VIABILIDADE DO PROTOCOL µTESLA EM REDES VEICULARES

CONCEIÇÃO, Rodrigo Martins da
Universidade Estadual Paulista (UNESP)
rodrigomc07@gmail.com

LOBATO, Renata Spolon
Universidade Estadual Paulista (UNESP)
renata@ibilce.unesp.br

MANACERO Junior, Aleardo
Universidade Estadual Paulista (UNESP)
aleardo@ibilce.unesp.br

SPOLON, Roberta
Universidade Estadual Paulista (UNESP)
roberta@fc.unesp.br

RESUMO: Uma rede veicular é um tipo especial de rede móvel na qual os nós representam estações base ou veículos que se deslocam rapidamente. Uma questão importante neste contexto é a segurança na rede. O protocolo µTESLA foi projetado para autenticar mensagens enviadas por radiodifusão. Esse mecanismo garante que uma determinada mensagem recebida realmente partiu do remetente indicado e que essa mensagem não foi alterada durante sua transmissão pela rede. O objetivo do nosso trabalho foi implementar esse protocolo no simulador de rede (NS-2) e testá-lo para verificar sua viabilidade nas redes veiculares. Foram realizados testes em diferentes cenários, mostrando o desempenho e escalabilidade do protocolo na troca de informação entre os nós da rede.
PALAVRAS-CHAVE: µTESLA, autenticação, segurança, radiodifusão, VANET.

ABSTRACT: *A vehicular network is a special type of mobile network in which nodes represent base stations or vehicles moving quickly. An important issue in this context is the network security. The µTESLA protocol was designed to authenticate messages sent in broadcasting. This mechanism ensures that a certain received message actually came from the sender indicated and that this message was not altered during transmission over the network. The aim of our study was to implement this protocol in simulator network (NS-2) and test it to verify its viability in vehicular networks. Tests were performed in different scenarios showing the performance and scalability of the protocol for the exchange of information between network nodes.*
KEYWORDS: *µTESLA, authentication, safety, broadcasting, VANET.*

1. INTRODUÇÃO

Uma VANET (*Vehicular Ad Hoc Network*) é um tipo de rede *ad hoc* móvel que permite a comunicação direta entre veículos e entre veículo e estação base por meio de um dispositivo sem fio instalado nos veículos. Além disso, existe uma alta mobilidade e velocidade dos nós e o tempo de conexão entre esses nós é bastante curto (HASSAN, 2009).

Contudo, não basta somente que os veículos se comuniquem. Para que as VANETs funcionem de forma adequada e satisfatória, é preciso haver uma comunicação eficiente e segura.

Quando a comunicação acontece entre dois veículos apenas, um simples esquema de criptografia simétrica garante a autenticidade e integridade das mensagens trocadas. Porém quando uma comunicação requer radiodifusão, ou seja, quando uma mensagem precisa ser enviada para vários veículos ao mesmo tempo, faz-se necessário um esquema mais complexo, como a criptografia assimétrica.

Esse é um desafio que se coloca até hoje, isto é, conseguir autenticação de

mensagens transmitidas por radiodifusão de forma rápida e com pouca sobrecarga no sistema (ZEADALLY, 2010).

Tendo esse desafio como base, neste artigo, propomos como solução a utilização do protocolo μTESLA nas redes veiculares e mostramos os resultados alcançados por meio de simulações realizadas no simulador de rede denominado *Network Simulator* versão 2, ou, simplesmente, NS-2.

As seções subsequentes deste artigo estão organizadas da seguinte forma: na Seção 2 são apresentados os trabalhos relacionados ao tema proposto. Na Seção 3 são apresentadas as principais características das redes veiculares. Na Seção 4 são explicados os modelos de comunicação usados nas redes veiculares. Na Seção 5 é descrito o protocolo μTESLA. Na Seção 6 são apresentados os detalhes da implementação do protocolo. Na Seção 7 são mostrados os resultados das simulações. Finalmente, na Seção 8 encontram-se as considerações finais.

2. TRABALHOS RELACIONADOS

Com a promessa de melhorar a segurança nas estradas e fornecer maior conforto aos motoristas e passageiros, as redes veiculares vêm atraindo a atenção de pesquisadores do setor acadêmico, governamental e industrial (ZEADALLY, 2010). Como consequência, diversos projetos importantes foram desenvolvidos ou ainda estão em desenvolvimento em diferentes partes do mundo.

Nos Estados Unidos, a organização denominada ITS *America*, juntamente com órgãos públicos, empresas privadas e instituições acadêmicas, dedica-se ao avanço da pesquisa, desenvolvimento e implantação de Sistemas Inteligentes de Transporte (ITS AMERICA, [2014?]).

Na União Europeia, destaca-se a ERTICO (*European Road Transport Telematics Implementation Coordenation Organization),* uma organização fundada pela união entre Comissão Europeia, ministérios de transporte e indústrias europeias. É uma rede de associados interessados na melhoria e padronização de serviços de transporte e ITS na Europa (ERTICO ITS EUROPE, [2014?]).

