



**UNIVERSIDADE FEDERAL DE SERGIPE**  
**PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA**  
**NÚCLEO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA**  
**COMPUTAÇÃO**

**GEORGE LEITE JUNIOR**

**I9Vanets: um modelo de arquitetura de**  
**software para rede veicular**

**São Cristóvão**

**2016**

**GEORGE LEITE JUNIOR**

**I9Vanets: um modelo de arquitetura de software para  
rede veicular**

Versão original

Dissertação apresentada à Pró-Reitoria de Pós-Graduação e Pesquisa da Universidade Federal de Sergipe para obtenção do título de Mestre em Ciência da Computação pelo Programa de Pós-graduação em Ciência da Computação.

Área de concentração: Redes de Computadores e Sistemas Distribuídos

Orientador: Prof.<sup>a</sup> Dr.<sup>a</sup> Douglas D. J. de Macedo

Coorientador: Prof. Dr. Rogério Patrício Chagas do Nascimento

São Cristóvão

2016

Dissertação de autoria de George Leite Junior, sob o título **“I9Vanets: um modelo de arquitetura de software para rede veicular”**, apresentada à Pró-Reitoria de Pós-Graduação e Pesquisa da Universidade Federal de Sergipe, para obtenção do título de Mestre em Ciência da Computação pelo Programa de Pós-graduação em Ciência da Computação, na área de concentração Redes de Computadores e Sistemas Distribuídos, aprovada em \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_ pela comissão julgadora constituída pelos doutores:

**Prof.<sup>a</sup> Dr.<sup>a</sup>** \_\_\_\_\_

Presidente

Instituição: \_\_\_\_\_

**Prof. Dr.** \_\_\_\_\_

Instituição: \_\_\_\_\_

**Prof. Dr.** \_\_\_\_\_

Instituição: \_\_\_\_\_

## Resumo

LEITE, George Junior. **I9Vanets: um modelo de arquitetura de software para rede veicular**. 2016. 32 f. Dissertação (Mestrado em Ciência da Computação) – Pró-Reitoria de Pós-Graduação e Pesquisa, Universidade Federal de Sergipe, Sergipe, 2016.

Em consequência do crescimento populacional, as grandes cidades enfrentam problemas cotidianos relacionados à mobilidade urbana tais como: congestionamentos, qualidade das rodovias, ineficiência de transportes públicos, entre outros. Iniciativas de sistemas de transportes inteligentes (ITS) agem como uma solução eficiente para melhorar o funcionamento e desempenho dos sistemas de tráfego, reduzindo congestionamentos e aumentando a segurança para os cidadãos. Atualmente, pesquisadores vem buscando nas redes veiculares ad-hoc (VANET) uma possível solução para os problemas referentes a mobilidade urbana. Contudo, VANETs ainda apresenta uma série de desafios que devem ser resolvidos para que seu uso seja consolidado. Deste modo, o presente trabalho apresenta uma arquitetura denominada i9Vanets, cujo intuito é virtualizar uma rede veicular para auxiliar nas soluções dos principais desafios relacionados à VANETs. Também é apresentado uma análise, dos testes realizados em laboratório, com objetivo de avaliar seu desempenho e capacidade operacional. Sendo assim, a conclusão deste trabalho, foi apresentar a viabilidade técnica da arquitetura i9Vanets bem como possíveis aplicações.

Palavras-chaves: Rede Veicular. Computação em Nuvem.

# Abstract

LEITE, George Junior. **I9Vanets: a software architecture model for a vehicular network**. 2016. 32 p. Dissertation (Master of Science of Computer) – Dean of Graduate Studies and Research, Federal University of Sergipe, Sergipe, 2016.

Bla bla bla bla bla bla Bla bla bla bla bla blaBla bla bla bla blaBla bla bla bla bla  
bla bla bla bla Bla bla bla bla bla bla Bla bla bla bla bla blaBla bla bla bla bla  
blaBla bla bla bla bla bla blaBla bla bla bla bla bla Bla bla bla bla bla blaBla bla  
bla bla bla bla blaBla bla bla bla bla blaBla bla bla bla bla bla Bla bla bla bla bla  
bla blaBla bla bla bla bla bla blaBla bla bla bla bla blaBla bla bla bla bla bla Bla  
bla bla bla bla bla blaBla bla bla bla bla blaBla bla bla bla bla blaBla bla bla bla  
bla bla bla Bla bla bla bla bla bla blaBla bla bla bla bla blaBla bla bla bla bla bla  
blaBla bla bla bla bla bla bla Bla bla bla bla bla bla blaBla bla bla bla bla blaBla bla  
bla bla bla bla bla.

Keywords: Vehicle Network. Cloud Computing.

## Lista de figuras

Figura 1 – Arquiteturas de redes veiculares(LUÍS, 2009) . . . . .	18
Figura 2 – Módulos definidos para a arquitetura I9Vanet. . . . .	19
Figura 3 – Modelo de comunicação V2A, I2A, AI2AI e AV2AI. . . . .	21
Figura 4 – Organização dos servidores na nuvem. . . . .	23
Figura 5 – Topologia de rede utilizada para a realização dos experimentos. Fonte: Criada pelo autor. . . . .	26

## Lista de algoritmos

## Lista de tabelas

Tabela 1 – Colunas das tabelas <i>Server</i> . . . . .	23
Tabela 2 – Colunas da tabela <i>Device</i> . . . . .	23
Tabela 3 – Linhas do arquivo de movimentação. . . . .	26



## Lista de abreviaturas e siglas

ITS	Intelligent Transport System
VANET	Vehicle Network Ad-hoc
VCC	Vehicle Cloud
V2V	Veículo para Veículo
V2I	Veículo para Infra-estrutura
IP	Internet Protocol
MAC	Media Access Control
NBAPI	Northbound Application Programming Interface
NFV	Network Function Virtualization
ONF	Open Networking Foundation
OSGI	Open Services Gateway Initiative
PHP	HyperText Preprocessor
SBAPI	Southbound Application Programming Interface
SDN	Software-Defined Networking
SQL	Structured Query Language
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

## Sumário

1	Introdução . . . . .	11
1.1	Justificativa, Problemática e Hipótese . . . . .	12
1.1.1	Justificativa . . . . .	12
1.1.2	Problemática . . . . .	15
1.1.3	Hipótese . . . . .	15
1.2	Objetivos da Dissertação . . . . .	15
1.2.1	Objetivos Específicos . . . . .	16
1.2.2	Metodologia . . . . .	16
1.2.3	Contribuições . . . . .	16
1.3	Organização do Trabalho . . . . .	16
2	Redes Veiculares . . . . .	17
2.1	Definição . . . . .	18
2.2	Características e Desafios . . . . .	18
2.3	Aplicações . . . . .	18
2.4	Protocolos . . . . .	18
3	Arquitetura I9Vanet . . . . .	19
3.1	Visão Geral . . . . .	19
3.2	Módulo de Comunicação . . . . .	19
3.2.1	Comunicação Infra-Cloud . . . . .	20
3.2.2	Comunicação Veículo-Cloud . . . . .	20
3.3	Módulo de Segurança . . . . .	20
3.4	Módulo de Gerenciamento dos Servidores . . . . .	22
3.5	Módulo de Roteamento . . . . .	24
3.6	Módulo de Aplicações . . . . .	24
4	Avaliação da Arquitetura . . . . .	25
4.1	Cenário Proposto . . . . .	25
4.2	Experimentos . . . . .	28
4.3	Resultados . . . . .	28

<b>5</b>	<b>Conclusão . . . . .</b>	<b>29</b>
<b>5.1</b>	<b>Conclusões deste Trabalho . . . . .</b>	<b>29</b>
<b>5.2</b>	<b>Trabalhos Futuros . . . . .</b>	<b>29</b>
	<b>Referências . . . . .</b>	<b>30</b>

# 1 Introdução

A conectividade inerente às cidades inteligentes abre uma fronteira bastante promissora no que se refere ao controle de acesso às informações (LI, 2010) (ZHU; LIU, 2015) e arquiteturas distribuídas para sistemas inteligentes de transporte (ICE et al., 2001).

