelecer uma autenticação entre o emissor e destinatário. Dessa forma, é necessário que um nó confie em seus vizinhos (HAFSLUND, 2004).

Em virtude desse *plugin* utilizar uma chave compartilhada entre todos os dispositivos, caso essa seja descoberta ou compartilhada com um nó malicioso, a segurança de toda rede pode ser comprometida. Dessa forma, um falso nó consegue ingressar na rede, enquanto essa chave for válida. Nesse caso, a chave precisa ser modificada e atualizada manualmente em todos dispositivos.

Proposta de uma arquitetura para gerenciamento de chaves simétricas

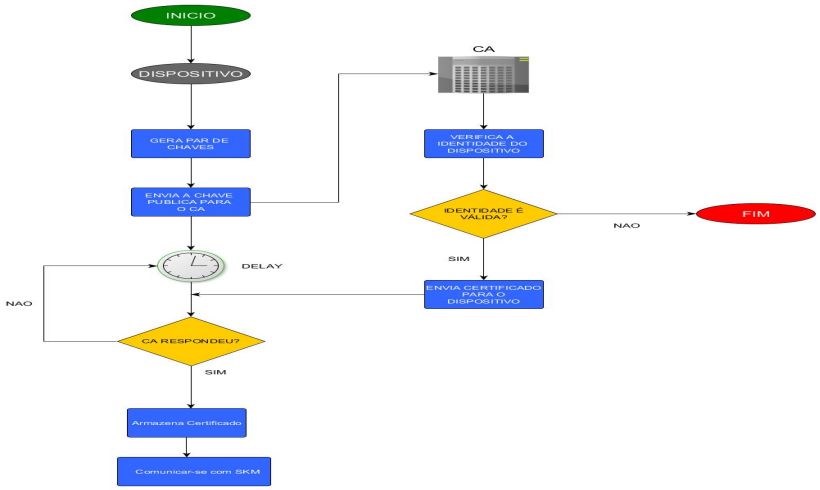
Com o objetivo de tornar possível a utilização de redes *mesh* em aplicações críticas (*Smart Grids*, por exemplo) e devido às vulnerabilidades inerentes a esta forma de comunicação, esse trabalho apresenta uma proposta que visa agregar segurança e confiabilidade a esse tipo de rede. Para isso foi

desenvolvida uma arquitetura para realizar o gerenciamento de chaves simétricas, as quais são utilizadas pelo *plugin Secure* OLSR.

Nesse contexto, essa arquitetura baseia-se em uma entidade, denominada *Symmetric Key Manager* (SKM), responsável por gerenciar e atualizar a chave simétrica utilizada pelos dispositivos que compõem uma rede *mesh*. Além disso, é utilizada uma entidade conhecida como *Certificate Authority* (CA), a qual é responsável por emitir certificados padrão X509 para dispositivos autorizados. Esses certificados são utilizados na comunicação entre dispositivos e a entidade responsável pelo gerenciamento da chave simétrica (SKM), para fins de comprovação de identidade e autenticidade entre ambos.

Para que um dispositivo possa obter acesso a chave simétrica, utilizada pelo *plugin Secure* OLSR, ele precisa estar devidamente autorizado pelo CA. Para isso, ele deve gerar um par de chave assimétrica (chave pública e chave privada) e solicitar um certificado X509 ao CA. Nessa solicitação, o dispositivo deve enviar sua chave pública e suas credenciais (configuradas manualmente no dispositivo e conhecidas pelo CA). Após a confirmação da identidade de um dispositivo solicitante de um certificado, o CA emite e envia o certificado para o requisitante. Basicamente, esse certificado é gerado da seguinte forma: o CA gera uma *hash* da chave pública do dispositivo, criptografa essa *hash* com a sua chave privada, adiciona a *hash* criptografada a chave pública do dispositivo. A Figura 3 ilustra esse processo.