Car-to-Car Communications Consortium, também na União Europeia, visa contribuir para o melhoramento da segurança e eficiência do tráfego rodoviário, por meio do princípio dos Sistemas de Transporte Inteligentes cooperativos (CAR 2 CAR COMMUNICATION CONSORTIUM, [2014?]).

No Japão, destaca-se o JARI (*Japan Automobile Research Institute*), uma organização de interesse público destinada às atividades de pesquisas e testes automotivos. O JARI avalia diversas tecnologias envolvendo veículos inteligentes, prevenção de colisões e sistemas de segurança (JAPAN AUTOMOBILE RESEARCH INSTITUTE, [2014?]).

No Brasil, o Instituto Nacional de Ciência e Tecnologia em Sistemas Embarcados Críticos (INCT-SEC) possui um grupo de trabalho que desenvolve sistemas de navegação autônoma e assistida para veículos terrestres. O projeto CARINA (Carro Robótico Inteligente para Navegação Autônoma) é uma aplicação que tem como objetivo reduzir o número de acidentes em ruas e rodovias e aumentar a eficiência no trânsito (INSTITUTO NACIONAL DE CIÊNCIA E TECNOLOGIA EM SISTEMAS EMBARCADOS CRÍTICOS, [2014?]).

Além desses projetos maiores, diversos outros trabalhos mais específicos estão sendo publicados atualmente. Segue uma breve descrição de alguns dos trabalhos relacionados à segurança da rede.

Rehman, Bourdoucen e Ould-Khaoua (2013) propõem um novo protocolo que se utiliza das informações dos veículos vizinhos para disseminar mensagens de alerta advertindo os veículos em perigo.

Lin e Li (2013) propõem um esquema cooperativo de autenticação de mensagens para VANETs, prometendo diminuir a sobrecarga e o atraso na autenticação dos veículos eliminando redundâncias.

Lyu et al. (2013) propõem um novo mecanismo de autenticação VANET, denominado VSPT (*VANET authentication with Signatures and Prediction-based TESLA*). A ideia desse protocolo é combinar as vantagens do ECDSA (*Curve Digital Signature Algorithm*) e *Prediction-based TESLA*.

O GSA (*Group-based Source Authentication*) é um protocolo desenvolvido por Lu et al. (2010) para autenticação de mensagem na origem. O protocolo usa os atributos de grupo para proteger a transmissão de dados em uma comunicação intragrupo e utiliza o esquema do protocolo TESLA para realizar autenticação na origem em uma comunicação intergrupo.

A proposta de Yeo e Youm (2010) é baseada no protocolo μTESLA. O 2LXORC (*2-Level eXclusive-Or Chain*) utiliza uma cadeia de XOR em dois níveis, garantindo tolerância à perda de dados em longo prazo e suporte para vários emissores.

Por fim, Wang et al. (2011) fazem uma análise do protocolo μTESLA através de uma modelagem que utiliza um processo formal de álgebra em CSP. De acordo com os autores, a realização de testes com ataques à rede mostraram bons resultados, demonstrando que a propriedade de autenticação por radiodifusão do protocolo é correta e satisfatória.

Esses trabalhos representam apenas uma pequena parcela de um vasto conjunto, ilustrando o quanto é importante e promissor esse campo de pesquisa. Apesar dos inúmeros trabalhos já realizados, ainda existem muitas questões em aberto, como por exemplo, a implementação de esquemas de autenticação e troca de mensagens mais eficiente e segura.

Essa situação reforça ainda mais a relevância do presente trabalho que acreditamos ser inédito, visto que não encontramos na literatura atual outra proposta para a utilização do protocolo μTESLA nas redes veiculares, com implementação e análise de resultados.

3. REDES *AD HOC* VEICULARES

As redes veiculares ou VANETs (*Vehicular Ad Hoc Networks*) são um subconjunto das MANETs (*Mobile Ad Hoc Networks*). Estas são redes compostas por dispositivos de comunicação, geralmente móveis, capazes de se interconectar espontaneamente (HOGIE et al., 2006).

As VANETs, porém, possuem algumas características particulares, elencadas por Hassan (2009):

- Os nós representam veículos e infraestruturas fixas de beira de estrada (denominadas também estações base ou RSUs – *RoadSide Units*);
- Os veículos se movimentam rapidamente;
- Os padrões de movimento dos veículos ficam restritos à topologia da estrada;
- Os veículos enviam e recebem mensagens ao mesmo tempo, tornando a rede muito dinâmica;
- A densidade veicular varia de tempos em tempos, de acordo com a hora do dia.

Essas características desafiam os pesquisadores gerando uma necessidade de protocolos e arquiteturas específicos para esse tipo de rede. Ao mesmo tempo, são desenvolvidas novas tecnologias e diversas aplicações em diferentes áreas, como segurança veicular, gerenciamento de tráfego, localização de posto de combustível, pagamento automático e acesso à Internet (ZEADALLY et al., 2010).