Um sistema inteligente de transporte ou System (ITS) representa “a aplicação de sensores avançados, computadores, dispositivos eletrônicos e tecnologias de comunicação e gerenciamento estratégico integrado visando melhorar a segurança e a eficiência do sistema de gerenciamento de tráfego” (NASIM; KASSLER, 2012).

De acordo com Ball e Dulay (2010), a segurança no trânsito é a motivação principal para as pesquisas no âmbito das redes veiculares. Da mesma maneira, conforme a Organização Mundial da Saúde (OMS), acidentes acometidos por veículos são responsáveis por mais de um milhão de mortos e 50 milhões de feridos anualmente em todo o mundo (PEDEN et al., 2004). Nos Estados Unidos e no Brasil, acidentes relacionados a veículos são a terceira maior causa de mortalidade evitável e incapacitação profissional precoce (SYSTEMATICS; MEYER, 2011) (DIB et al., 2007).

Segundo o IPEA (Instituto de Pesquisa Econômica Aplicada), a mobilidade urbana, nos dias atuais, é um dos principais problemas dos grandes centros. Os reflexos sobre o transporte urbano são evidentes, caracterizados principalmente pelo aumento do tráfego nas vias e conseqüentemente dos congestionamentos (CARVALHO et al., 2010). Porém, baseado em Gupte and Younis (2012), é possível gerir o tráfego de forma eficiente utilizando redes de veículos ad-hoc (VANET) de maneira eficiente e autônoma.

Segundo Cavalcanti (2008), a crescente quantidade de dispositivos eletrônicos que podem ser embarcados em veículos automotores como DVD, TV, GPS e telefone celular, faz com que os veículos deixem de ser apenas um meio de transporte e passem a oferecer uma rede de serviços e entretenimento para os condutores e/ou passageiros.

Os Sistemas Inteligentes de Transporte (ITS), são atualmente a mais importante aplicação de uma rede veicular, fornecendo serviços de segurança rodoviários (XU et al., 2003) ou informações relativas às situações de tráfego. Dito por Sumra (2011) e Luís (2009), o principal objetivo de uma VANET é prover segurança aos passageiros nas estradas.

O IEEE (*Institute of Electrical and Electronics Engineers*) estabeleceu que a tecnologia de comunicação utilizada entre veículos, sendo nos Sistemas de Transporte Inteligente ou na comunicação Veículo para Veículo, é derivada do padrão 802.11p, este padrão também

conhecido como rede local sem fio cujo espectro de frequência utilizada é na faixa de 5,9 GHz, sendo atribuída numa base harmonizada na Europa, em consonância com atribuições semelhantes no EUA.

## 1.1 Justificativa, Problemática e Hipótese

### 1.1.1 Justificativa

É crescente o número de pesquisas sobre VANETs com intuito de aumentar a segurança viária, bem como ofertar e comercializar novos serviços a motoristas e passageiros. VANETs que usam veículos como nós móveis são uma subclasse de rede móveis ad hoc chamadas de MANETs (*Mobile Ad hoc Network*), elas fornecem comunicação entre os veículos próximos e entre veículos e equipamentos à beira da rodovia mas, aparentemente difere de outras redes por suas próprias características do ambiente.

Diante da evolução das redes veiculares e da necessidade de garantir o táfego dos dados dentro da alta mobilidade dos nós numa VANET, vê-se a importância de utilizar tecnologias que permitam auxiliar na solução dos principais desafios relacionados a este tipo de rede.

O uso de VANETs representa uma oportunidade para melhora na segurança, eficiência e conforto relacionados ao trânsito nas grandes cidades, porém, possui características específicas que apresentam obstáculos a serem contornados.

Os nós numa rede VANET são muito dinâmicos pois os veículos possuem velocidade e direção variável. A alta mobilidade dos nós conduz a uma topologia de rede dinâmica caracterizada pela constante perda de comunicação fazendo com que os algoritmos de roteamentos tornem-se complexos e limitados.

Normalmente, VANETs apresentam três aspectos para pesquisas: roteamento, segurança e privacidade, bem como suas aplicações ([LIANG et al., 2015](#)).

O roteamento em VANETs é baseada na comunicação sem fio e devido à topologia dinâmica e conectividade intermitente, a maioria dos algoritmos em MANETs não estão disponíveis para os vários cenários VANETs, tornando o tema ainda em aberto e sendo foco de várias pesquisas na atualidade com objetivo de dar confiabilidade na comunicação levando em consideração número de emissores e receptores e tipos de roteamento.

Comentado por Cavalcanti(2008), os nós de uma rede veicular apresentam como principal característica, a alta mobilidade, por serem dinâmicos e apresentarem rápidas variações das condições do meio sem fio são os principais geradores de desafios. O alto dinamismo representa um contratempo para as comunicações, visto que nem sempre o tempo de contato dos veículos é suficiente estabelecer uma comunicação efetiva.

Trabalhos como , Nandan et al. (2005) Chen (2006) e Hadaller (2007) , apresentam propostas que vão desde a implementação de uma infra-estrutura fixa até definir esquemas de agrupamento de nós, visando reduzir a latência de entrega de mensagens para sistemas de segurança. Nandan et al. (2005), propuseram uma comunicação eficiente para sistemas P2P com disponibilização de conteúdo utilizando um mecanismo de partição dos arquivos disponíveis em pequenos pedaços. Este particionamento visa promover uma forma de transferência distribuída, na qual é possível que um nó esteja recebendo dois pedaços distintos do mesmo arquivo de duas fontes distintas e simultaneamente. Cabe ressaltar que em grande parte as propostas visam agilizar o processo de transferência de dados em VANETs e implementam mecanismos específicos para minimizar os problemas gerados pela alta mobilidade dos nós.

Dentro desse contexto, este artigo apresenta como principal contribuição, um modelo de arquitetura de software flexível e extensível, que utiliza rede veicular e Computação em Nuvem, com a finalidade de proporcionar uma alternativa para os principais desafios relacionados à VANETs tais como:

- **meio físico:** interferências devido à prédios, árvores e outros obstáculos;
- **alta mobilidade:** dificulta a troca de informações mais completas;
- **topologia:** VANET possui uma característica dinâmica, devido à velocidade que os veículos se movimentam;
- **baixa densidade:** quando a densidade de tráfego é baixa e os veículos estão distantes uns dos outros;
- **alta densidade:** muitos veículos em uma pequena área faz com que a quantidade de mensagens trocadas seja um problema;
- **segurança:** como as VANETs suportam aplicações de emergência em tempo real e lidam com informações críticas de segurança no trânsito, estas devem satisfazer os seguintes requisitos de segurança: confidencialidade, integridade, disponibilidade,

autenticidade, privacidade e não repudição para prover segurança na comunicação dos dados (SAMARA; AL-SALIH; SURES, 2010) (MATOS et al., 2013).