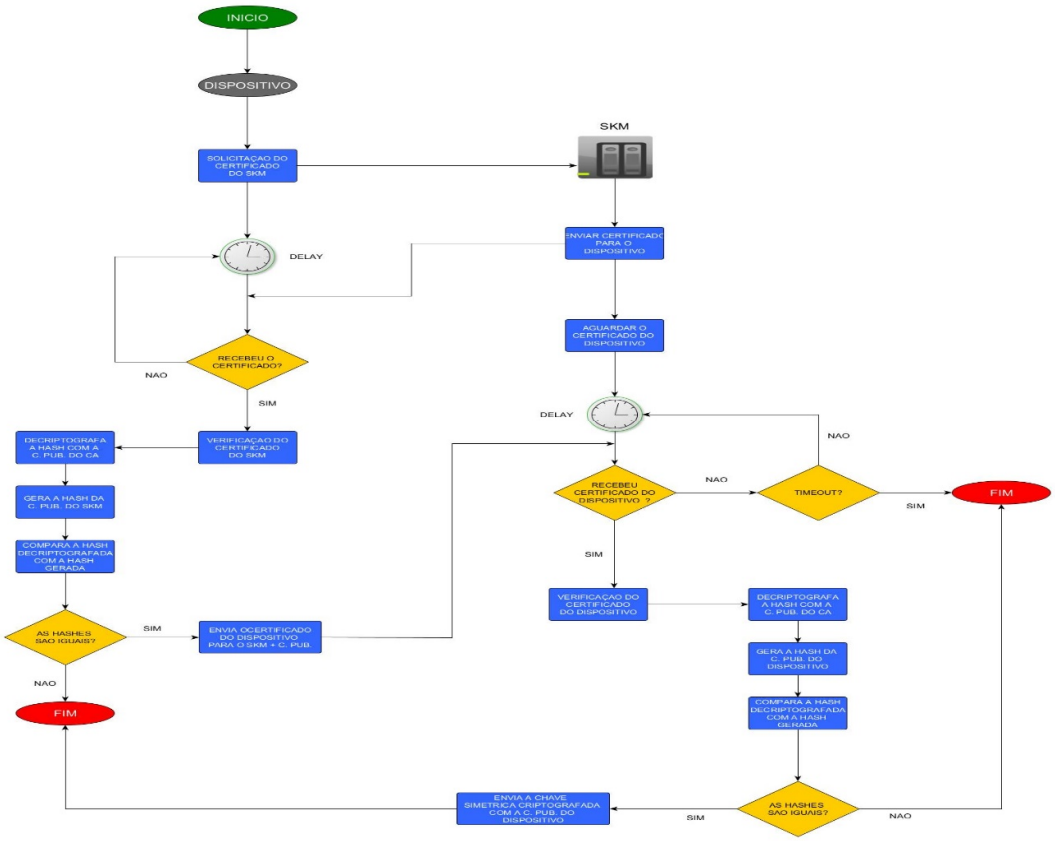
Figura 3. Comunicação entre dispositivo e CA.



Fonte: Acervo Pessoal.

Após receber o certificado, o dispositivo deve realizar a sua primeira comunicação com o SKM, a fim de comprovar que é um dispositivo autorizado a participar da rede e, por conseguinte, obter a chave simétrica utilizada pelos demais dispositivos participantes da rede *mesh*. Para isso, ele realiza a solicitação do certificado do SKM e verifica a sua autenticidade, ou seja, ele compara se o certificado é válido. Em caso de sucesso, ele envia o seu certificado e sua chave pública para o SKM, que também realiza a verificação de autenticidade do certificado recebido. O SKM após confirmar que a solicitação é proveniente de um dispositivo autorizado, envia a chave simétrica para o solicitante. Essa mensagem é criptografada com a chave pública do dispositivo. Dessa forma, garante-se que apenas o dispositivo poderá realizar a decriptografia dessa mensagem, utilizando a sua chave privada. A Figura 4 apresenta esse processo de comunicação entre um dispositivo e o SKM e, por conseguinte, o envio da chave simétrica para o dispositivo solicitante.

Figura 4. Comunicação entre dispositivo e SKM.



Fonte: Acervo Pessoal.

Conforme podemos visualizar na Figura 4, se um atacante tentar conseguir obter acesso a chave simétrica utilizada na rede *mesh,* interceptando uma comunicação para obter um certificado válido, ele não conseguirá obter a chave, visto que ele precisaria conhecer a chave privada de um dispositivo.