Na Figura 1 é ilustrada uma típica VANET com veículos e estações base. É possível observar as trocas de mensagens de segurança ocorrendo diretamente entre veículos e entre estação base e veículos após um evento de emergência. Essa situação fornece condições para que os motoristas possam reagir de forma mais consciente e segura com relação ao acidente.

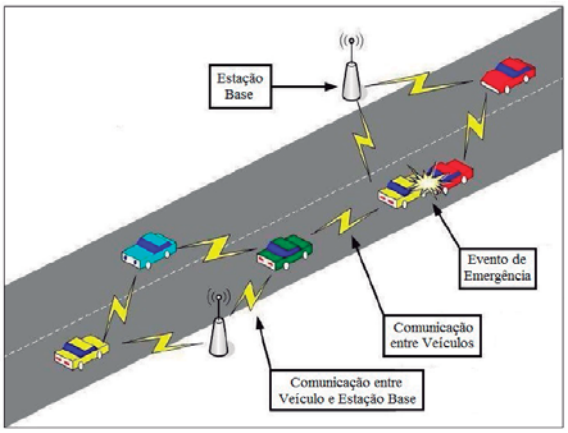

Figura 1 - Uma VANET (RAYA; HUBAUX, 2007).

4. COMUNICAÇÃO

Nas VANETS todos os veículos atuam como transceptores (enviam e recebem mensagens ao mesmo tempo) e roteadores para enviar informações para toda a rede. Além disso, os veículos são equipados com um GPS (*Global Positioning System*) para informação de posição e uma interface de rádio ou OBU (*OnBoard Unit*) para tornar possível a comunicação sem fio.

A seguir são descritas três configurações possíveis de comunicação, apresentadas por ZEADALLY et al. (2010).

4.1. COMUNICAÇÃO ENTRE VEÍCULOS

A comunicação entre veículos utiliza uma transmissão por multissaltos para enviar as informações relacionadas ao tráfego para um grupo de receptores, conforme apresentado na Figura 2.

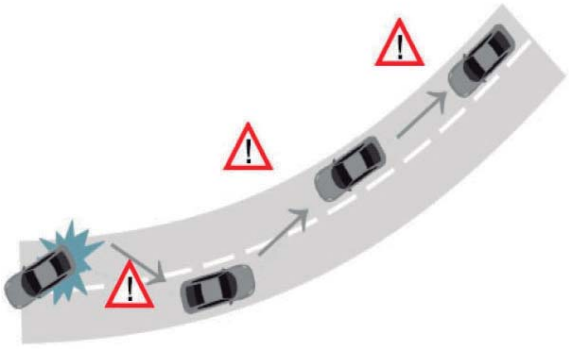

Figura 2 - Comunicação entre veículos (ZEADALLY et al., 2010).

Nas comunicações entre veículos pode haver dois tipos de encaminhamento de mensagem, sendo difusão simples e difusão inteligente. Em ambos os casos, o veículo que recebe uma mensagem verifica se esta veio de um veículo que está mais atrás ou mais adiante. Caso a mensagem tenha partido de um veículo que está mais atrás, ela é simplesmente ignorada; porém, se a mensagem foi enviada por um veículo que está mais adiante, ela é retransmitida em uma nova difusão, garantindo que todos os veículos a recebam.

A diferença entre essas duas abordagens é que na difusão simples os veículos enviam as mensagens periodicamente e em intervalos de tempo regulares, enquanto que na difusão inteligente o número de transmissões de mensagens é limitado.

4.2. COMUNICAÇÃO ENTRE VEÍCULO E ESTAÇÃO BASE

Nesse tipo de comunicação a estação base envia as mensagens para todos os veículos próximos, através de uma transmissão de único salto, conforme mostrado na Figura 3.

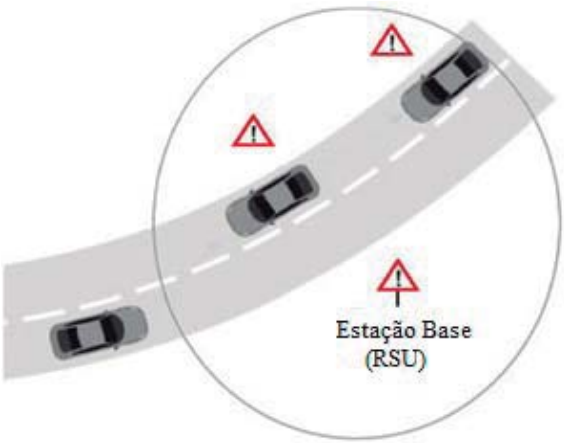

Figura 3 - Comunicação entre veículo e estação base (ZEADALLY et al., 2010).

As estações base são instaladas a cada quilômetro ou menos e possuem uma conexão de banda larga com os veículos, podendo transmitir altas taxas de dados. Dessa forma, a estação base pode estabelecer, por exemplo, um limite de velocidade utilizando um cronograma ou registro de condições de tráfego.