Segundo Falchetti et al, os principais desafios técnicos que são abordadas pela maioria dos pesquisadores, podem ser agrupados em seis categorias: mobilidade, latência, confiabilidade, criticidade(prioridade das mensagens), topologia e segurança.

Portanto, pesquisar sobre redes veiculares e computação em nuvem, traz a possibilidade de construção de uma plataforma capaz de criar uma VANET virtualizada facilitando a comunicação entre os nós virtuais da rede e simplificando a implementação dos algoritmos de roteamento, segurança e aplicações.

Os autores Liu et al. utilizaram um controlador SDN (*Networking Defined Software*) para enviar mensagens para os veículos a partir de RSU conectados ao controlador, porém, os veículos devem pertencer à mesma geolocalização.

Foi criado por Hajji e Bargaoui , a plataforma Testbed que serve para testar a implementação real dos roteadores móveis em uma rede veicular virtualizado. A característica principal consiste na sua capacidade para satisfazer ao mesmo tempo as exigências de mobilidade do veículo e o número elástico de roteadores Móveis.

Olariu et al pela primeira vez realizou uma mudança de paradigma imaginado para o ambiente de VANET tradicional para VANET baseado em nuvem. Eles vieram com uma nova noção de VANET chamado AVC (*Autónomas Veiculares Nuvens*). Yan et al. delineou os desafios de segurança e privacidade nas nuvens veiculares. No entanto, Olariu et al. discutiu sobre nuvens veiculares de maneira abstrata, sem mencionar em particular arquitetura do framework.

Recentemente, Hussain et al. propôs três quadros de arquitetura para VANETs baseados em nuvem ou seja VC (*Vehicular Clouds*), VUC (textitVANET using Clouds)), e HVC (*Hybrid Vehicular Clouds*) e propôs um esquema conhecido como TIaaS (*Information of traffic as Service*), onde a infraestrutura de nuvem atua para fornecer informações de tráfego para os veículos que se deslocam sobre a estrada.

Qin et al. (QIN; HUANG; ZHANG, 2012) propôs um esquema de roteamento, para VANETs, baseado em nuvem chamado VehiCloud que fornece serviços de roteamento através da infraestrutura de nuvem. Veículos compartilham a sua actual e futuras informações de localização na forma de *waypoints*) com infraestrutura de nuvem e, em seguida, a nuvem lhes proporciona óptima informações de roteamento.

De acordo com Falchetti et al. , a integração de VANETs com a nuvem aumenta a utilização das capacidades computacionais que são subutilizados por aplicações de segurança e supera os problemas de roteamento em comunicação V2V (Vehicle to Vehicle).

Lee et al. visam interligar recursos OBU (*On Board Unit*) e RSU (*Road Side Unit*) em nuvem para tarefas cooperativas sensoriais, de armazenamento e de computação, enquanto outros (HUSSAIN et al., 2012) (MERSHAD; ARTAIL, 2013) propõem que as RSUs ajam como gateways para nuvens tradicionais para as OBUs.

Foi proposto por Lee et al. (LEE et al., 2014) o VCN (*Vehicular Cloud Networking*) que está sendo visto como uma revolução para modernizar a tradicional VANET, que integra informações de redes e Cloud Computing com VANETs tradicionais. No VCN, veículos e RSU compartilham seus recursos em uma plataforma virtual.

Gerla (2012) , introduziu um novo modelo de computação em, o VCC (*Vehicle Cloud Computing*). VCC é uma variante do MCC (textitMovel Cloud Computing) móvel de computação em nuvem (MCC) (FERNANDO; LOKE; RAHAYU, 2013).

### 1.1.2 Problemática

Como criar uma plataforma em nuvem capaz de permitir a construção de aplicações voltadas para VANETs, considerando o desenvolvimento simplificado e extensível com foco nos principais desafios atualmente encontrados em redes veiculares.

### 1.1.3 Hipótese

Avaliar se o tempo de resposta da arquitetura proposta permite os mais variados tipos de aplicações em VANETs, dada a particularidade deste tipo de rede.

## 1.2 Objetivos da Dissertação

O objetivo principal desta dissertação é apresentar uma arquitetura de software flexível e extensível, com capacidade de virtualizar nós de uma rede VANET, realizando a comunicação entre os elementos de forma virtual na tentativa de corroborar com a solução de alguns dos principais desafios relacionados à rede veiculares.



### 1.2.1 Objetivos Específicos

Para avançar a integração anterior, alguns objetivos específicos devem ser atendidos, a saber:

- Definir a arquitetura de software de maneira que permita a extensibilidade e flexibilidade;
- Construir a arquitetura utilizando a linguagem Java;
- Implementar cada camada da arquitetura para realização de testes;
- Construir uma aplicação como prova de conceito;
- Realizar testes simulados para avaliar seu desempenho.

### 1.2.2 Metodologia

### 1.2.3 Contribuições

A computação veicular em nuvem é um novo campo de pesquisa com o potencial de mudar a vida das pessoas. Ele vai trazer eficiência, segurança e conforto para motoristas e passageiros ([FALCHETTI; AZURDIA-MEZA; CESPEDES, 2015](#)).

## 1.3 Organização do Trabalho

O presente artigo está organizado da seguinte forma: no capítulo II são descritos os trabalhos relacionados ao tema deste estudo; no capítulo 3 é apresentado a proposta da arquitetura I9Vanet; no capítulo 4 é descrito o processo de avaliação da arquitetura; no capítulo 5 é apresentado a conclusão desse estudo, bem como os possíveis trabalhos futuros que poderão ser realizados.

## 2 Redes Veiculares

As VANETs que usam veículos como nós móveis são uma subclasse de rede móveis ad hoc chamadas de MANETs. Elas fornecem comunicação entre os veículos próximos e entre veículos e equipamentos à beira da rodovia. Os nós numa rede VANET são muito mais dinâmicos, pois os veículos possuem velocidade e direção variável. A alta mobilidade dos nós conduz a uma topologia de rede dinâmica caracterizada pela constante perda de comunicação (BUBENIKOVA; DURECH; FRANEKOVA, 2014) (JAKUBIAK; KOUCHERYAVY, 2008). E podem ser categorizadas segundo o tipo de ligações existentes. Dessa maneira, consideremos as três arquiteturas de redes veiculares (LUÍS, 2009), como mostra a figura 1:

- **Arquitetura WLAN ou celular:** baseada na utilização de antenas fixas, colocadas ao longo da rodovia funcionando como pontos de acesso à rede. Não existe qualquer ligação direta entre os veículos. Em um cenário de auto-estrada, a implantação dos equipamentos à beira da rodovia de maneira suficiente para permitir a cobertura necessária pode tornar uma solução bastante dispendiosa.
- **Arquitetura ad-hoc:** é considerada uma arquitetura ad-hoc quando não há qualquer uso de infra-estruturas para realizar a comunicação, sendo as ligações feitas diretamente entre os nós envolvidos. Fatores como velocidade ou densidade dos nós podem pôr em *check* o desempenho deste tipo de rede.
- **Arquitetura híbrida:** tem objetivo de retificar as falhas existentes nas duas arquiteturas anteriores, utilizando concomitantemente as arquiteturas ad-hoc e WLAN