Após o recebimento da chave simétrica, o dispositivo está apto a ingressar na rede *mesh*, ou seja, ele já pode utilizar essa chave para assinar as mensagens de controle do protocolo OLSR, através do *plugin Secure* OLSR.

Além de realizar a distribuição da chave simétrica para novos dispositivos, o SKM é responsável por realizar a atualização dessa chave. Ou seja, após determinado período, uma nova chave é gerada e informada para todos os dispositivos que já possuem a chave antiga. Para isso, o SKM possui um algoritmo que gera chaves de 128 bits aleatórias.

Após gerar uma nova chave, o SKM envia uma mensagem em *multicast* (ou seja, para o grupo de dispositivos que já possuem a chave antiga), informando que uma nova chave está disponível). Quando um dispositivo receber essa mensagem e inicia o processo de obtenção da nova chave simétrica. Para isso, ele realiza o processo apresentado na Figura 4, ou seja:

- Primeiramente, ele realiza o estabelecimento de uma sessão de comunicação (troca de certificados);
- SKM envia a nova chave criptografada com a chave pública do dispositivo solicitante;
- Dispositivo utiliza a sua chave privada para decriptografar a mensagem recebida e atualiza a sua chave simétrica, ou seja, descarta a chave anterior e começa a utilizar a chave recebida para assinar as suas mensagens de controle (através do *plugin Secure* OLSR).

Dessa forma, garante-se que apenas dispositivos autorizados receberão a nova chave simétrica. Caso ela fosse enviada para todos dispositivos de uma única vez (criptografada com a chave simétrica anterior), a segurança poderia ser comprometida, uma vez que um atacante que teve acesso a chave antiga, sempre conseguiria obter a nova chave.

Outro aspecto importante na arquitetura apresentada nesse trabalho é verificação da validade da chave simétrica que um dispositivo possui. Para

demonstrar essa funcionalidade, usaremos o seguinte cenário: um determinado dispositivo apresentou uma falha e não recebeu uma nova simétrica enviada pelo SKM. Dessa forma, ele não possui uma chave simétrica válida e teria que aguardar que uma nova chave fosse gerada para recebe-la. Para resolver esse problema, um dispositivo pode enviar uma solicitação para o SKM para verificar se a chave simétrica que ele possui é válida.

Nesses termos, o dispositivo gera uma *hash* de chave simétrica que possui e envia para o SKM (após o estabelecimento de sessão de comunicação). Ao receber, o SKM gera uma *hash* da chave atual e verifica se elas são iguais. Caso não, isso significa que o dispositivo não possui a chave simétrica atual, o SKM envia a chave atual para o dispositivo, para que ele possa estar sincronizado com os demais dispositivos da rede *mesh*.

Testes e Resultados

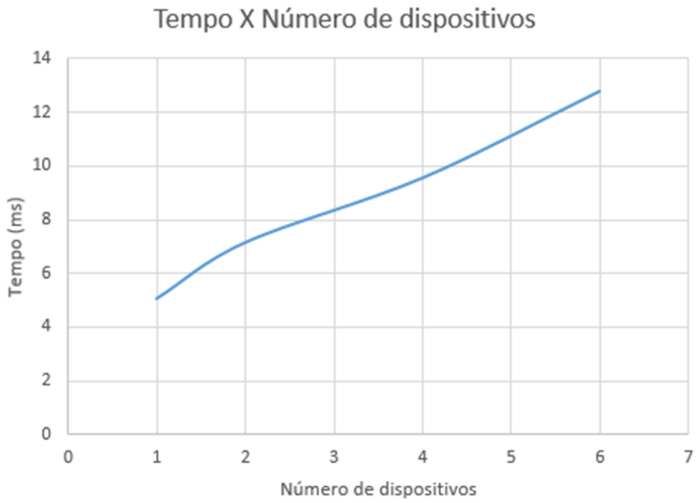
Antes de avaliar o desempenho do protocolo OLSR para implementar uma rede *mesh* foi realizado um teste para verificar o funcionamento do gerenciador de chave simétrica e o tempo necessário para que os dispositivos recebam uma chave após ela ser atualizada. Para isso, utilizou-se dispositivos com sistema Operacional Linux. Os códigos foram desenvolvidos na linguagem de programação C e compilados com o compilador GCC. Para realizar a geração de chaves assimétricas, emissão/verificação de certificados X509 e criptografia/decriptografia de mensagens utilizou-se a biblioteca *Openssl*.