Caso algum veículo exceda esse limite de velocidade, a estação base envia uma mensagem de alerta para que o motorista diminua a velocidade, de forma compatível com a estrada.

4.3. COMUNICAÇÃO BASEADA EM ROTEAMENTO

Na Figura 4 é ilustrado um cenário em que o veículo *A* envia uma mensagem destinada ao veículo *B*. Na comunicação baseada em roteamento a mensagem é

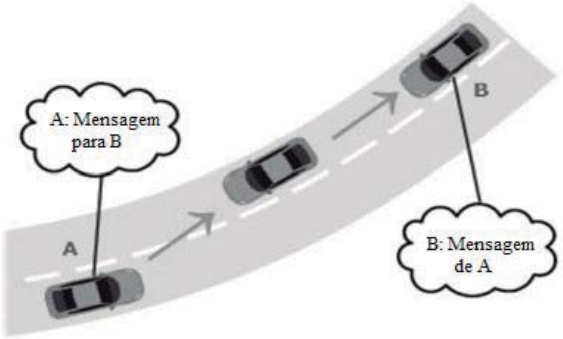

Figura 4 - Comunicação baseada em roteamento (ZEADALLY et al., 2010).

propagada de veículo a veículo (transmissão multissaltos), até que seja alcançado o veículo que detém a informação desejada.

Quando o veículo alvo recebe a mensagem, seu aplicativo responde imediatamente, enviando outra mensagem ao remetente fornecendo a informação solicitada. Da mesma forma, a mensagem não é enviada diretamente de *B* para *A*, isto é, primeiro *B* transmite para o veículo intermediário, que por sua vez retransmite para *A*.

5. PROTOCOLO µTESLA

O projeto SPINS (*Security Protocols for Sensor Networks*), desenvolvido por Perrig et al. (2001), possui duas estruturas de blocos de segurança, o SNEP (*Secure Network Encryption Protocol*) e o µTESLA. O primeiro fornece importantes primitivas básicas de segurança como confidencialidade e autenticação de dados entre duas partes com baixa sobrecarga. O segundo é uma versão do protocolo TESLA (*Timed Efficient Stream Loss-tolerant Authentication*) descrito em Perrig et al. (2000).

Esse protocolo fornece autenticação por radiotransmissão para as redes de sensores, que possuem recursos de memória, de processamento e de energia limitados. Apesar das redes veiculares não terem essas limitações, essa característica é bastante interessante, pois, torna as trocas de mensagens mais rápidas e eficientes. A seguir, segue a descrição do funcionamento do protocolo.

5.1. FUNCIONAMENTO

O funcionamento do µTESLA exige que o tempo seja dividido em intervalos, sendo que o emissor associa cada chave da cadeia de chaves com um intervalo de tempo. Ou seja, no intervalo de tempo *t*, o emissor usa a chave K_t para calcular o valor MAC dos pacotes desse intervalo. A chave K_t será revelada após um atraso de δ intervalos após o final do intervalo de tempo *t*.

A primeira coisa a ser feita é a geração de uma sequência de chaves secretas pelo nó emissor. Para gerar uma cadeia de chaves de tamanho *n*, o emissor escolhe a última chave K_n e gera os outros valores aplicando uma função unidirecional *F* sucessivamente, de forma que $K_j = F(K_{j+1})$.

Na sequência, é preciso que os nós estejam fracamente sincronizados no tempo. Nessa etapa, emissor e receptor trocam mensagens com alguns parâmetros específicos – como chave, hora atual e atraso na divulgação de chaves – e tomam conhecimento do limite superior sobre o erro máximo de sincronização. Ao término dessa etapa, os nós estão aptos a trocarem mensagens autenticadas.

Quando recebe um pacote de dados, o receptor precisa ter certeza de que este não foi alterado, o que pode acontecer se alguém já tiver conhecimento da chave divulgada no outro intervalo de tempo. Por isso, o receptor precisa estar certo de que o emissor não divulgou a chave correspondente ao pacote recebido, o que é chamado de condição de segurança. Se o pacote recebido satisfaz a condição de segurança, o receptor armazena o pacote, caso contrário, o receptor descarta o pacote, pois o mesmo pode ter sido alterado.

Assim que recebe uma chave K_j de um intervalo de tempo anterior, o nó a autentica verificando que ela corresponde à última chave autêntica conhecida K_i, usando um pequeno número de aplicações da função unidirecional $F: K_i = F^{j-i}(K_j)$. De posse dessa chave o nó pode autenticar o pacote de dados armazenado anteriormente.

5.2. EXEMPLO PRÁTICO

O protocolo µTESLA funciona com

um conjunto de chaves relacionadas com intervalos de tempo, onde a chave K_0 é uma chave mestre compartilhada entre todos os nós participantes da rede, a chave K_1 está relacionada com o intervalo 1, a chave K_2 relaciona-se com o intervalo 2 e assim por diante.