As comunicações em VANETs são categorizadas em 4 tipos:

- Em veículos: pode ser utilizado para detectar a fadiga e/ou sonolência de um motorista que representa risco na segurança;
- Entre veículos: a comunicação V2V (veículo para veículo) pode fornecer uma plataforma de intercâmbio de dados para compartilhamento de informações de advertência de modo a alertar o motorista;
- Entre veículos e rodovia (V2I): permite atualização em tempo real do tráfego e fornece detecção e monitoramento do ambiente;

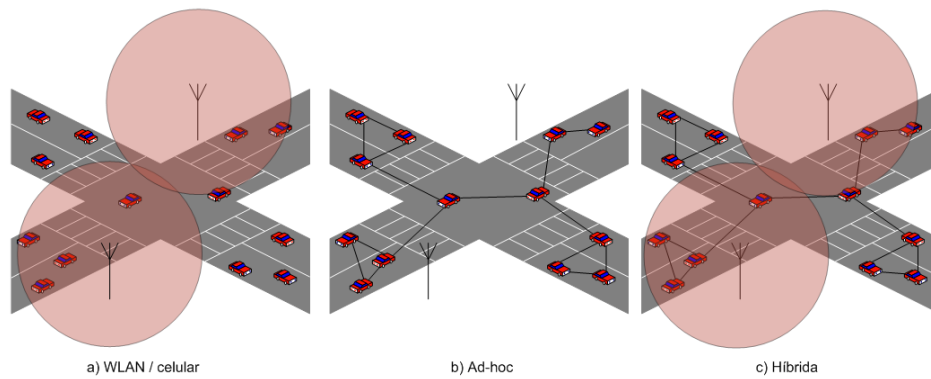


Figura 1 – Arquiteturas de redes veiculares(LUÍS, 2009)

- Entre veículo e nuvem: os veículos podem comunicar-se através da banda larga sem fio tais como 3G, 4G ou WIMAX podendo enviar dados para uma central, o que permitiria um controle mais abrangente do tráfego e assistência ao motorista

As aplicações VANETs podem ser divididas em 3 classes: segurança, entretenimento e assistência ao motorista. O principal desafio relacionado à segurança, está ligado à velocidade no aviso ao condutor, para que possa ter tempo hábil de reação. Nas aplicações de entretenimento, destacam-se os sistemas de compartilhamento de conteúdo e jogos. Já sobre a assistência ao motorista, são aquelas que auxiliam o condutor como por exemplo, através do uso de informações sobre as condições do trânsito (SOUZA RONIEL DE; SOARES, 2014).

## 2.1 Definição

## 2.2 Características e Desafios

## 2.3 Aplicações

## 2.4 Protocolos

### 3 Arquitetura I9Vanet

#### 3.1 Visão Geral

A arquitetura proposta tem a finalidade de criar redes veiculares virtuais em nuvem com foco no auxílio dos principais desafios relacionados à VANETs tais como alta densidade, baixa densidade, alta mobilidade, segurança e privacidade, entre outros. A arquitetura consiste em um modelo aberto, dividido em módulos com funções bem definidas. Cada módulo possui funcionalidades específicas e padrões de comportamentos. Sendo possível estender suas funcionalidades ou até mesmo substituí-las de maneira que atenda às novas necessidades.

A Figura 2 mostra os módulos necessários da arquitetura: *VANETS Applications*, *Server Management Cloud*, *Routing between Nodes*, *Secutity OBU/RSU*, *Infra-Cloud Communication* e *Vehicle-Cloud Communication*. Sendo que cada módulo segue um modelo de arquitetura com objetivos bem definidos, sendo que suas interfaces de comunicação são padronizadas, permitindo substituir um módulo por outro, com a mesma característica, sem que interfira no modelo.

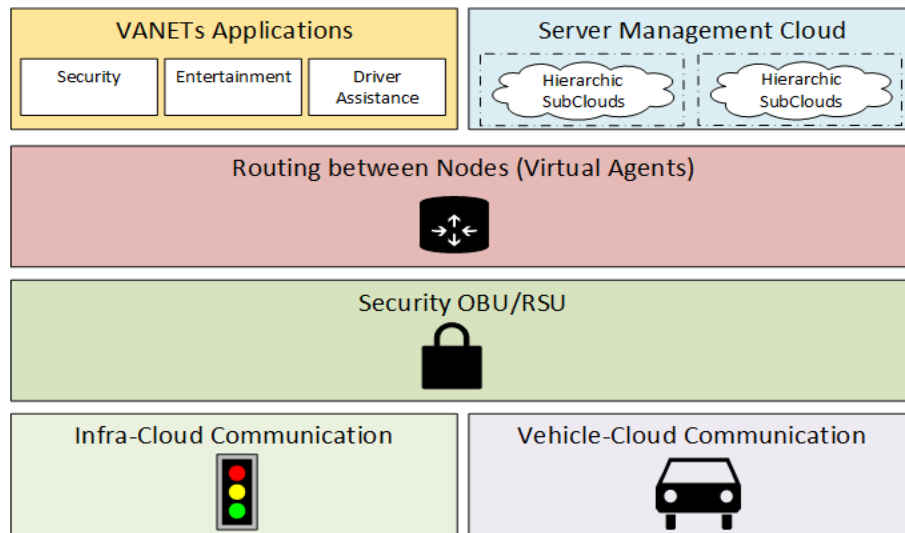


Figura 2 – Módulos definidos para a arquitetura I9Vanet.

#### 3.2 Módulo de Comunicação

Módulo responsável por definir o modelo de troca de mensagens entre os servidores e os dispositivos, sejam OBU ou RSU.

### 3.2.1 Comunicação Infra-Cloud

A forma como a comunicação será efetivamente implementada, não é preocupação da arquitetura, cabe ao desenvolvedor utilizar a tecnologia (socket, webservices, websocket, push, etc) que mais se adequa a sua necessidade e/ou região. Estas mensagens são de entrada e saída, portanto, os equipamentos RSU podem enviar como também receber informações. Em VANETs, pode existir a comunicação entre veículos e dispositivos à beira da rodovia (V2I), todavia, não é o propósito desta arquitetura realizar, de forma direta, a comunicação entre os equipamentos contudo, será realizada entre os respectivos agentes virtuais .

Todos os equipamentos RSU serão virtualizados e representados por agentes virtuais. Sendo assim, pode haver a comunicação agente RSU e o agente do veículo (AV2AI) como também entre agentes RSUs (AI2AI), como mostrado na Figura 3.

### 3.2.2 Comunicação Veículo-Cloud

define a comunicação entre os servidores e os equipamentos instalados nos veículos (OBU). Detalhes da implementação podem ser substituídos permitindo utilizar a melhor tecnologia do momento. Assim como no módulo de Comunicação Infra-Cloud, a comunicação V2V não ocorrerá diretamente, como ocorre nos modelos tradicionais, ela será feita através dos agentes virtuais, que nada mais são do que a representação virtual dos veículos. Desta forma, a comunicação se dará entre os agentes (AV2AV) e entre o agente e o veículo físico (V2A) este processo é exemplificado na Figura 3.

