TRABALHO DE GRADUAÇÃO

ESTUDO DAS VULNERABILIDADES DE TECNOLOGIAS SEM FIO UTILIZADAS EM AMBIENTES IOT

Catharina Daher Teixeira
Mariana de Lacerda Clarim

Brasília, 08 de Dezembro de 2017

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA
UNIVERSIDADE DE BRASÍLIA

Universidade de Brasília
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

ESTUDO DAS VULNERABILIDADES DE TECNOLOGIAS SEM FIO UTILIZADAS EM AMBIENTES IOT

Catharina Daher Teixeira
Mariana de Lacerda Clarim

Relatório submetido ao Departamento de Engenharia Elétrica como requisito parcial para obtenção Do grau de Engenheiro de Redes de Comunicação.

Banca Examinadora

Prof. Georges Daniel Amvame Nze, Dr., UnB/ENE
Orientador

Fábio Lúcio Lopes de Mendonça, MSc., UnB/EnE
Examinador Interno

Prof. Robson de Oliveira Albuquerque, Dr., UnB/EnE
Examinador Externo

AGRADECIMENTOS

Agradecemos, primeiramente, a Deus que foi nosso alicerce para realização deste trabalho, dando-nos, diariamente, a força para seguir em frente.

Ao Professor Georges Daniel Amvame Nze que desempenhou o papel de orientador e, com palavras sábias e críticas valiosas, transmitiu-nos seus conhecimentos acadêmicos e pessoais.

Ao coorientador Robson de Oliveira Albuquerque que nos despertou interesse pelo tema e sempre esteve à disposição para apoiar, incentivar e compartilhar experiências que nos engrandeceram.

Agradecemos, imensamente, aos nossos pais, Marco Aurélio e Carla, Cristina e em especial ao Gilvan que não está mais entre nós, mas nunca mediu esforços para dar todo o apoio necessário durante o tempo que esteve em vida, e, também, nossos irmãos, Carlos Eduardo e Isabel, que são nossa base familiar, dando-nos o incentivo necessário desde o início da educação até o ensino superior do curso de Engenharia de Redes de Comunicação.

Agradecemos, especialmente, ao Bruno e Túlio por serem grandes companheiros, apoiadores e pelo auxílio ao longo do curso.

Aos nossos amigos de curso que nos acompanharam ao longo da caminhada na Universidade de Brasília e foram capazes de transformar o cotidiano de estudo.

Agradecemos, também, aos colegas do laboratório UIoT da Universidade que nos disponibilizaram o ambiente e informações necessárias para realização do trabalho.

RESUMO

Este trabalho aborda o universo da Internet of Things (IoT), redes que estão cada vez mais presentes no cotidiano da sociedade, na qual dispositivos tais como televisores ou uma caixa de som estão conectados e transmitem dados através de redes de comunicação. Por este motivo, os problemas relacionados à segurança em redes IoT crescem à medida que as mesmas evoluem. A finalidade deste trabalho tem como base os padrões de comunicação sem fio Wi-Fi, Bluetooth e ZigBee definidos, respectivamente, pelo IEEE 802.11, IEEE 802.15 e IEEE 802.15.4. Dessa forma, o presente trabalho tem como tem como objetivo explorar as vulnerabilidades existentes em redes IoT que fazem uso dos padrões IEEE 802.11, IEEE 802.15 e IEEE 802.15.4. As análises dessas vulnerabilidades serão realizadas através de dispositivos e ferramentas como Kali Linux e Wireshark que permitem a realização de diferentes ataques como quebra de chave, interceptação de dados, negação de serviço e escaneamento da rede. Como resultados, são propostas soluções para minimizar a possibilidade de um usuário tornar-se alvo destes ataques.

Palavras-chave: IoT. Wi-Fi. Bluetooth. ZigBee. Quebra de chave. Negação de serviço. Interceptação de dados. Escaneamento da rede.

ABSTRACT

This work approaches the universe of Internet of Things (IoT), networks that are progressively more apparent in society, where simple objects such as a television or a sound system, connect and transmit data through IP networks. Due to this, problems related to the security aspect of IoT networks grow at the same pace as their evolution. The basis of this work are the Wi-Fi, Bluetooth and ZigBee communication standards: IEEE 802.11, IEEE 802.15 and IEEE 802.15.4. Thus, the present work aims to exploit vulnerabilities in IoT networks that make use of the IEEE 802.11, IEEE 802.15 and IEEE 802.15.4 standards. Vulnerability analyzes will be addressed through the use of devices and tools such as, Kali Linux and Wireshark which permit the emulation of various attacks, for instance password cracking, data interception, denial of service and network scanning. As a result, solutions are proposed to minimize the possibility of a user becoming the target of these attacks.

Keywords: IoT. Wi-Fi. Bluetooth. ZigBee. Password cracking. Denial of service. Data interception. Network scanning.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURA E SIGLAS