Os pacotes enviados no mesmo intervalo de tempo são autenticados com a mesma chave. Na Figura 5 é ilustrado um exemplo em que o pacote *P1* foi enviado no intervalo de tempo 1 e possui um valor MAC gerado com a chave K_1, os pacotes *P2* e *P3* foram enviados no intervalo 2 e possuem um valor MAC gerado com a chave K_2 e assim sucessivamente.

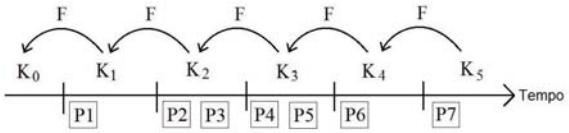

Figura 5 - Funcionamento do protocolo µTESLA.

Supondo que as chaves são divulgadas a cada 2 intervalos de tempo, no intervalo 3 a chave K_1 é revelada e pode ser autenticada verificando $K_0=F(K_1)$ e o pacote *P1* seria, então, autenticado sem problemas. Os pacotes *P2* e *P3* seriam autenticados no intervalo 4 após a chave K_2 ser divulgada e os pacotes *P4* e *P5* seriam autenticados no intervalo 5 após a divulgação da chave K_3.

O protocolo µTESLA ainda apresenta o seguinte mecanismo para evitar perda de pacotes: caso os pacotes do intervalo 3, por exemplo, tenham sido perdidos por problemas na rede, nenhum receptor, em princípio, poderia autenticar os pacotes enviados no tempo 1, pois o pacote que revelaria a chave K_1 teria sido perdido.

Contudo, no intervalo 4 a chave K_2 é revelada e, então, pode ser autenticada usando a função unidirecional $K_0=F(F(K_2))$, podendo também recuperar K_1 fazendo $K_1=F(K_2)$. Dessa forma, os pacotes *P2* e *P3* poderiam ser autenticados com K_2 e o pacote *P1*, com K_1.

6. IMPLEMENTAÇÃO

O NS-2 é desenvolvido a partir de duas linguagens. A linguagem C++ é utilizada na manipulação de *bytes*, pacotes e cabeçalhos e na implementação de algoritmos que manipulam grandes conjuntos de dados. Por outro lado, a linguagem OTcl, por ser interpretada, é mais adequada na execução das simulações que requerem muitos ajustes de parâmetros e configurações de diversos cenários (FALL; VARADHAN, 2011).

Sendo assim, a implementação do µTESLA foi feita em C++, enquanto que os *scripts*, com as configurações dos cenários de simulação, foram feitos em OTcl.

Para que o novo protocolo fosse integrado e funcionasse de forma conjunta com o simulador foi necessária a criação de um cabeçalho e de novos agentes. Na Figura 6 pode ser observado o diagrama de classes de MTeslaHeaderClass.

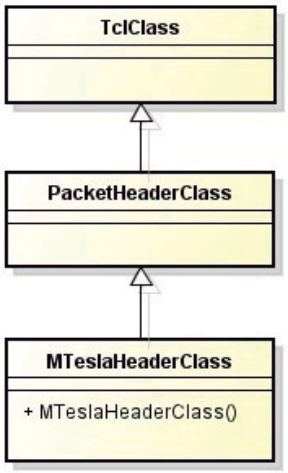

Figura 6 - Diagrama de classes de MTeslaHeaderClass.

A classe MTeslaHeaderClass é derivada da classe PacketHeaderClass, que, por sua vez, é derivada da classe TclClass. MTeslaHeaderClass define o cabeçalho dos pacotes utilizados pelo µTESLA, isto é, quando um pacote de dados é transmitido, todas as informações referentes ao protocolo (como valor MAC, chave e mensagem) se encontram nesse cabeçalho.

Foram criadas, também, as classes Estacao_BaseClass e VeiculoClass, ambas derivadas da classe TclClass, conforme ilustrado na Figura 7.

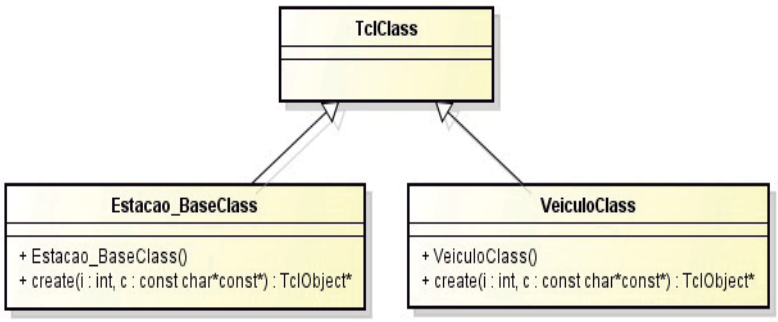

Figura 7 - Diagrama de classes de Estacao_BaseClass e VeiculoClass.

Essas classes criam uma ligação entre as variáveis usadas em Tcl e C++, para que não haja conflito no uso, em conjunto, dessas duas linguagens.