## 3.3 Módulo de Segurança

A segurança na troca de informações em redes veiculares é um dos desafios que devem ser solucionado antes mesmo da implantação de uma solução VANET ser posta no mercado, Então, como garantir os princípios da autenticidade, integridade não repúdio na troca de mensagens entre os veículos? Já que é inviável fornecer as chaves de segurança de todos os veículos durante a comunicação.

Na arquitetura I9Vanet, foi pensado em um processo de comunicação entre os equipamentos OBU e RSU e os servidores em Nuvem, de forma que deva garantir os

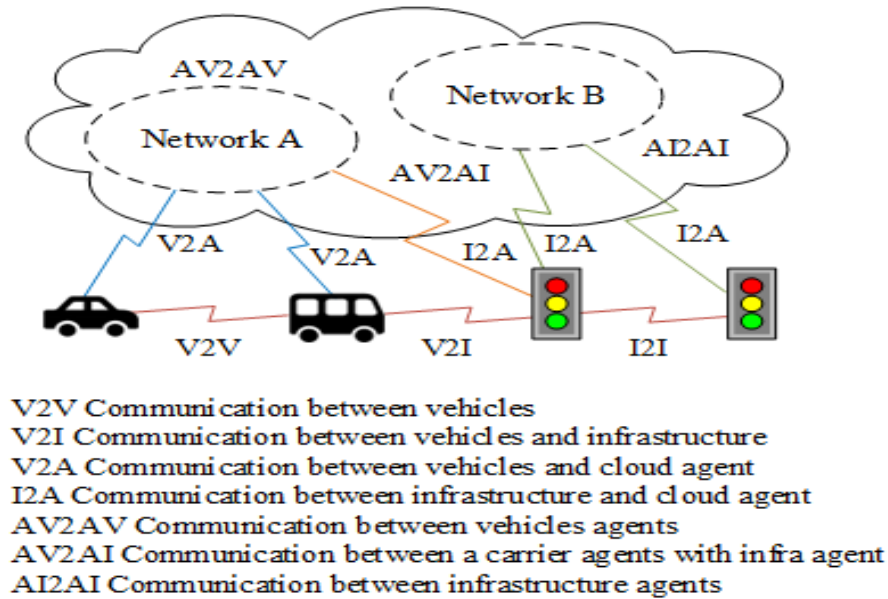


Figura 3 – Modelo de comunicação V2A, I2A, AI2AI e AV2AI.

princípios da autenticidade, integridade e não repúdio através do uso de assinatura digital baseado em algoritmos assimétricos.

A geração das chaves para cada equipamento deve ser controlado pelo órgão responsável pelos veículos, o DENATRAN (Departamento Nacional de Trânsito). Quando um novo veículo for cadastrado em sua base, veículo zero KM, deve ser gerado a chave pública e secreta, sendo que a pública deve ser enviada para os servidores em nuvem responsável pelo gerenciamento da arquitetura I9Vanets e a secreta, inserida no equipamento, no veículo semelhante ao Token e-CPF.

A cada renovação de licenciamento, um novo token deve ser gerado e fornecido ao equipamento e publicado nos servidores. Como também, em caso de venda de um veículo, ao passar para o nome do novo proprietário, deve ser gerado um novo token.

No momento que um equipamento deseja enviar uma mensagem para seu agente em nuvem, ela deve ser assinada com sua chave privada, assim, através da verificação da assinatura com a chave pública, o agente terá certeza que o remetente é confiável. Porém, a mensagem será assinada com a chave secreta do servidor para que o equipamento, OBU ou RSU, possa validar através da chave pública do servidor.

Para o momento não foi necessário criptografar a mensagem, mas este passo pode ser inserido restando apenas avaliar o custo de processamento tanto nos servidores quanto nos dispositivos OBU e RSU.

### 3.4 Módulo de Gerenciamento dos Servidores

Os servidores devem ser distribuídos hierarquicamente de forma que permita controlar regiões separadamente e independentemente. Cada servidor deverá conter informações do servidor pai, no caso, o servidor nível 0 não possui pai.

Quando um veículo iniciar a transmissão, será conectado automaticamente ao servidor Nível 0, informando o comando *StartConnection*, o qual realizará uma busca pela coordenada GPS enviada pelo veículo para localizar o servidor responsável pela região onde se encontra o veículo, podendo ocorrer duas situações:

1. O veículo encontra-se em uma área sem cobertura de cerca virtual, assim, o próprio servidor raiz irá ser responsável por gerenciar a rede VANET deste veículo;
2. O veículo encontra-se em uma área com cobertura de cerca virtual que é de responsabilidade de algum servidor filho. Então o servidor nível 0 irá passar o comando, *ChangeServer*, informando o ip do servidor que deve ser responsável pela rede dessa área, e assim o veículo faz uma nova conexão para o novo servidor.

Cada movimentação do veículo deve ser transmitida para o servidor da área, pelo comando *SendMoviment*, o qual irá verificar se o veículo ainda está em sua região, de acordo com a cerca virtual. Caso não esteja mais sob sua área, o servidor irá enviar o comando *ChangeServer* com o endereço do servidor Pai. Assim, o veículo poderá refazer a conexão com este novo servidor, através do comando *StartConnection*, o qual deverá verificar se a coordenada pertence a este servidor ou de algum de seus filhos.

Em todos os servidores deve existir uma base de dados para definir a hierarquia como também o cadastro dos dispositivos. As tabelas 1 e 2 mostram respectivamente o dicionário de dados da tabela *server* e *device* utilizadas no sistema.

Tabela <i>Server</i>		
Coluna	Tipo do Dado	Descrição
server_id	<i>integer</i>	Identificador único no sistema
address	<i>varchar</i>	Endereço Ip do servidor
server_idsuper	<i>integer</i>	Identificador do servidor pai
virtual_fence	<i>polygon</i>	Cerca virtual de responsabilidade do servidor

Tabela 1 – Colunas das tabelas *Server*.

Tabela <i>Device</i>		
Coluna	Tipo do Dado	Descrição
device_id	<i>integer</i>	Identificador único no sistema
identification	<i>varchar</i>	Identificador do dispositivo (Imei, Mac Address, etc.)
type	<i>char</i>	Tipo do dispositivo (OBU ou RSU)
public_key	<i>byte[]</i> <i>integer</i>	Chave pública do dispositivo

Tabela 2 – Colunas da tabela *Device*.

A Figura 4 mostra como os servidores devem ser organizados para melhor controle. O servidor nível 0 é a raiz da árvore e responsável por gerenciar os nós abaixo dele. A estrutura não é uma árvore binária e sim uma estrutura em árvore com N nós filhos. Quando um veículo que está em um servidor precisar mudar de servidor, a mudança deve ser solicitada ao servidor pai imediato, caso este não seja o servidor de destino, deve perguntar ao pai deste e assim sucessivamente até encontrar o servidor responsável pela área de destino ou o servidor nível 0 será responsável por este gerenciamento .