ACK – *Acknowledgement*

AES – *Advanced Encyptation Standart*

AP – *Access Point*

ARP – *Address Resolution Protocol*

BSS – *Basic Service Set*

CBC-MAC – *Cipher Block Chaining Message Authentication Code*

CCMP – *Counter Mode CBC MAC Protocol*

CERP – *Cluster of European Research Projects*

CRC – *Cyclic Redundancy Check*

CSMA-CA – *Carrier Sense Multiple Access with Collision Avoidance*

CTR – *Counter*

DIFS – *Function Inter Frame Space*

DoS – *Denial of Service*

DSSS – *Direct Sequence Spread Spectrum*

EAP – *Extensible Authentication Protocol*

ESS – *Extended Service Set*

FHSS – *Frequency Hopping Spread Spectrum*

GCMP – *Galois/Counter Mode Protocol*

GPS – *Global Positioning System*

GTK – *Group Temporal Key*

IEEE – *Institute of Electricaland Eletronics Engineers*

IETF – *Internet Engineering Task Force*

IoT – *Internet of Things*

IP – *Internet Protocol*

ISM – *Industrial, Scientific, Medical*

IV – *Initialization Vector*

KRACKs – *Key Reinstallation Attacks*

KSA – *Key Scheduler Algorithm*

LANs – *Local Area Network*

LMP – *Link Manager Protocol*

MAC – *Message Authentication Code*

MIC – *Message Integrity Code*

MK – *Master Key*

MQTT – *Message Queue Telemetry Transport*

OSI – *Open System Interconnection*

OWASP – *Open Web Application Security Project*

PIN – *Personal Identification Number*

PLCP – *Physical Layer Convergence Protocol*

PMD – *Physical Medium Dependent Sublayer*

PMK – *Pairwise Master Key*

PPDU – *PHY Protocol Data Unit*

PPP – *Point-to-Point Protocol*

PRGA – *Pseudo Random Generation Algorithm*

PSK – *Pre-Shared Key*

PTK – *Pairwise Transient Key*

RADIUS – *Remote Authentication Dial In User Service*

RF – Rádio Frequência

RSSF – Redes de Sensores Sem Fio

RSSI – *Received Signal Strenght Indicator*

SIFS – *Short Inter Frame Spacing*

SOAP – *Simple Object Access Protocol*

SSID – *Service Set Identifier*

STA – *Station*

TCP – *Trasmission Control Protocol*

TDD – *Time Division Duplexing*

TDMA – *Time Division Multiple Access*

TK – *Temporal Key*

TKIP – *Temporal Key Integrity Protocol*

TLS – *Transport Layer Security*

TMK – *Temporal* MIC *Key*

UIoT – *Universal Internet of Things*

WAP – *Wireless Application Protocol*

WEP – *Wired Equivalent Privacy*

WLAN – *Wireless Local Area Network*

WSA – *Web Services Addressing*

WSD – *Web Services Dynamic Discovery*

WPA – *Wi-Fi Protected Access*

WPAN – *Wireless Personal Area Network*

XML – *eXtensible Markup Language*

SUMÁRIO

1. INTRODUÇÃO

Este trabalho consiste no estudo das vulnerabilidades das tecnologias de comunicação móveis *Bluetooth,* Wi-Fi e *ZigBee*, que são alguns dos padrões utilizados na *Internet of Things* (IoT). A IoT, segundo a CERP (*Cluster of European Research Projects*) (2009), pode ser definida como uma infraestrutura de rede global dinâmica com capacidade de configuração através de protocolos de comunicação. As "coisas" possuem atributos e identidades próprias e utilizam interfaces inteligentes na rede de informações.

As "coisas" devem ser capazes de interagir e comunicar entre si e com o ambiente, trocando dados e informações, enquanto reagem de forma autônoma aos eventos do mundo real, influenciando-o através da execução de processos que desencadeiam ações e criam serviços com ou sem intervenção humana. Esses objetos são limitados em recursos como energia, capacidade de processamento e armazenamento, alcance do rádio, dentre outros (CERVANTES, 2014).

Cervantes (2014) ainda ressalta que os dispositivos conectados em uma rede IoT são extremamente vulneráveis a ataques devido às limitações computacionais que os mesmos possuem e por isso é necessário utilizar de mecanismos para garantir a segurança. Os requisitos de segurança são: confidencialidade, que deve garantir que uma informação seja acessível somente por dispositivos autorizados dentro da rede; integridade, que deve garantir que os dados não sofreram modificações entre a origem e o destino; escalabilidade, que assegura que a rede IoT se adapte ao aumento do número de dispositivos conectados à rede; e disponibilidade, que garante que um dispositivo possua capacidade de acessar à rede. A rede IoT deve levar em conta questões de segurança e privacidade, que é o foco principal deste projeto.

Dessa forma, o trabalho explora as vulnerabilidades identificadas nos padrões de comunicação sem fio e, a partir da realização de ataques à ambientes Wi-Fi e Bluetooth, pode-se analisar cada cenário abordado e, também, definir as recomendações de proteção mínima para os mesmos, como forma de minimizar a chance de sucesso dos ataques.

1.1 Descrição do problema

Este trabalho expõe os problemas comuns encontrados em tecnologias utilizadas frequentemente em dispositivos presentes em redes IoT, como Wi-Fi, *Bluetooth* e *ZigBee* que

são padrões descritos pelo IEEE – *Institute of Electrical and Eletronics Engineers,* respectivamente, IEEE 802.11, IEEE 802.15 e IEEE 802.15.4.

Em se tratando da crescente utilização de dispositivos cotidianos inseridos em uma rede, há necessidade de evidenciar formas de ataques possíveis ao universo IoT. Dessa forma, as vulnerabilidades podem ser exploradas mediante ataques que afetam as informações e o funcionamento da rede WLAN ou WPAN em que os dispositivos estão inseridos. Dispositivos estes que são utilizados por usuários que, de maneira geral, não possuem conhecimento sobre maneiras simples de minimizar a chance tornar-se vítima de ataques.

1.2 Objetivos

1.2.1 Objetivo geral

Este trabalho tem como objetivo explorar os ataques a rede IoT utilizando os padrões de comunicação utilizados com frequência na comunicação entre dispositivos IoT, são eles: IEEE 802.11, 802.15 e 802.15.4, que correspondem, respectivamente, às tecnologias Wi-Fi, *Bluetooth* e *ZigBee*.

1.2.2 Objetivos específicos

Como forma de atingir o objetivo final do projeto, os objetivos específicos são:

- Caracterizar as arquiteturas e o funcionamento dos padrões IEEE 802.11, 802.15 e 802.15.4;
- Identificar as vulnerabilidades dos principais padrões utilizados em ambientes IoT mencionados;
- Identificar as ferramentas utilizadas para realização dos ataques;
- Simular ataques aos padrões escolhidos em ambiente IoT;
- Definir formas de proteção mínima para os ataques.

1.3 Justificativa do estudo

Para a comunidade acadêmica, o estudo terá importância, pois contribuirá para identificação de possíveis ataques aos dispositivos IoT disponíveis não só no laboratório UIoT

(*Universal Internet of Things*) como também nos laboratórios da Universidade e como forma de melhoria dos mecanismos de defesa. Para a sociedade, o estudo pode ser ampliado e utilizado como referência de informação sobre os possíveis ataques e vulnerabilidades dos padrões abordados em redes IoT implementadas na sociedade.

1.4 Organização do trabalho

O trabalho é organizado por cinco capítulos. O capítulo 1 apresenta ao leitor uma visão geral sobre o que será tratado no decorrer do trabalho, bem como os objetivos específicos e geral. O capítulo 2 expõe conceitos e definições descritos em livros, artigos, sites e outras fontes essenciais para o entendimento dos padrões estudados. Consiste no detalhamento de arquiteturas e funcionamento das tecnologias usualmente implementadas em dispositivos IoT, Wi-Fi, *Bluetooth* e *ZigBee*. O capítulo 3 aborda a metodologia utilizada no trabalho. Detalha as características principais do projeto, como o universo da pesquisa e o tipo de projeto desenvolvido. Neste capítulo são expostas as etapas de realização do projeto, bem como os cenários de simulação dos ataques realizados em ambiente controlado e no laboratório UIoT. No capítulo 4 são mostrados os resultados e análises. Este capítulo apresenta os ataques executados como forma de análise às tecnologias abordadas neste projeto. São expostos os procedimentos realizados em cada cenário configurado, os resultados obtidos, a análise dos mesmos e, também as propostas de solução. Expõe as formas conhecidas de proteção para minimizar a possibilidade de sucesso dos ataques abordados. Por fim, o capítulo 5 apresenta as conclusões obtidas a partir dos resultados dos procedimentos realizados na execução dos ataques. Apresenta, também, possíveis trabalhos a serem realizados que possuam este projeto como base.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo expõe conceitos e definições essenciais para sobre o universo da IoT e o entendimento dos padrões de comunicação abordados neste trabalho. Detalha as arquiteturas, o funcionamento e questões importantes de segurança das tecnologias utilizadas em dispositivos que se comunicam em redes WLAN (*Wireless Local Area Network*) e WPAN (*Wireless Personal Area Network*), são elas: Wi-Fi, *Bluetooth* e *ZigBee*.