Na Figura 8 é apresentado o diagrama de classes de Estacao_Base e de Veiculo. Ambas são derivadas da classe Agent, sendo esta derivada de Connector. Esta última é derivada de NsObject, que, por sua vez, é derivada de TclObject e Handler.

Essas classes são as responsáveis pela criação dos novos agentes (Estação Base e Veículo), possuindo funções de acesso ao cabeçalho e funções específicas do protocolo. Esses agentes são as entidades capazes de enviar e receber pacotes de dados, ou seja, são eles que trocam mensagens no momento da comunicação.

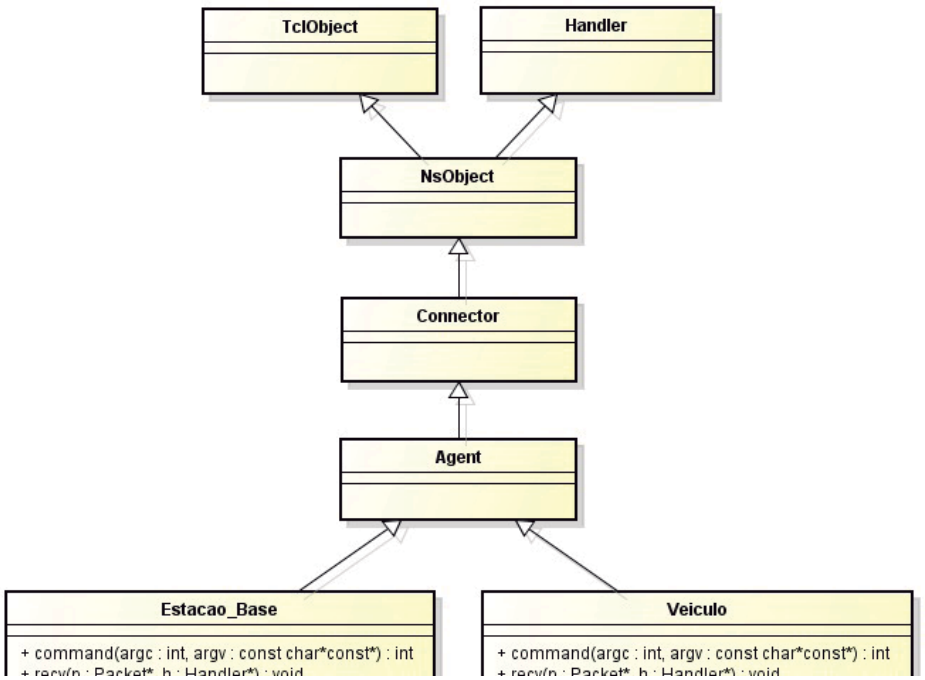

Figura 8 - Diagrama de classes de Estacao_Base e de Veiculo.

7. RESULTADOS

Os cenários para as simulações foram configurados para usar o modelo de propagação Nakagami, uma antena omnidirecional, o protocolo de roteamento DSDV e uma área de 1000 x 1000 metros.

A partir dessas configurações fo-

ram realizados os testes com o µTESLA implementado no NS-2, gerando os seguintes resultados:

7.1. DESEMPENHO

Para o recebimento de mensagens autenticadas são necessárias duas etapas:

- Sincronização entre os nós emissor e receptor;
- Autenticação da mensagem recebida.

O tempo gasto durante esses processos pode ser observado na Tabela 1. O tempo gasto na sincronização entre estação base (emissor) e veículo (receptor) foi de 0,002 segundo. No pior caso, o tempo necessário para a autenticação foi de 0,098 segundo. Esse tempo pode ser menor, dependendo do momento em que a mensagem for enviada. Somados, esses tempos geraram um atraso de 0,1 segundo.

Portanto, em apenas um décimo de segundo é possível que o veículo receba mensagens autenticadas da estação base. É importante mencionar que após a sincronização diversas mensagens podem ser recebidas e autenticadas pelo veículo, sem a necessidade de nova sincronização.

Tabela 1 - Tempo de autenticação de mensagem (em segundos).

Sincronização	Autenticação	Mensagem Autenticada
0,002	0,098	0,1

O motivo dessa rapidez é o fato do protocolo µTESLA evitar o uso de algoritmos complexos, como os de criptografia assimétrica e de assinatura digital. De maneira mais simples, µTESLA autentica suas mensagens usando MACs e divulgação tardia das chaves de autenticação.

7.2. ESCALABILIDADE

A escalabilidade é outra métrica importante nas redes veiculares, pois o número de veículos na rede pode ser grande. Em um cenário com uma maior densidade de veículos, a tendência é diminuir o desempenho da rede, pois, o número de veículos causa um impacto sobre a conectividade da rede e sobre a probabilidade de congestionamento no canal sem fio.

Na Tabela 2 é mostrada a porcentagem de perda de pacotes conforme aumenta o número de veículos na rede.

Tabela 2 - Escalabilidade do protocolo µTESLA.