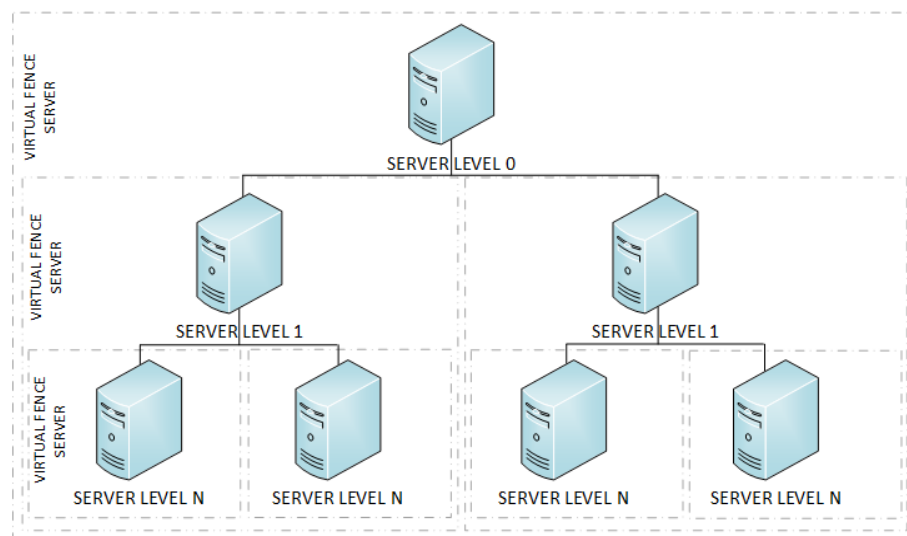


Figura 4 – Organização dos servidores na nuvem.



### 3.5 Módulo de Roteamento

Toda comunicação entre os nós é realizado entre os agentes virtuais da rede (AV2AV e AV2AI e AI2AI). As regras de roteamento podem ser substituídas ou expandidas, basta implementar novos algoritmos seguindo o modelo da arquitetura do módulo. Quando houver a necessidade de enviar algo para algum nó físico da rede, esta mensagem deve ser enviada através do Módulo Comunicação Veículo-Cloud ou Infra-Cloud e eles se responsabilizarão pela entrega.

O módulo de roteamento pode ser implementando de acordo com a necessidade, porém, para fins de avaliação da arquitetura, foi implementado o algoritmo que utiliza os RSU como cluster da rede. Podendo ser utilizando RSU físicos ou virtuais, ou seja, não é necessário implantar um RSU à beira da rodovia, basta definir pontos no mapa e este pode funcionar como um RSU virtual, tendo o comportamento de cluster da rede VANET.

Os veículos escolhem um RSU para participar da rede e a escolha é feita calculando o RSU mais próximo. Porém, a cada transmissão de uma nova coordenada pelo OBU, não é necessário buscar um novo RSU mais próximo, uma vez que é definido uma distância máxima para que o veículo permaneça na rede de uma RSU, quando essa distância rompida, aí é realizado novamente o processo de busca de um RSU mais próximo.

Adicionar Figuras

### 3.6 Módulo de Aplicações

Permite adicionar novas aplicações tendo como base os recursos disponibilizados pelos outros módulos. Porém, para que as aplicações possam enviar e receber informações da arquitetura, é necessário definir um contrato com as classes da arquitetura, através do uso de interfaces e classes abstratas. Sendo assim, quando uma informações chegar na arquitetura, será possível executar um método padronizado de uma classe da aplicação, fazendo com que os dados sejam entregues corretamente.

## 4 Avaliação da Arquitetura

Com o objetivo de avaliar a plataforma para redes veiculares desenvolvida, chamada de I9Vanet, principalmente no que diz respeito à sua capacidade de comunicação entre nós da rede, foram realizados alguns experimentos.

### 4.1 Cenário Proposto

O ambiente proposto para avaliação consiste em simular movimentações de veículos transmitindo e recebendo informações dos servidores que compoem a arquitetura I9Vanets.

Cada teste realizado, levou em consideração quantidades diferentes de dispositivos representando os veículos, sendo da seguinte forma: 10, 50, 100, 250, 500, 1000 e 2000. Para cada faixa, também foi ajustado a largura de banda para definir a velocidade máxima de comunicação de cada dispositivo, seguindo os modelos 2G, 3G e 4G.

Todo o ambiente, tanto o cliente quanto os servidores, foi montado em máquinas virtuais. No caso dos servidores, foram criados 7 máquinas com linux ubuntu server 16.04 com o postgresql 9.5 e plugin postgis 2.0 e Glassfish 4.1.1 como servidor de aplicação. Todos os servidores possuíam a mesma configuração, sendo 1 GB de Ram e 1 processador.

Para simulação dos clientes, os OBUs e RSUs, foi utilizado máquinas virtuais com linux ubuntu server 14.04 LTS 64 bits com o Mininet 2.2.1 configurado. Cada rede virtual Mininet instanciou no máximo 50 hosts, para que seu desempenho não fosse comprometido. Todas as máquinas com Mininet utilizaram a mesma configuração: 2GB de RAM e 2 processadores virtuais. O modelo do ambiente é mostrado na figura 5.

Foi criado um script em Python, conforme Código 4.1 com intuito de automatizar a criação dos hosts, limitar largura de banda a ser testada entre 2G, 3G e 4G, iniciar a execução da aplicação responsável para simular as funcionalidades de um OBU e RSU, realizar a comunicação com a arquitetura I9Vanet e registrar em log, os dados referente à comunicação de cada host, em arquivos individuais.

Um host virtual tem a função de representar um veículo real, inclusive com uso de movimentações reais, coletadas por 12 meses, de um sistema de monitoramento de veículos de uma empresa de taxi com 60 carros totalizando mais de 12 milhões de movimentações.

A base de movimentação foi extraída, separada por veículo e em seguida, armazenada em arquivo e colocada em cada máquina Mininet para que cada host pudesse simular uma