2.1 Internet of Things

A IoT pode ser considerada uma extensão da Internet atual e faz com que seja possível que objetos comuns, "coisas", com capacidade de comunicação e processamento, se conectem à rede (SANTOS et al, 2016). Esses objetos possuem nova utilidade e é possível controlá-los, realizar trocas de informações entre eles, acessar serviços da Internet e interagi-los com as pessoas. Para isso, é necessário que esses objetos possuam inteligência para as tomadas de decisão.

Segundo Santos et al (2016), a IoT possui blocos básicos para que ela seja construída. São eles: identificação, que consiste em identificar os objetos de forma única para conectá-los à rede; sensores, que coletam informações sobre o contexto que os objetos estão inseridos e as armazenam; comunicação, que consiste nas técnicas para conectar objetos inteligentes; computação, que inclui a unidade de processamento; serviços, que compõem os diversos serviços que o IoT pode prover; e semântica, que se refere a capacidade de inteligência dos objetos na IoT.

A previsão é que o IoT cresça muito nos próximos anos (MAGALHÃES, 2016). Para isso, é necessário que esse aumento seja acompanhado por uma infraestrutura capaz de suportar o alto tráfego, armazenamento e processamento dos dados, com segurança e eficiência. Porém, o risco também aumenta e a preocupação com a segurança se faz necessária.

O OWASP (*Open Web Application Security Project*) disponibilizou uma lista que contém os principais problemas de segurança em IoT. São eles: interface *web* insegura, autenticação insuficiente, serviços de rede inseguros, falta de criptografia no transporte, preocupações com a privacidade, interface de nuvem insegura, interface móvel insegura, configuração de segurança insuficiente, *software* inseguro e falta de segurança física.

As redes em ambientes IoT são predominantemente sem fio. As mais utilizadas são do tipo WPAN, que tem como exemplo o *Bluetooth* e *ZigBee*, e WLAN, na qual pode-se citar o Wi-Fi. Com isso, todos as vulnerabilidades das redes sem fio se estendem à IoT.

Dispositivos IoT podem ser encontrados em muitos lugares e fornecem serviços ao ambiente em que estão inseridos. Eles podem criar uma falsa impressão de segurança e instigar os indivíduos a divulgar informações pessoais sem a noção real das consequências de como divulgação de dados poderia afetá-los e, também, torná-los alvo de ataques, pois o atacante, em posse de informações pessoais, pode direcionar determinados tipos de ataques aos usuários de acordo com os dados divulgados.

Dessa forma, a questão de privacidade no universo da IoT é um ponto a ser tratado com cuidado. É importante conciliar requisitos de funcionalidade e privacidade nas diferentes fases de desenvolvimento e operação de um produto IoT, pois na grande rede de dispositivos conectados pela IoT, e entende-se que parte desses dispositivos são projetados para a coleta de dados no ambiente em que está inserido, e por meio dessa coleta, são incluídos dados relacionados às pessoas. Portanto, torna-se fundamental a proteção dos dados recolhidos pelos dispositivos que são de natureza pessoal. (FIGUEIRA, 2016).

2.2 Wi-Fi

Uma das principais tecnologias sem fio utilizada em ambientes IoT é o Wi-Fi, baseada no padrão IEEE 802.11. Neste tipo de rede, a transmissão é feita por sinais de radiofrequência, que se propagam pelo ar e possuem um alcance maior se comparado aos outros protocolos e pode chegar a dezenas de metros. Os sinais são transmitidos nas frequências de 2,4GHz ou 5GHz, o que permite uma maior transmissão de dados (JOBSTRAIBIZER, 2010).

O IEEE 802.11 atua nas duas primeiras camadas do modelo OSI (*Open System Interconnection*): física e de enlace (FERNANDES, 2006). A camada física é responsável pela definição dos meios de transmissão e recepção. Ela define três tecnologias para codificação da transmissão: infravermelho, FHSS (*Frequency Hopping Spread Spectrum*) e DSSS (*Direct Sequence Spread Spectrum*). A camada física é dividida em duas subcamadas. A PMD (*Physical Medium Dependent Sublayer*) define e especifica as funções para um canal físico específico, além de definir a modulação e codificação do canal. A PLCP (*Physical Layer Convergence Protocol*) possibilita a interconexão entre a PMD e a subcamada MAC, sendo elas independentes. A camada de enlace, ou camada MAC, determina o modo de

acesso ao meio e como enviar a informação e define a interface entre a máquina e a camada física.

Para uma rede Wi-Fi ser estabelecida, uma estação, conhecida como STA (*Station*), deve se conectar a dispositivos conhecidos como AP (*Access Point*), que fornecem o acesso à rede. Após a conexão entre a STA e o AP, é formada uma rede denominada BSS (*Basic Service Set*) (EDUARDO, 2011). A BSS é a topologia básica de redes Wi-Fi e necessita de apenas um AP e um ou mais STAs para ser formada. Cada BSS deve receber um SSID (*Service Set Identifier*), um conjunto de caracteres inserido no cabeçalho de cada pacote de dados da rede, para a identificação e segurança da mesma (ALECRIM, 2013).

A BSS possui baixa capacidade de cobertura, porém é possível criar redes sem fio que abrangem maior área através do agrupamento de várias BSSs em uma ESSs (*Extended Service Set*), segundo Fernandes (2006). Todos os AP na ESS possuem o mesmo SSID e estão ligados pela mesma rede cabeada. Todas as estações dentro de uma ESS, mesmo que não estejam na mesma BSS, podem se comunicar.

Existem protocolos que são responsáveis pela segurança na rede. Os principais utilizados são WEP, WPA e WPA2. A diferença entre eles está na criptografia e chaves utilizadas e possuem vantagens e desvantagens entre si.