Quantidade de Veículos	Pacotes Enviados	Pacotes Recebidos	Pacotes Perdidos	Porcentagem de Pacotes Perdidos (%)
20	52.220	53.189	31	0,06
40	117.639	116.871	768	0,65
60	192.473	188.109	4.364	2,27
80	275.134	264.968	10.166	3,69
100	371.352	355.525	15.827	4,26

Pode ser observado que o protocolo praticamente não perde pacotes com até 20 veículos conectados na rede, sendo que a porcentagem de perda é de 0,06%. Quando o número de veículos aumenta para 40, 60, 80 e 100, a porcentagem de perda sobe para 0,65%, 2,27%, 3,69% e 4,26%, respectivamente.

Esses resultados mostram que apesar do aumento significativo do tráfego na rede, a porcentagem de perda de pacotes do µTESLA é pequena e varia pouco à medida que aumenta o número de veículos na rede.

Isso ocorre porque cada veículo que entra na rede exige apenas duas men-

sagens adicionais, transmitidas no momento da sincronização. Depois disso, nenhuma mensagem adicional é necessária, pois cada veículo gerencia seu processo de autenticação de mensagens recebidas.

7.3. SIMULAÇÃO DE ACIDENTE COM CONGESTIONAMENTO

Nesse teste foi simulada uma rodovia com 150 veículos trafegando a 110 k/h. Num determinado momento aconteceu um acidente e, logo, um engarrafamento começou a se formar.

Rapidamente as mensagens de aviso do acidente começaram a ser transmitidas para todos os veículos, dando condições de reação para os motoristas e, consequentemente, evitando novos acidentes. Esse cenário é ilustrado na Figura 9.

Figura 9 - Acidente seguido de congestionamento (CAR 2 CAR COMMUNICATION CONSORTIUM, [2014?]).

De acordo com os resultados da simulação foram enviadas 939.996 mensagens, sendo que 893.126 foram recebidas com sucesso pelos veículos e 46.870 foram perdidas antes de chegarem aos seus destinos.

Apesar de parecer grande à primeira vista, o número de mensagens perdidas não é muito significativo quando comparado com o total de mensagens transmitidas. Ou seja, a porcentagem de mensagens perdidas é de apenas 4,99% do total de mensagens emitidas em razão do acidente.

8. CONSIDERAÇÕES FINAIS

Os resultados da seção anterior mostram que o protocolo µTESLA, apesar de seu atraso por conta da divulgação tardia das chaves de autenticação, é bastante eficiente e escalável. Alguns parâmetros do protocolo são ajustáveis, tornando-o um pouco mais rápido ou mais lento de acordo com a configuração. Contudo, é importante ressaltar que quanto mais rápido, maior é o custo com relação ao processamento e o tráfego de pacotes pela rede.

Além disso, o protocolo µTESLA gera menos sobrecargas de processamento e armazenamento, pois não necessita de outros mecanismos como criptografia assimétrica e assinatura digital, conseguindo fornecer autenticação por radiotransmissão com primitivas unicamente simétricas e introduzindo assimetria com a divulgação tardia das chaves. Essa característica o torna um candidato forte para ser utilizado nas redes veiculares.

9. REFERÊNCIAS BIBLIOGRÁFICAS

CAR 2 CAR COMMUNICATION CONSORTIUM (C2C-CC). **Mission and objectives.** Braunschweig, [2014?]. Disponível em: <http://www.car-2-car.org/>. Acesso em: 7 jun. 2014.

ERTICO ITS EUROPE. **Intelligent transport systems and services for Europe** – bringing intelligence into mobility for people and goods across Europe. [2014?]. Disponível em: <http://www.ertico.com/ertico-its-europe/>. Acesso em: 7 jun. 2014.

FALL, K.; VARADHAN, K. **The ns Manual**, nov, 2011. Disponível em: <http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf>. Acesso em: 20 jan. 2013.

HÄRRI, J.; FIORE, M. **VanetMobiSim – vehicular ad hoc network mobility extension to the CanuMobiSim framework**. Torino: Institut Eurécom, Politecnico di Torino, 2005/2006.

HASSAN, A. **VANET simulation**, 2009. 43 f. Dissertação (Mestrado em Engenharia Elétrica) – School of Information Science, Computer and Electrical Engineering, Halmstad University, 2009.

HOGIE, L.; BOUVRY, P.; GUINAND, F. An overview of MANETs simulation, **Journal Electronic Notes in Theoretical Computer Science (ENTCS)**, v. 150, issue 1, Mar, 2006. p. 81-101.

INSTITUTO NACIONAL DE CIÊNCIA E TECNOLOGIA EM SISTEMAS EMBARCADOS CRÍTICOS

(INCT-SEC). **CARINA**, [2014?]. Disponível em: <http://www.inct-sec.org/br/aplicacoes/carina>. Acesso em: 12 ago. 2014.

ITS AMERICA. **Technology transforming transportation**, [2014?]. Disponível em: <http://www.itsa.org/>. Acesso em: 7 jun. 2014.