A comunicação entre cada parte da arquitetura proposta (CA, SKM e dispositivos) foi realizada por meio de sockets TCP. Para isso, utilizou-se uma porta específica e um protocolo de comunicação, no qual definiu-se o conjunto e sequência de mensagens a ser utilizado.

Nesses termos, esse teste foi realizado em cenários diferentes: com um, dois, quatro, seis e dispositivos. Em cada um desses cenários, o SKM gerou uma nova chave e enviou uma mensagem em *multicast* para os dispositivos. Dessa forma, o tempo decorrido entre a geração da nova chave simétrica e a atualização da chave em todos dispositivos pode ser visualizado na Figura 5, na qual podemos observar que a arquitetura proposta apresentou um desempenho com tendência linear. Nesses termos, o tempo necessário para a convergência

da rede, ou seja, para que todos os dispositivos tenhas suas chaves simétricas atualizadas está relacionado ao tamanho da rede (quantidade de dispositivos), visto que, o SKM deve atender a todas requisições dos dispositivos simultaneamente.

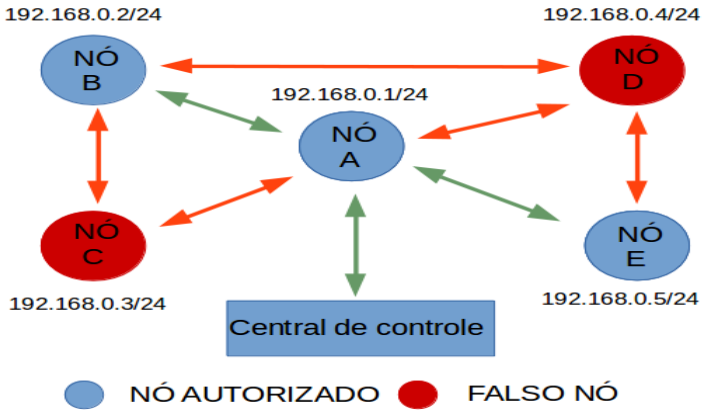
Figura 5. Tempo necessário para atualizar a chave em todos dispositivos



Fonte: Acervo Pessoal

Para verificar a segurança em uma rede *mesh*, foi implementado um cenário de testes, onde são representados os equipamentos ativos em uma *Smart Grid*. A Figura 6, ilustra a topologia utilizada e os endereços IP de cada dispositivo.

Figura 6. Cenário de testes.



Fonte: Acervo Pessoal.

De acordo com a Figura 6, para simplificar o cenário de testes, será utilizado um nó para representar um concentrador de dados (nó A), o qual será responsável por enviar os dados para a central de controle. Os demais nós (B e E) representarão os dispositivos instalados na residência do cliente, por exemplo, um medidor inteligente. Além disso, são adicionados falsos nós que tentarão obter acesso a rede (C e D).

Com base nesse cenário, os testes foram realizados foram divididos em duas etapas, onde utilizou-se:

a) Protocolo OLSR nativo;
b) Protocolo OLSR e o *plugin Secure* OLSR (SOLSR).

O primeiro passo para a realização dos testes, é a configuração dos dispositivos. Após as devidas configurações nos dispositivos, é necessário realizar as configurações exigidas pelo protocolo OLSR. Para isso, foi editado o arquivo "/etc/olsr/olsrd.conf", o qual é criado durante a instalação do *daemon "olsrd*".