2.2.1 WEP

O protocolo WEP (*Wired Equivalent Privacy*) é o algoritmo de privacidade criptográfica opcional especificado pelo IEEE 802.11 e utilizado para fornecer confidencialidade dos dados (IEEE, 1994). Foi criado como pioneiro na proteção de dados em redes sem fio (PAIM, 2011). Ele atua na camada de enlace e a segurança fornecida por ele é semelhante a uma rede cabeada, por isso recebe este nome. Ele possui muitas falhas e tornou-se obsoleto no quesito segurança, porém ainda é utilizado por usuários que desconhecem os aspectos que o tornam uma má escolha em uma rede sem fio.

O funcionamento do WEP, mostrado na figura 1, se dá através da autenticação, encriptação e decriptação de dados. A criptografia ocorre no tráfego do canal de comunicação sem fio e não atua no tráfego roteado para fora da rede (DUARTE, 2010). O algoritmo RC4 é utilizado pelo WEP para a criptografia dos dados. O RC4 foi criado por Ronald Rivest e, inicialmente, foi mantido em segredo até 1994, quando foi vazado em páginas na Web. Este algoritmo gera um fluxo de chaves que, ao sofrer uma operação XOR com um texto simples, forma um texto cifrado.

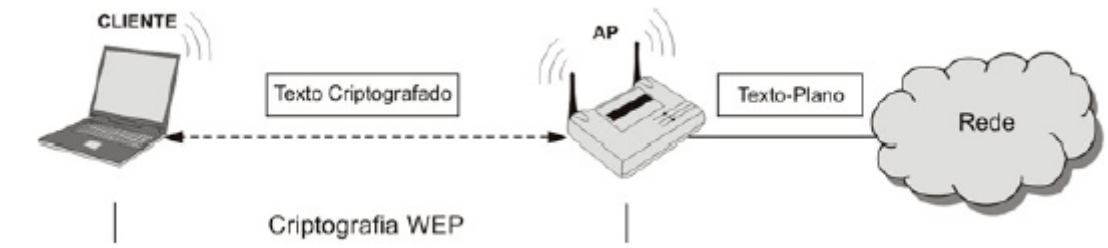

Figura 1 – Funcionamento do WEP.
Fonte: DUARTE, 2010.

Segundo Paim (2011), o KSA (*Key Scheduler Algorithm*) é um algoritmo que tem como função gerar uma permutação pseudo-aleatória de conteúdo de uma chave secreta. Já o PRGA (*Pseudo Random Generation Algorithm*) é responsável pela encriptação da mensagem a partir do valor retornado pelo KSA. Para isso, ele realiza operações XOR entre a permutação da chave secreta e a mensagem, formando uma mensagem cifrada. A permutação realizada pelo KSA deve ser diferente a cada mensagem enviada e por isso é utilizado um IV (*Initialization Vector*) pseudo-aleatório que é recalculado a cada iteração e acrescido a chave secreta. Este IV é anexado ao texto cifrado que será enviado. O WEP utiliza um IV de 24 bits que é anexado a uma chave simétrica de 40 ou 104 bits para criar uma chave de 64 ou 128 bits, que é utilizada para criptografar o fluxo de dados (PINZON, 2009).

No processo de encriptação, figura 2, o valor da sequência do IV é calculado e concatenado com a chave secreta compartilhada entre o remetente e destinatário. O valor da permutação é gerado pelo KSA. A mensagem original é dividida a fim de caber em um quadro WEP e o valor de *hash* é calculado através do CRC (*Cyclic Redundancy Check*) de 32 bits. O CRC é um algoritmo que gera um identificador único que é utilizado para verificar se os dados recebidos são os mesmos que enviados. O CRC é adicionado à mensagem como um verificador de integridade. Após isso, é aplicado o algoritmo PRGA sobre a mensagem e seu valor de *hash*. O resultado deste processo é concatenado com o valor atual do IV e enviado (PAIM, 2011).

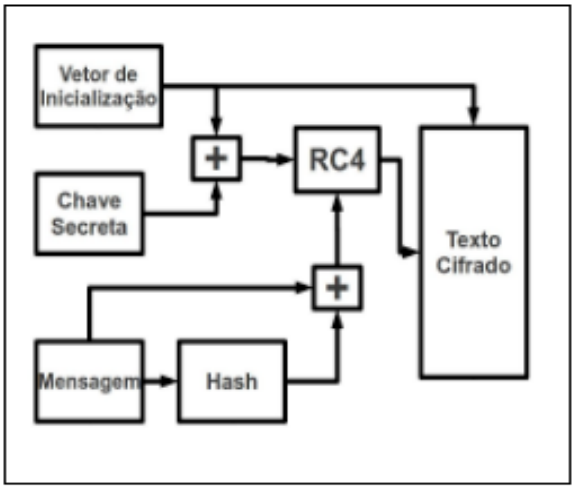

Figura 2 – Encriptação no WEP.
Fonte: PAIM, 2011.

O processo de decriptação, figura 3, da mensagem ocorre de forma reversa ao de encriptação. O IV é separado da mensagem cifrada e concatenado com a chave secreta compartilhada. Em seguida, ele passa novamente pelo KSA e é calculada a mensagem original acrescida do *hash* no PRGA. Então, é aplicado o mesmo CRC para comparar a mensagem enviada e recebida. Se houver diferenças nos resultados, é solicitado o reenvio do quadro. Se forem iguais, uma mensagem de confirmação de recebimento é enviada.

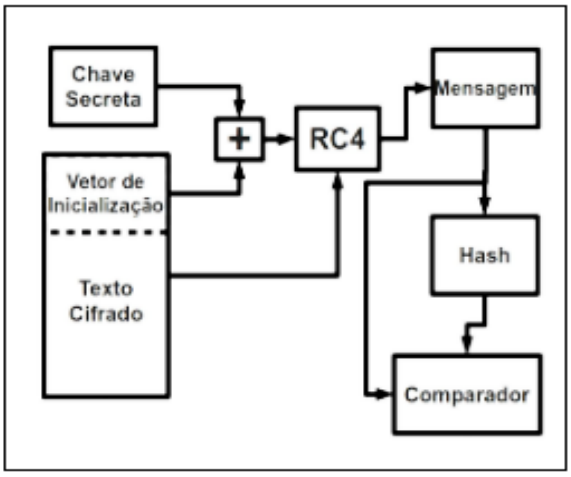

Figura 3 – Decriptação no WEP.
Fonte: PAIM, 2011.

2.2.1.1 Autenticação no WEP

O WEP utiliza dois modos para autenticação: *open system* e *shared key* (PAIM, 2011). No *Open System*, sistema aberto, basta que o usuário conheça o SSID da rede para realizar a autenticação. A máquina que deseja se conectar a ela deve realizar uma requisição

junto com o SSID. Se o mesmo estiver correto, o AP envia uma resposta positiva de conexão, caso contrário, uma resposta negativa é enviada.