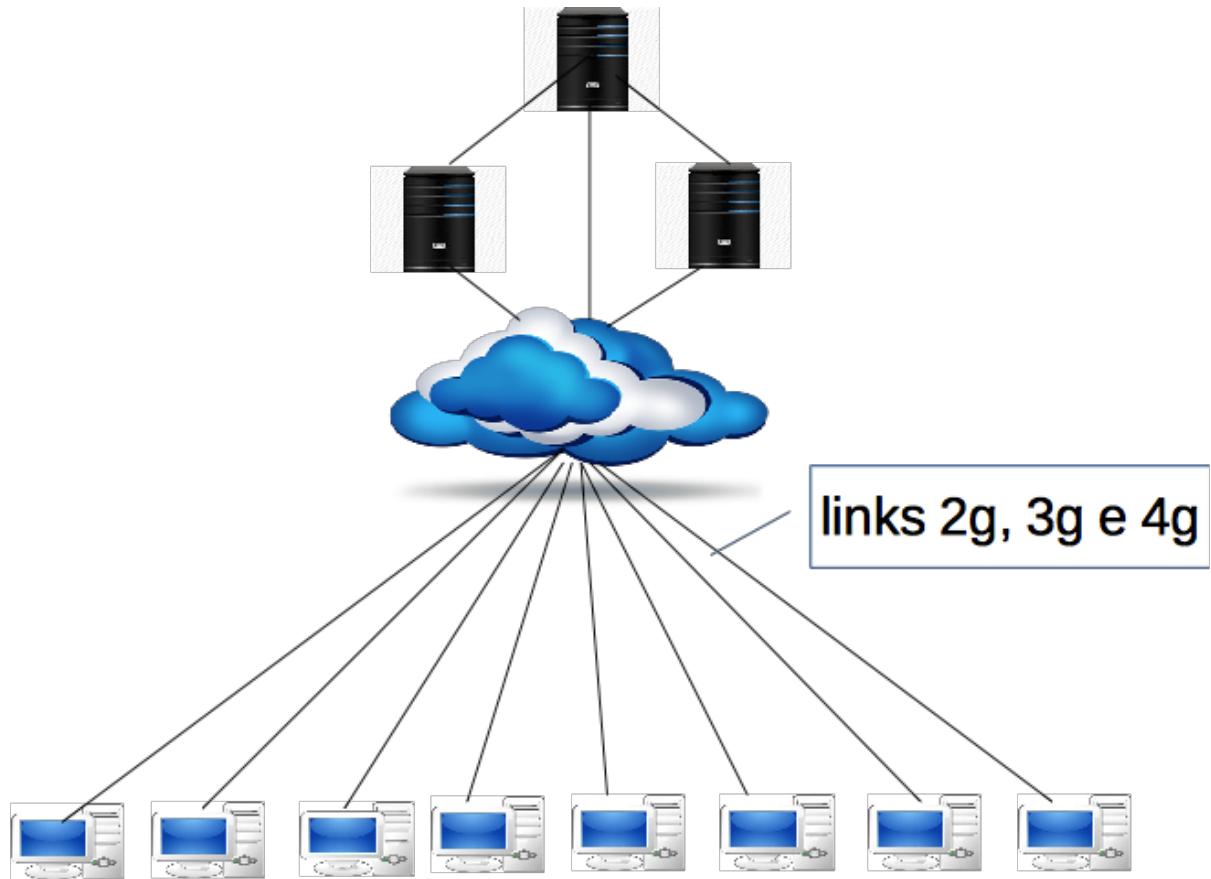


Figura 5 – Topologia de rede utilizada para a realização dos experimentos. Fonte: Criada pelo autor.

movimentação real. Cada arquivo de movimentação recebeu um número, por exemplo: 1.csv, 2.csv e assim por diante. Assim, cada host realizará um sorteio entre os arquivos disponíveis para ler as informações de movimentação e transmiti-la para o servidor. A tabela 3 mostra exemplo do arquivo de movimentação.

Latitude	Longitude	Velocidade
-10.9062183	-37.0619264	0
-10.906164143173685	-37.061813870099655	10.7
-10.904381347197752	-37.0671766923126	21.121
-10.904279005332516	-37.067784265675165	26.259
-10.904165425504841	-37.06846398328098	29.678

Tabela 3 – Linhas do arquivo de movimentação.

```

1  #!/usr/bin/python
2  ...
3  class SingleSwitchTopo(Topo):
4  def build(self, n=2):
5      switch = self.addSwitch('s1')
6      #Cria todos os hosts e conecta ao switch
7      for h in range(n):
8          host = self.addHost('h%s' % (h + 1))
9          self.addLink(host, switch)
10 def limit( bw=0.2, cpu=1 ):
11     intf = custom( TCIntf, bw=bw )
12     myTopo = SingleSwitchTopo(n=2)
13     for sched in 'rt', 'cfs':
14         if sched == 'rt':
15             release = quietRun( 'uname -r' ).strip('\r\n')
16             output = quietRun( 'grep CONFIG_RT_GROUP_SCHED /boot/config-%%
17                 s'
18             % release )
19             if output == '# CONFIG_RT_GROUP_SCHED is not set\n':
20                 continue
21             host = custom( CPULimitedHost, sched=sched, cpu=cpu )
22             net = Mininet( topo=myTopo, intf=intf, host=host)
23             net.addNAT().configDefault()
24             net.start()
25             # Para cada host criado, executa a aplicacao de simulacao
26             for host in net.hosts:
27                 if 'h' in host.name:
28                     host.cmd('java -jar SVVMMainTest.jar ' + str(host) + '.
29                         csv /home/mininet/vanet &')
30             CLI( net )
31 if __name__ == '__main__':
32     setLogLevel( 'info' )
33     limit()

```

Listing 4.1 – Script Python para criação dos hosts no mininet.

## 4.2 Experimentos

Os experimentos foram divididos em 21 partes, levando em consideração a quantidade de hosts clientes, 7, e a velocidade na comunicação, 3.

Para analisar os cenários propostos, foram coletadas as taxas de transmissão de todos os tráfegos gerados, além da utilização de cada um dos caminhos da rede, que consiste no total de *bytes* trafegando por cada um deles. Os caminhos presentes na rede são Caminho 1 (formado pelos *switches* SW1, SW2 e SW4) e Caminho 2 (formado pelos *switches* SW1, SW3 e SW4).

## 4.3 Resultados

## **5 Conclusão**

### **5.1 Conclusões deste Trabalho**

O presente trabalho apresentou o modelo de uma arquitetura para rede veiculares em nuvem, denominado i9Vanets, como também uma avaliação com intuito de medir a sua eficácia e eficiência, visando atender às demandas e desafios relacionados à VANETs. Este modelo permite montar uma rede veicular em nuvem e realizar todo gerenciamento e comunicação de maneira virtual, tornando os equipamentos OBUs e RSUs mais simples e baratos. Foram realizados alguns testes estatísticos, com objetivo de analisar o comportamento da arquitetura. Sobre o teste da normalidade da distribuição, tanto no cliente quanto no servidor, foi constatado que distribuição de probabilidade associada às amostras não podem ser aproximada. Outra análise foi relacionado à dispersão dos dados através do coeficiente de variação, o qual avaliou, sobre as amostras de latência, que a dispersão em cada teste define um comportamento semelhante, porém, o coeficiente de variação medido em cima do tempo de processamento no servidor, mostrou que os testes 50, 500 e 750, 1000 e 2000 tiveram grande dispersão. Como trabalhos futuros, é possível criar várias aplicações tais como: sistema de detecção de congestionamento de trânsito, permitindo um RSU avisar a outros RSU e/ou OBU; Controle de passagem livre para veículos de urgência e emergência; utilizar a arquitetura pra criar um simulador que possa simular toda uma cidade.