Com base na Figura 6, o primeiro cenário de teste visou analisar a segurança nativa do protocolo OLSR, ou seja, se o protocolo OLSR permite que falsos dispositivos, que possuam as mesmas configurações dos nós participantes da rede, insiram informações na rede. Após a configuração de cada dispositivo, o protocolo OLSR foi iniciado em ambos. Após as trocas de mensagens de controle entre os dispositivos, podemos visualizar a tabela de roteamento do nó A, fornecida pelo *daemon "olsrd"* na Figura 7.

Figura 7. Tabela de roteamento do nó A.

```
       *** olsr.org -  0.6.6.1-git_0000000-hash_41d32a614ae55e881b7c0456c8e3ed54 (2013-10-26 05:10
:52 on toyol) ***

--- 16:41:54.898958 ------------------------------------------------------ LINKS

IP address       hyst         LQ       ETX
192.168.0.3      0.000  1.000/1.000    1.000
192.168.0.2      0.000  1.000/0.000    INFINITE
192.168.0.5      0.000  1.000/0.000    INFINITE
192.168.0.4      0.000  1.000/1.000    1.000

--- 16:41:54.89 ---------------------------------------------- NEIGHBORS

     IP address  LQ     NLQ    SYM   MPR   MPRS  will
192.168.0.4      0.000  YES    YES   YES   3
192.168.0.3      0.000  YES    YES   YES   3

--- 16:41:54.899048 ---------------------- TWO-HOP NEIGHBORS

IP addr (2-hop)  IP addr (1-hop)  Total cost
192.168.0.5      192.168.0.4      2.000
                 192.168.0.2      2.128
                 192.168.0.3      2.362
192.168.0.4      192.168.0.5      2.000
                 192.168.0.2      2.000
                 192.168.0.3      2.063
192.168.0.3      192.168.0.5      2.362
                 192.168.0.2      2.063
                 192.168.0.4      2.063
192.168.0.2      192.168.0.4      2.000
                 192.168.0.3      2.058
                 192.168.0.5      2.128
```

Fonte: Acervo Pessoal

Conforme a Figura 7, os dispositivos C e D conseguiram realizar alterações na topologia da rede, ou seja, eles tiveram suas mensagens de controle aceitas e constam nas rotas para a comunicação entre os nós verdadeiros da rede *mesh* implementada. Um exemplo disso é a comunicação entre os nós A e E. Além do link direto (apenas um salto) entre eles, ela pode ser realizada por links de dois saltos, passando pelos seguintes nós: B, C e D. Dessa forma, evidencia-se a necessidade de utilizar um mecanismo de autenticação entre os dispositivos, quando para prover uma rede segura.

No segundo cenário de testes utilizou-se o *plugin Secure* OLSR. Para utilizar esse *plugin*, é necessário editar o arquivo de configuração do *daemon* do protocolo OLSR ("olsrd"), inserindo os seguintes parâmetros: versão do *plugin* e localização do arquivo que contém a chave simétrica a ser utilizada. Utilizando o cenário apresentado na Figura 6, os nós A, B e E utilizam a mesma chave para assinar as mensagens de controle do protocolo OLSR, o nó C utiliza uma chave diferente e o nó D, não acrescenta nenhuma assinatura em seus pacotes. Dessa forma, após a inicialização do *daemon "olsrd*" em todos os dispositivos, apenas os nós que utilizam a mesma chave, chamada de chave de grupo, podem fazer parte da topologia da rede.

Nesse caso, a Figura 8 apresenta a tabela de roteamento do nó A. Nela podemos visualizar os vizinhos descobertos por esse nó e as rotas que podem ser utilizadas para comunicação com outros nós da rede.

Figura 8: Tabela de roteamento do nó A, utilizando o *plugin Secure* OLSR.

```
       *** olsr.org - 0.6.8-git_0000000-hash_f90f6d7f8b957fff3eea9cf8ba30a665 (2
015-06-17 17:49:32 on alexandre-note) ***

--- 18:27:54.438184 ------------------------------------------------- LINKS

IP address       hyst         LQ       ETX
192.168.0.5      0.000  1.000/0.854    1.169
192.168.0.2      0.000  1.000/1.000    1.000

--- 18:27:54.438205 ---------------------------------------------- NEIGHBORS

     IP address Hyst    LQ      ETX     SYM   MPR   MPRS  will
192.168.0.5     0.000   1.000/0.854     1.169   YES   NO    NO    3
192.168.0.2     0.000   1.000/1.000     1.000   YES   NO    NO    3

--- 18:27:54.438225 ----------------------- TWO-HOP NEIGHBORS

IP addr (2-hop)  IP addr (1-hop)  Total cost
192.168.0.5      192.168.0.2      2.128
192.168.0.2      192.168.0.5      2.528
```

Fonte: Acervo Pessoal.