No *Shared Key*, chave compartilhada, a conexão só é realizada se, além do SSID, o usuário conhecer a chave secreta da rede. Quando o acesso à rede é solicitado, o AP envia uma mensagem composta de um número inteiro aleatório. A máquina deve responder com uma mensagem que contém o valor correto deste número encriptado, de acordo com o processo de encriptação. O AP, ao receber a mensagem, realiza o processo de decriptação. Se o valor decriptado for igual ao original, o AP envia uma mensagem de confirmação de conexão, caso contrário, uma mensagem de falha de conexão.

2.2.1.2 Problemas no WEP

O WEP é um protocolo que possui diversas vulnerabilidades que podem ser exploradas por usuários com tendência maliciosa. Um dos grandes problemas do IV no WEP é sua reutilização e, por possuir 24 bits, não é necessário muito tempo para que ele seja quebrado (REIS, 2008). Além disso, sempre que houver a repetição de um IV, a sequência pseudo-aleatória gerada pelo IV e a chave secreta será também repetida, tornando mais fácil sua quebra.

Um grande problema está no fato de que todos os usuários da rede devem conhecer essa chave para realizar a conexão. Com isso, é mais difícil manter essa informação em sigilo e ela pode ser exposta a invasores. Outro problema relacionado às chaves é que devido ao seu compartilhamento entre vários usuários, a chance de haver uma colisão entre IVs é muito maior.

O CRC possui linearidade e, como consequência, o pacote pode sofrer alterações que passam despercebidas pelos dispositivos emissores e receptores. Se o atacante souber a *string* pseudo-aleatória utilizada na encriptação do texto, ele pode alterar a mensagem original. A informação pode ser interceptada e alterada de acordo com o desejo do atacante sem que isto seja detectado e sem alterar o verificador de integridade (REIS, 2011).

2.2.2 WPA

Devido às vulnerabilidades existentes no WEP, a Wi-Fi Alliance e o IEEE desenvolveram, em 2002, uma nova especificação de segurança Wi-Fi, o WPA (*Wi-Fi Protected Access*) que aumenta consideravelmente o nível de proteção de dados. *(*Wi-Fi Alliance, 2003). Ele possui melhores mecanismos de autenticação, privacidade e integridade e

foi projetado para rodar em *hardwares* baseados em WEP, ou seja, funciona como um *upgrade* de *firmware* e não necessita de alterações na infraestrutura de *hardware* (PINZON, 2009).

Uma das melhorias do WPA em relação ao WEP é a utilização do RC4 dentro do protocolo TKIP (PAIM, 2011). O TKIP utiliza chaves de 128 bits e IV de 64 bits e uma chave temporária que é resultado da combinação entre a chave compartilhada do AP e do cliente e o endereço MAC da placa de rede do cliente, o que torna a chave única para cada cliente que se conecta à rede (DUARTE, 2010). Essa chave é dita temporária por ser alterada periodicamente. Esse tempo pode ser programado manualmente ou ela é trocada, por padrão, a cada dez mil quadros. Isto garante mais segurança, pois se o atacante quebrar essa chave, ela só será válida em um determinado intervalo de tempo.

Segundo Paim (2011), o MIC (*Message Integrity Checksum*) é utilizado pelo TKIP como um verificador de integridade, que evita a alteração de pacotes durante a transmissão. O TKIP utiliza metade da chave do MIC, chamada TMK (*Temporal* MIC *Key*), para embaralhar o conteúdo da mensagem original com o endereço MAC de origem e destino e retorna um valor de oito bytes após algumas operações. O computador e o AP utilizam metades distintas do TMK para o cálculo do MIC. O receptor e o transmissor comparam o MIC, se for diferente, é entendido que houve uma alteração dos dados (DUARTE, 2010).

2.2.2.1 Autenticação no WPA

O TKIP utiliza o protocolo EAP (*Extensible Authentication Protocol*) para realizar a autenticação. Esse protocolo autentica cada usuário que deseja se conectar a rede e possibilita inúmeras formas de autenticação. O EAP, mostrado na figura 4, é composto de três elementos: um cliente, um AP e um servidor de autenticação, que trabalha com o protocolo RADIUS (*Remote Authentication Dial In User Service*). O cliente solicita o acesso à rede através do AP, que é responsável pela autenticação. O AP busca as informações do cliente no servidor de autenticação e se elas estiverem corretas, é concedido o acesso do cliente à rede (DUARTE, 2010).

Segundo Paim (2011), este processo é composto de quatro mensagens: requisição, resposta, sucesso e falha. O primeiro passo consiste no envio de uma mensagem de requisição pelo suplicante ao AP. Ele retorna uma mensagem com um pedido de identidade do suplicante. Ao receber a resposta do mesmo, o AP a envia para o servidor RADIUS. Um desafio é criado e o suplicante deve utilizar a senha que possuir para passá-lo. Se estiver

correta, ele é conectado à rede, caso contrário, é retornada uma mensagem de falha na conexão. Após isso, é feito um acordo entre o AP e o suplicante para decidir os valores das chaves temporais. O fato de não haver comunicação direta entre o cliente e o servidor é uma forma de garantir a segurança do mesmo.

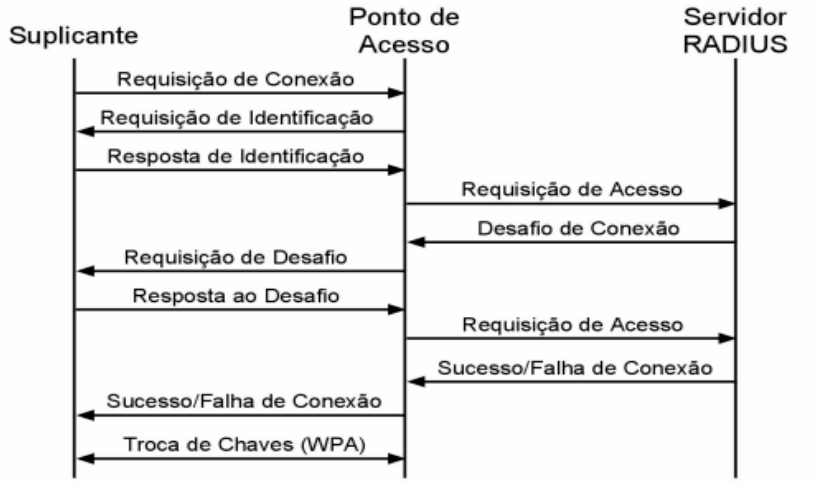

Figura 4 – Autenticação EAP.
Fonte: PAIM, 2011.