### **5.2 Trabalhos Futuros**

## Referências

- BALL, R.; DULAY, N. Enhancing traffic intersection control with intelligent objects. *Urban Internet of Things Towards Programmable Realtime Cities*, 2010. Nenhuma citação no texto.
- BUBENIKOVA, E.; DURECH, J.; FRANEKOVA, M. Security solutions of intelligent transportation system's applications with using vanet networks. In: IEEE. *Control Conference (ICCC), 2014 15th International Carpathian*. [S.l.], 2014. p. 63–68. Citado na página 17.
- CARVALHO, C. H. R. d. et al. Mobilidade urbana e posse de veículos: análise da pnad 2009. Instituto de Pesquisa Econômica Aplicada (Ipea), 2010. Citado na página 11.
- CAVALCANTI, S. R. *Veer: Um algoritmo de seleção de pares em redes ad hoc veiculares*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2008. Nenhuma citação no texto.
- CHEN, W. et al. Dynamic local peer group organizations for vehicle communications. In: IEEE. *Mobile and Ubiquitous Systems-Workshops, 2006. 3rd Annual International Conference on*. [S.l.], 2006. p. 1–8. Nenhuma citação no texto.
- DIB, R. P. E. et al. A systematic review of the interventions to promote the wearing of hearing protection. *São Paulo Medical Journal*, SciELO Brasil, v. 125, n. 6, p. 359–361, 2007. Citado na página 11.
- ELTOWEISSY, M.; OLARIU, S.; YOUNIS, M. Towards autonomous vehicular clouds. In: SPRINGER. *International Conference on Ad Hoc Networks*. [S.l.], 2010. p. 1–16. Nenhuma citação no texto.
- FALCHETTI, A.; AZURDIA-MEZA, C.; CESPEDES, S. Vehicular cloud computing in the dawn of 5g. In: IEEE. *2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. [S.l.], 2015. p. 301–305. Citado na página 16.
- FERNANDO, N.; LOKE, S. W.; RAHAYU, W. Mobile cloud computing: A survey. *Future Generation Computer Systems*, Elsevier, v. 29, n. 1, p. 84–106, 2013. Citado na página 15.
- GERLA, M. Vehicular cloud computing. In: IEEE. *Ad Hoc Networking Workshop (Med-Hoc-Net), 2012 The 11th Annual Mediterranean*. [S.l.], 2012. p. 152–155. Nenhuma citação no texto.
- GUPTE, S.; YOUNIS, M. Vehicular networking for intelligent and autonomous traffic management. In: IEEE. *Communications (ICC), 2012 IEEE International Conference on*. [S.l.], 2012. p. 5306–5310. Nenhuma citação no texto.
- HADALLER, D. et al. Vehicular opportunistic communication under the microscope. In: ACM. *Proceedings of the 5th international conference on Mobile systems, applications and services*. [S.l.], 2007. p. 206–219. Nenhuma citação no texto.
- HAJJI, T.; BARGAOUI, H. Design of a vanet testbed based on cloud computing. Nenhuma citação no texto.

HUSSAIN, R. et al. Rethinking vehicular communications: Merging vanet with cloud computing. In: IEEE. *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*. [S.l.], 2012. p. 606–609. Citado na página 15.

ICE, R. et al. *REGIONAL ITS ARCHITECTURE GUIDANCE: DEVELOPING, USING, AND MAINTAINING AN ITS ARCHITECTURE FOR YOUR REGION*. [S.l.], 2001. Citado na página 11.

JAKUBIAK, J.; KOUCHERYAVY, Y. State of the art and research challenges for vanets. In: IEEE. *Consumer communications and networking conference, 2008. CCNC 2008. 5th IEEE*. [S.l.], 2008. p. 912–916. Citado na página 17.

KATHI, S. *OpenFlow Wiki - CS244: Advanced Topics in Networking*. 2014. Disponível em: <http://yuba.stanford.edu/cs244wiki/index.php/Overview>. Nenhuma citação no texto.

LEE, E. et al. Vehicular cloud networking: architecture and design principles. *IEEE Communications Magazine*, IEEE, v. 52, n. 2, p. 148–155, 2014. Citado na página 15.

LI, C. Travel information service system for public travel based on soa. In: IEEE. *Service Operations and Logistics and Informatics (SOLI), 2010 IEEE International Conference on*. [S.l.], 2010. p. 321–324. Citado na página 11.

LIANG, W. et al. Vehicular ad hoc networks: architectures, research issues, methodologies, challenges, and trends. *International Journal of Distributed Sensor Networks*, Taylor & Francis, Inc., v. 2015, p. 17, 2015. Citado na página 12.

LIU, Y.-C.; CHEN, C.; CHAKRABORTY, S. A software defined network architecture for geobroadcast in vanets. In: IEEE. *2015 IEEE International Conference on Communications (ICC)*. [S.l.], 2015. p. 6559–6564. Nenhuma citação no texto.

LUÍS, N. M. A. *Melhoria de protocolos de encaminhamento em VANETs de alta densidade*. Tese (Doutorado) — FCT-UNL, 2009. Citado 3 vezes nas páginas 5, 17 e 18.

MATOS, L. B. C. d. et al. Análise de desempenho de algoritmos criptográficos assimétricos em uma rede veicular (vanet). *Ciência da Computação*, 2013. Citado na página 14.

MCKEOWN, N. et al. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 2, p. 69–74, 2008. Nenhuma citação no texto.

MERSHAD, K.; ARTAIL, H. Finding a star in a vehicular cloud. *IEEE Intelligent transportation systems magazine*, IEEE, v. 5, n. 2, p. 55–68, 2013. Citado na página 15.

NANDAN, A. et al. Co-operative downloading in vehicular ad-hoc wireless networks. In: IEEE. *Second Annual Conference on Wireless On-demand Network Systems and Services*. [S.l.], 2005. p. 32–41. Nenhuma citação no texto.

NASIM, R.; KASSLER, A. Distributed architectures for intelligent transport systems: A survey. In: IEEE. *Network Cloud Computing and Applications (NCCA), 2012 Second Symposium on*. [S.l.], 2012. p. 130–136. Citado na página 11.

PEDEN, M. et al. *World report on road traffic injury prevention*. [S.l.]: World Health Organization Geneva, 2004. Citado na página 11.



QIN, Y.; HUANG, D.; ZHANG, X. Vehicloud: Cloud computing facilitating routing in vehicular networks. In: IEEE. *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*. [S.l.], 2012. p. 1438–1445. Citado na página 14.

SAMARA, G.; AL-SALIH, W. A.; SURES, R. Security issues and challenges of vehicular ad hoc networks (vanet). In: IEEE. *New Trends in Information Science and Service Science (NISS), 2010 4th International Conference on*. [S.l.], 2010. p. 393–398. Citado na página 14.

SOUZA RONIEL DE; SOARES, A. C. B. Estimativa e sinalização de congestionamentos de tráfego através de redes veiculares v2v. *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, 2014. Citado na página 18.

SUMRA, I. A.; HASBULLAH, B.; MANAN, J. Comparative study of security hardware modules (edr, tpd and tpm) in vanet. In: *Third National Information Technology Symposium”, NITS 2011 Symposium*. [S.l.: s.n.], 2011. p. 6–9. Nenhuma citação no texto.

SYSTEMATICS, C.; MEYER, M. Crashes vs. congestion-what’s the cost to society. *Prepared for the American Automobile Association*, 2011. Citado na página 11.

VSWITCH, O. *Open vSwitch*. 2014. Disponível em: <http://openvswitch.org/>. Nenhuma citação no texto.

XU, Q. et al. Design and analysis of highway safety communication protocol in 5.9 ghz dedicated short range communication spectrum. In: IEEE. *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*. [S.l.], 2003. v. 4, p. 2451–2455. Citado na página 11.

YAN, G. et al. Security challenges in vehicular cloud computing. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 14, n. 1, p. 284–294, 2013. Nenhuma citação no texto.

ZHU, T.; LIU, Z. Intelligent transport systems in china: Past, present and future. In: IEEE. *Measuring Technology and Mechatronics Automation (ICMTMA), 2015 Seventh International Conference on*. [S.l.], 2015. p. 581–584. Citado na página 11.