De acordo com a Figura 8, apenas os nós autorizados (que utilizaram a chave de grupo) conseguiram acessar a rede, ou seja, tiveram suas mensagens de controle aceitas, conforme pode ser visualizado na Figura 9, onde a assinatura (*hash*) adicionada em um pacote enviado pelo nó B teve sua autenticidade comprovada e foi aceita por A.

Figura 9. Pacote enviado por B, utilizando o *plugin Secure* OLSR.

```
Recevied hash:
 187  83   6 219 190 137 175 161 152  60 120 253 110 191  68  66
Calculated hash:
 187  83   6 219 190 137 175 161 152  60 120 253 110 191  68  66
[ENC]Received timestamp 1434576463 diff: 12
[ENC]Packet from 192.168.0.2 OK size 60
```

Fonte: Acervo Pessoal.

Dessa forma, o nó A possui um link de um salto com os nós B e E. Além disso, o nó A consegue se comunicar com o nó E, passando pelo nó B, e com o nó B, através do nó E.

As mensagens de controle enviadas pelos nós que tentaram obter acesso a rede foram descartas, conforme pode ser visualizado na Figura 10 (mensagem do nó C) e na Figura 11 (mensagem do nó D).

Figura 10. Pacote do nó C.

```
Recevied hash:
 158  44  37 207 250  18  64 124 241  12 215  25  28 226  93  48
Calculated hash:
 244  81 147 243 221  39  93 217 254   0 159 166 230 171  91 164
[ENC]Signature missmatch
[ENC]Rejecting packet from 192.168.0.3
```

Fonte: Acervo Pessoal.

Figura 11. Pacote do nó D.

```
[ENC]Packet not sane!
[ENC]Rejecting packet from 192.168.0.4
```

Fonte: Acervo Pessoal.

Conforme pode ser visualizado nas Figuras 10 e 11, a utilização desse *plugin*, impossibilita que dispositivos não autorizados ingressem na rede.

Conclusões

De acordo com os resultados apresentados no primeiro cenário de testes (protocolo OLSR nativo), não existe nenhum mecanismo para restringir o acesso não autorizado a uma rede. Dessa forma, qualquer dispositivo consegue ingressar na rede, divulgar rotas e participar da tabela de roteamento dos dispositivos ativos na rede.

A utilização do *plugin secure* OLSR, contribui para solucionar essa situação. Com base nos resultados obtidos, esse trabalho contribui para implementar uma rede *mesh* segura, visto que para participar da rede, um nó precisa ter suas credenciais cadastradas no servidor de chaves (SKM). Dessa forma, é possível garantir a autenticidade e integridade das mensagens de controle trafegadas na rede. Em razão do descarte dos pacotes que não obtiveram sucesso na verificação da assinatura, um falso nó não pode realizar nenhuma alteração nas tabelas de roteamento dos dispositivos participantes da rede.

Outro aspecto importante é o gerenciamento de chaves realizado pelo SKM, o qual garante que apenas dispositivos autorizados tenham acesso a chave simétrica utilizada na rede *mesh.* Como a arquitetura proposta realiza o gerenciamento automático dessa chave, caso ela seja descoberta através de ataques de força bruta, ela deixará de ser válida após o SKM gerar uma nova chave. Nesse caso, um dispositivo não autorizado não consegue obter a nova chave, visto que, o SKM envia essa chave criptografada para os dispositivos autorizados. Dessa forma, seria necessário um novo ataque na rede. Entretanto, ataques de força bruta requerem um determinado tempo para serem efetuados. Dessa forma, a descoberta de uma chave válida no momento é improvável, visto que a atualização da chave ocorre em intervalos de tempos menores que o tempo necessário para realizar esse tipo de ataque.