2.2.2.2 Vantagens e desvantagens do WPA

Segundo Duarte (2010), as vantagens do WPA é que ele trabalha com criptografia dinâmica e autenticação mútua, que oferecem mais segurança no acesso à rede, se comparado com o WEP. Cada usuário possui uma senha única e a mesma muda periodicamente como prevenção de intrusos na rede, enquanto o WEP utiliza uma chave fixa e conhecida por todos da rede. Além disso, permite trabalhar em redes híbridas que possuem WEP ou WPA2, necessitando apenas de atualização de *software.*

Porém, o WPA também possui suas vulnerabilidades, apesar de ter solucionado grande parte dos problemas encontrados no WEP. Ele ainda utiliza o RC4 como algoritmo de criptografia mesmo com a existência de outros mais fortes. Seu algoritmo de combinação de chaves é considerado fraco. O MIC é eficiente contra ataques de força bruta, porém é suscetível a ataques de negação de serviço (DoS–*Denial of Service*). Se forem detectados dois erros em um intervalo de tempo pequeno, a chave de integridade é alterada e por isso com uma simples injeção de pacotes forjados é possível realizar o ataque de DoS (DUARTE, 2010).

2.2.3 WPA2

Em 2004, o WPA2 foi desenvolvido pela Wi-Fi Alliance baseado no IEEE 802.11i (Wi-Fi Alliance, 2012). É apresentado como uma versão nova e aprimorada do WPA e utiliza o protocolo CCMP (*Counter Mode CBC MAC Protocol*) em conjunto com o AES para realizar a criptografia, que é mais forte que o RC4 com o TKIP, e o EAP também é utilizado para a autenticação (DUARTE, 2010).

O AES é um algoritmo de criptografia simétrica de cifra de bloco e suporta chaves de 128, 192 ou 256 bits e utiliza um IV de 48 bits. O WPA2 utiliza chave de 128 para implementar o AES. Ele funciona em rodadas com operações de permutações e combinações de bits e por isso pode ser calculado rapidamente (PAIM, 2011). O CCMP é um modo de operação que habilita uma única chave para ser utilizada na criptografia e autenticação. Ele utiliza dois modos: CTR (*Counter*), que é utilizado para criptografia dos dados, e CBC-MAC (*Cipher Block Chaining Message Authentication Code*), que é utilizado para verificar a integridade. O CBC-MAC, diferentemente do MIC, gera um componente de autenticação como resultado do processo de criptografia, como mostrado na figura 5 (DUARTE, 2010).

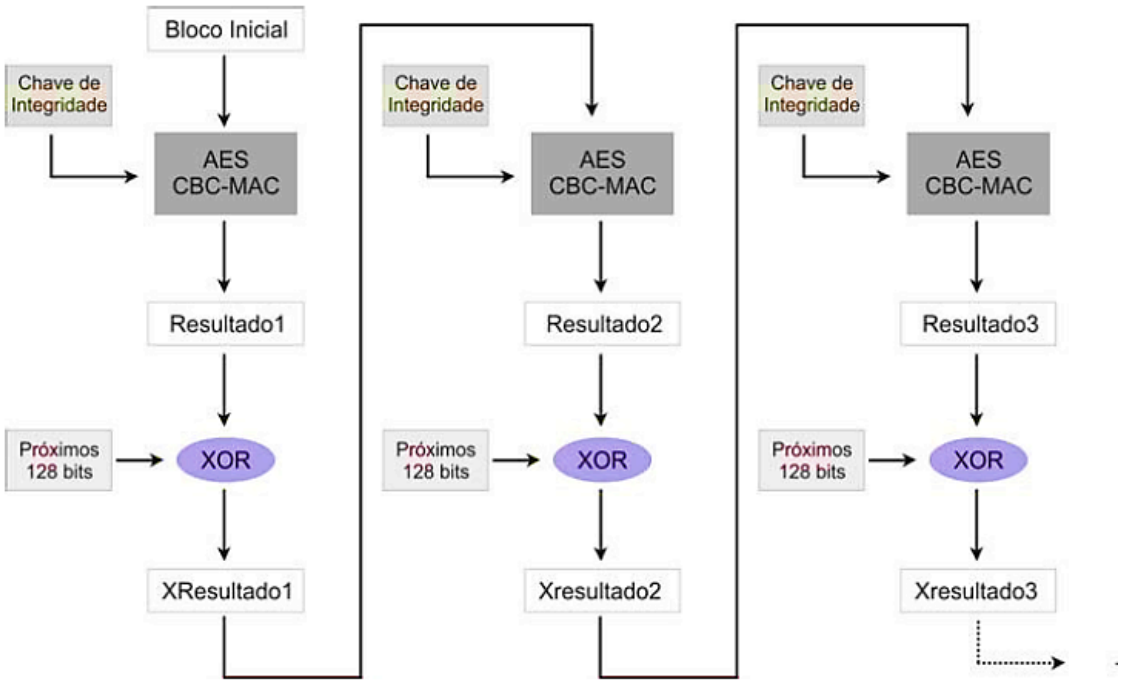

Figura 5 – CBC-MAC no WPA2.
Fonte: DUARTE, 2010.

O WPA2 opera em quatro fases (figura 6): descoberta, na qual o AP anuncia sua presença e oferece as formas de autenticação e criptografia e o cliente solicita o que ele

desejar; autenticação mútua e geração de chave mestra (MK – *Master Key*), em que ocorre a autenticação entre o cliente e o servidor de autenticação e é gerada uma chave mestra conhecida por ambos; geração de chave mestra pareada (PMK - *Pairwise Master Key*), na qual a MK é usada para gerar a PMK que será enviada pelo servidor ao AP; e geração de chave temporal (TK – *Temporal Key*), em que o cliente e AP geram chaves adicionais para a comunicação e o TK é utilizado para realizar a criptografia em nível de enlace (KUROSE; ROSS, 2010).

Figura 6 – Fases de operação do WPA2.
Fonte: Adaptado de KUROSE; ROSS, 2010.

2.2.3.1 Vantagens e desvantagens do WPA2

O WPA2 possui algumas vulnerabilidades. Ele é suscetível a ataques de camada física, como *jamming* e *flooding* e de DoS devido a falta de criptografia nos quadros de controle e gerenciamento (ZAMPERLINI; SANTOS, 2016). Foi descoberta uma falha na chave GTK (*Group Temporal Key*). Enquanto a PTK (*Pairwise Transient Key*), que é única para cada cliente, consegue detectar quando um endereço MAC é forjado, a GTK, que é usada para *broadcast* e é compartilhada entre os clientes autorizados na rede, não consegue detectar essa falsificação. Além disso, o CCMP requer um novo *hardware* para seu funcionamento e por isso é não compatível com dispositivos WEP e WPA.