JAPAN AUTOMOBILE RESEARCH INSTITUTE (JARI). **About JARI**, [2014?]. Disponível em: <http://www.jari.or.jp/tabid/200/Default.aspx?language=en-US>. Acesso em: 7 jun. 2014.

LIN, X; LI, X. Achieving efficient cooperative message authentication in vehicular ad hoc networks. **IEEE Transactions on Vehicular Technology**, v. 62, n. 7, p. 3339-3348, Sept, 2013.

LU, Y. et al. Group-based secure source authentication protocol for VANETs, In: **Globecom 2010 Workshop on Heterogeneous, Multi-hop Wireless and Mobile Networks**, Miami: IEEE, Dec. 2010. p. 202-206.

LYU, C. et al. Efficient, fast and scalable authentication for VANETs. In: **Wireless Communications and Networking Conference (WCNC)**, Shanghai: IEEE, Apr. 2013. p. 1768-1773.

PERRIG, A. et al. Efficient authentication and signing of multicast streams over lossy channels In: **IEEE Symposium on Security and Privacy**, Berkley: IEEE, May, 2000. p. 56-73.

PERRIG, A. et al. SPINS: security protocols for sensor networks. Berkley: University of California, Department of Electrical Engineering and Computer Sciences. In: **Mobile Computing and Networking**, 2001.

RAYA, M.; HUBAUX, J.-P. Securing vehicular ad hoc networks. **Journal of Computer Security**, v. 15, p. 39-68, Jan. 2007.

REHMAN, O. M. H.; BOURDOUCEN, H.; OULD-KHAOUA, M. Efficient alert messages dissemination in VANETs using single-hop distributed protocols, In: **6th Joint IFIP Wireless and Mobile Networking Conference (WMNC)**, Dubai: IEEE, Apr. 2013. p. 1-4.

WANG, M. et al. Modeling and analyzing the μTESLA protocol using CSP. In: **5th IEEE International Conference on Theoretical Aspects of Software Engineering (TASE),** Xi'an: IEEE, Aug. 2011. p. 247-250.

YEO, D.; YOUM, H. An μTESLA protocols with multi-senders based on a 2-level XOR chain with data-loss tolerance. In: **10th Annual International Symposium on Applications and the Internet (SAINT)**, Seoul: IEEE. July 2010. p. 269-272.

ZEADALLY, S. et al. Vehicular ad hoc networks (VANETS): status, results, and challenges. **Telecommunication Systems,** v. 50, 4, Aug. 2010. p. 217-241.

Rodrigo Martins da Conceição é graduado em Ciência da Computação (UNIVEM – 2011) e mestrando em Ciência da Computação na área de Arquitetura de Computadores e Sistemas Distribuídos. Seus interesses de pesquisa abrangem as áreas de segurança e arquitetura de computadores.

Renata Spolon Lobato é docente da Universidade Estadual Paulista (UNESP) desde 2004. Foi docente da Universidade Federal de Mato Grosso do Sul de 1995 a 2004. Possui graduação e mestrado em Ciências de Computação e doutorado em Ciências pela Universidade de São Paulo. É livre-docente em Sistemas Distribuídos pela UNESP e docente credenciada do Programa de Pós-graduação em Ciência da Computação da UNESP. Suas áreas de interesse são: simulação, simulação distribuída, avaliação de desempenho, sistemas distribuídos, computação paralela e de alto desempenho.

Aleardo Manacero Junior é Livre-Docente em Sistemas de Computação (2004) pela Universidade Estadual Paulista (UNESP). Graduou-se em Engenharia Elétrica (1987), e obteve os títulos de Mestre (1991) e Doutor em Engenharia Elétrica/Automação (1997), todos pela Faculdade de Engenharia Elétrica e Computação da Universidade Estadual de Campinas. Foi professor visitante junto ao *Dept. of Computer and Information Sciences* da Universidade de Oregon, EUA, entre agosto de 2010 e dezembro de 2011. Atualmente é professor adjunto da Universidade Estadual Paulista Júlio de Mesquita Filho, tendo coordenado o curso de graduação em Ciência da Computação, e também o Programa de Pós-Graduação em Ciência da Computação da UNESP. Atua principalmente nos seguintes temas: análise de desempenho, programação paralela, simulação, sistemas de tempo-real e metodologias de ensino de computação.

Roberta Spolon é docente da Universidade Estadual Paulista (UNESP) desde 1992. É livre-docente em Sistemas Distribuídos pela UNESP, Doutora em Ciências e Mestre em Ciências de Computação pela Universidade de São Paulo (USP) e graduada em Ciência da Computação pela UNESP. É líder do Grupo de Pesquisa “Sistemas Computacionais Avançados”. É professora do Programa de Pós-Graduação em Ciência da Computação da UNESP. Atua como revisora de periódicos e membro de diversos comitês científicos. Suas áreas de interesse incluem simulação, simulação distribuída, sistemas distribuídos, virtualização, computação de alto desempenho e paralelismo.