Nesses termos, a solução apresentada nesse trabalho apresentou resultados satisfatórios em relação a autenticação dos dispositivos que desejam ingressar em uma rede *mesh*. Assim, essa forma de comunicação pode ser

aplicada na implementação de uma *Smart Grid*, atendendo aos requisitos necessários para garantir uma comunicação segura e confiável entre os dispositivos ativos no sistema elétrico de potência, como por exemplo, um concentrador de dados e os dispositivos atendidos por ele.

Portanto, a principal contribuição deste trabalho é provar que uma rede *mesh* pode fornecer uma comunicação segura e pode ser utilizada na implementação de um *Smart Grid*.

Agradecimentos

Os autores agradecem a CAPES pelo apoio para o desenvolvimento desse trabalho.

Referências

ABELÉM, A. J. G. et al. Redes mesh: Mobilidade, qualidade de serviço e comunicação em grupo. 2007.

BOWITZ, A. G. et al. BatCave: Adding security to the BATMAN protocol. In: Digital Information Management (ICDIM), 2011 Sixth International Conference on. IEEE, 2011. p. 199-204

CARDOSO, T. M.; MARQUES, P. C. F. FURLANETTO, P. C. Rede Mesh: topologia e aplicação. In: Revista iTEC. Vol. IV, n. 4, p. 16, 2012.

CLAUSEN, T.; JAQCQUET, P. Optimized link state routing (OLSR) RFC 3626. IETF Networking Group. 2003.

EKANAYAKE, J. et al. Smart grid: technology and applications. John Wiley & Sons, 2012.

FERNANDES, N. C. Análise de Ataques e Mecanismos de Segurança em Redes Ad Hoc. 2006. 117 f. Dissertação(Doutorado Engenharia Elétrica)-Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2006.GTREI. Grupo de Trabalho de Redes Elétricas Inteligentes. Smart Grid. Relatório. 2010.

HAFSLUND, A. et al. Secure Extension to the OLSR protocol. In: Proceedings of the OLSR Interop and Workshop, San Diego, 2004.

JUNIOR, A. A.; DUARTE, O. C. M. B. Segurança no roteamento em redes móveis ad hoc. In: Seminário de Tópicos Especiais em Redes de Computadores, 2003. Rio de Janeiro: GTA-Universidade Federal do Rio de Janeiro, 2003. p. 1-16.

LOPES, Y. et al. Smart Grid e IEC 61850: Novos Desafios em Redes e Telecomunicações para o Sistema Elétrico. In: XXX Simpósio Brasileiro de Telecomunicações, 2012.Brasília: Universidade de Brasília ,2012.

MACHADO, A. L. Autenticação centralizada com Freeradius em infraestrutura de redes mesh. 2013. 55 f. Monografia (Graduação em Redes de Computadores) - Instituto Federal de Educação, Ciência e Tecnologia Catarinense, Sombrio, 2013.

RAMOS, M. L. S. Proposta de um método de segurança da informação para sistemas de automação em redes elétricas inteligentes. 2012. 108 f. Dissertação (Mestrado Desenvolvimento de Tecnologia). Instituto de Engenharia do Paraná, Curitiba, 2012.

SEN, J. Security and privacy issues in wireless mesh networks: A survey. In: Wireless networks and security. Springer Berlin Heidelberg, 2013. p. 189-272.

SILVA, Z. S. Construindo roteadores Wi-Mesh com GNU/Linux E OLSR. 2015. 94 f. Monografia (Especialização Administração de Redes Linux). Universidade Federal de Lavras, Lavras, 2011.

WANG, W.; XU, Y.; KHANNA, M. A survey on the communication architectures in smart grid. Computer Networks, v. 55, n. 15, p. 3604-3629, 2011.

ZHANG, Y.; LUO, J.; HU, H. Wireless mesh networking: architectures, protocols and standards. CRC Press, 2006.

ZUCCHI, W. L. O que é uma rede mesh e como o padrão IEEE 802.16 se aplica a esse tipo de topologia. In: Revista RTI, p. 104-107. 2006.