Porém, mesmo com suas vulnerabilidades, o WPA2 é mais vantajoso que seus antecessores, pois solucionou os problemas encontrados nos mesmos e proporcionou maior

segurança e estabilidade devido à utilização de mecanismos mais avançados de criptografia e autenticação. O quadro 1 abaixo mostra uma comparação entre os três protocolos Wi-Fi.

Quadro 1- Comparação entre WEP, WPA e WPA2.
Fonte: autoral.

Características/ Protocolos	**WEP**	**WPA**	**WPA2**
Criptografia	RC4	RC4 + TKIP	CCMP + AES
IV	24 bits	64 bits	48 bits
Chaves	64 ou 128 bits	128 bits	128 bits
Integridade	CRC	MIC	CBC-MAC
Autenticação	*Open system* ou *shared key*	EAP	EAP
Desvantagens	- Reutilização de IV - Chave fixa conhecida por todos da rede	- Algoritmo de combinação de chaves é fraco - MIC suscetível a ataques	Falta de criptografia nos quadros de gerenciamento e controle

2.2.4 Ataques Wi-Fi

Assim como as demais tecnologias sem fio, o Wi-Fi possui suas vulnerabilidades e ataques podem ser realizados em cima das mesmas (CONTRACTI TI, 2016). Ataques WLAN *Scanners* ocorrem quando algum dispositivo opera na mesma frequência do AP e, portanto, pode captar os sinais transmitidos. O *NetStumbler* é um exemplo de uma ferramenta utilizada para tal ataque e permite a descoberta de LANs sem fio no Windows, além de localizar os pontos de acesso através do GPS (*Global Positioning System*).

Man-in-the-middle consiste na interceptação de dados na rede. Com ele, é possível que o atacante injete pacotes forjados na rede a fim de espionar e reproduzir pacotes enviados através da mesma. Um exemplo desse ataque é o seqüestro de uma conexão TCP. A ferramenta *Ettercap* pode ser utilizada para realizar este tipo de ataque.

O IP *Spoofing* utiliza uma técnica para alterar o endereço IP original por um falso, fazendo com o que o atacante se passe por um usuário da rede. O *Spoofing* pode ocorrer também com a alteração do endereço MAC, no qual o endereço físico da placa de rede é

alterado. Este tipo de ataque é chamado de *ARP-Spoofing* (ou *ARP-Poisoning*) e é um meio eficiente para execução do ataque de *Man-in-the-middle*, que permite que o atacante intercepte informações confidenciais posicionando-se no meio da comunicação entre duas ou mais máquinas. Consiste em uma falsa resposta ARP enviada à uma requisição ARP original, no qual o roteador pode ser convencido a enviar dados para o dispositivo atacante que redireciona os dados para o destinatário legítimo, portanto não há interrupção da comunicação (VIEIRA, 2008). O ARP *Spoofing* é utilizado para associar o endereço MAC destino a partir de seu endereço IP.

Os *Sniffers* são utilizados para capturar pacotes na rede. Eles exploram o tráfego dos pacotes das aplicações TCP/IP. Com isso, qualquer informação que não esteja criptografada pode ser obtida. Um exemplo bastante conhecido de *sniffer* é o *Wireshark.*

Em outubro de 2017, foi descoberta uma nova vulnerabilidade no WPA2 por pesquisadores de segurança (VANHOEF, 2017). O KRACKs (*Key Reinstallation Attacks*) permite que os atacantes leiam informações anteriormente assumidas como criptografadas com segurança. Com isso, é possível adquirir informações confidenciais, como número de cartão de crédito, senhas, mensagens de bate-papo, e-mails, dentre outros. O problema apontado não está em produtos ou implementações individuais, mas no próprio padrão Wi-Fi, o que torna qualquer rede suscetível ao ataque. Fabricantes como Android, Linux, Apple, Windows, OpenBSD, MediaTek, Linksys são afetados por alguma variante do ataque.

Para demonstrar o ataque, foi executado um ataque de reinstalação de chave em um *smartphone* Android. Na demonstração, o atacante é capaz de decriptografar todos os dados que a vítima transmite e por isso ataque não se limita apenas a recuperar credenciais de *login.* Além disso, dependendo do dispositivo utilizado, é possível decriptografar dados enviados para a vítima. O principal ataque é contra o 4-*way handshake* do protocolo WPA2. Este *handshake* é executado quando um cliente deseja se juntar a uma rede Wi-Fi protegida e é utilizado para confirmar se o cliente e o AP possuem as credenciais corretas. Além disso, ele negocia uma nova chave de criptografia para ser utilizada em todo o tráfego. A falha permite que essa chave seja resetada, e assim, decriptografar todo este tráfego. Atualmente, todas as redes Wi-Fi protegidas utilizam esse *handshake*, e mais uma vez isto implica que todas elas podem ser afetadas pelo ataque. Para as redes que utilizam WPA-TKIP ou GCMP (*Galois/Counter Mode Protocol*) o ataque pode ser piorado, pois, além de decriptografar, é possível forjar e injetar pacotes. Para evitar o ataque, os usuários devem atualizar os produtos afetados com as novas atualizações de segurança que surgirem.

2.3 *Bluetooth*

O *Bluetooth* é um padrão descrito pelo IEEE 802.15, caracterizado pela conectividade sem fio de baixa potência em redes WPAN, que são de pequeno alcance. É, principalmente, utilizado para transmissão de áudio, de dados e de *broadcast* entre dispositivos.

Os dispositivos que se comunicam por *Bluetooth* formam uma rede classificada como *piconet*, segundo Nakamura (2007). O funcionamento é baseado no relacionamento escravo-mestre, isto é, o escravo solicita um canal ao mestre e para que o canal seja reservado para transmissão de dados, é necessária que a senha seja pré-estabelecida entre os dispositivos para garantir a relação.

Conforme descrito por Mcdermott (2004), a frequência utilizada por dispositivos *Bluetooth* é ISM *(Industrial, Scientific, Medical)* centrada em 2,45GHz na camada física e baseia-se nos saltos de frequência com o objetivo de sanar possíveis interferências, porém a sequência de saltos utilizada é de conhecimento público, o que torna este mecanismo falho, portanto o mecanismo de chave compartilhada entre o escravo e o mestre compromete, também, a falha de segurança identificada.

Em uma *piconet,* o mestre transmite em *slots* de tempo pares enquanto os escravos transmitem apenas em *slots* de tempo ímpares. Em cada *slot* de tempo, devido ao mecanismo de saltos de frequência, um canal de frequência diferente é utilizado, isto é, após cada transmissão ou recebimento de pacotes, o canal é alterado (TANENBAUM, 2003).

2.3.1 A pilha de protocolos na tecnologia *Bluetooth*

O padrão *Bluetooth* especifica a pilha de protocolos do padrão 802.15 dividindo-o em três grupos (SIG, 2017): grupo de protocolos de transporte, grupo de protocolos de *middleware* e grupo de protocolo de aplicação, conforme a figura 7.

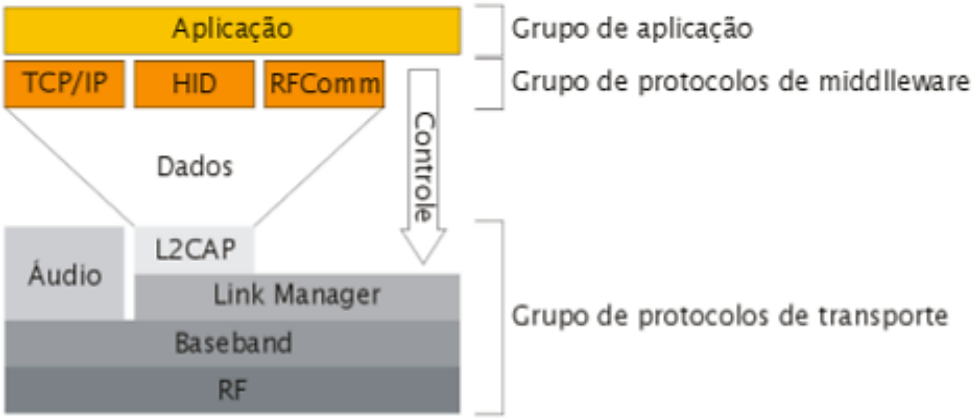

Figura 7- Pilha de protocolos da tecnologia *Bluetooth.*
Fonte: TANENBAUM, 2003.

Conforme a descrição dos grupos de protocolos de Tanenbaum (2003), a camada de Rádio Frequência (RF) corresponde essencialmente ao projeto dos *transceivers* e é composto pelo sintetizador de rádio frequência, recuperação de *clock* e detector de dados;

A camada *Baseband* mostrada na figura é essencial no processo de como os dispositivos *Bluetooth* identificam e se conectam a outros dispositivos, isto é, as funções de mestre e escravo são definidas, assim como os padrões de saltos de frequência utilizados. Nesta camada também são definidos os tipos de pacotes, procedimentos de processamento de pacotes, estratégias de detecção de erros, criptografia, transmissão e retransmissão de pacotes.

O *Link Manager* tem função de implementar o *Link Manager Protocol (LMP)*, que gerencia as propriedades do meio de transmissão entre os dispositivos que se comunicam. O protocolo é responsável pela alocação de taxa de transferência de dados e de áudio, pela autenticação através de métodos de desafio-resposta (*challenge-response*), pelos níveis de confiança entre dispositivos e, também, pela criptografia de dados e controle do gasto de energia;

A camada L2CAP (*Logical Link Controland Adaptation*) é a interface presente entre os protocolos de camadas superiores e os protocolos de transporte de camadas inferiores e possui a função de fragmentação e remontagem de pacotes.

O grupo de protocolos de transporte formado pelas camadas RF, *Baseband*, LMP e L2CAP, permite que dispositivos *Bluetooth* identifiquem outros dispositivos e, também, sejam capazes de gerenciar links físicos e lógicos para as camadas superiores. Estes protocolos correspondem às camadas físicas e de enlace do modelo OSI.

Os protocolos de *middleware* permitem que aplicações já existentes e novas aplicações operem sobre links *Bluetooth*. Protocolos de padrões industriais incluem *Point-to-Point Protocol* (PPP), *Internet Protocol* (IP), *Trasmission Control Protocol* (TCP), *Wireless*

Application Protocol (WAP) e protocolos desenvolvidos pelo próprio SIG também foram incluídos como o RFComm, que permite aplicações legadas operarem sobre os protocolos de transporte *Bluetooth* (SIQUEIRA, 2006).

Em relação ao grupo de aplicação, é formado pelas próprias aplicações que utilizam links *Bluetooth*, podendo incluir aplicações legadas ou aplicações orientadas a *Bluetooth*.

2.3.2 Topologias *Bluetooth*

A tecnologia permite três diferentes tipos de topologias segundo SIG (2017). A primeira é a Ponto a Ponto, em que a comunicação é de um-para-um, na qual um cliente se conecta ao único dispositivo. Este tipo de topologia é ideal para transferências de dados.

A segunda é a *Broadcast*, caracterizada pela comunicação um-para-muitos em que um nó é capaz de enviar dados para dois ou mais nós da rede. A topologia otimiza o compartilhamento de informações localizadas, como ponto de interesse e serviços de busca de itens e de caminho.

A terceira topologia utilizada em redes que utilizam o padrão 802.15 é conhecida como *Mesh* ou malha, em que a comunicação é de muitos-para-muitos, utilizada em automação de edifícios, rede de sensores, rastreamento de ativos e qualquer solução em que múltiplos dispositivos se comunicam.

Os tipos de topologias apresentados abrangem a tecnologia *Bluetooth* e possuem características utilizadas com frequência em ambientes IoT, já que, neste tipo de rede, não há necessidade de alto processamento e são arquiteturas que utilizam dispositivos de pequeno alcance.

2.3.3 Autenticação no *Bluetooth*

A autenticação é o modelo de segurança utilizado no padrão IEEE 802.15, segundo descrito por Nakamura e Zeus (2007), tanto para serviços, como para conexão entre dispositivos e este modelo é baseado em modos de segurança, em níveis de confiança em dispositivos e em níveis de segurança de serviços, como são especificados no padrão.

Em relação aos modos de segurança, existem três diferentes modos. O modo um não apresenta mecanismos de segurança. O modo dois, quando configurado, implementa segurança no nível de serviço, ou seja, o gerenciador faz o controle de acesso aos serviços e dispositivos, após a configuração do canal. Ao estabelecer o modo três, a segurança é

implementada a nível de enlace, antes da configuração do canal, o processo de pareamento é utilizado.

Ainda no modelo de segurança do *Bluetooth*, os níveis de confiança em dispositivos são baseados no relacionamento permanente ou não entre si e são classificados em não confiáveis e confiáveis. Dessa forma, o dispositivo não possui permissão para acessar determinados serviços, portanto, a relação não é permanente ou fixa ou os dispositivos que possuem relacionamento fixo e acesso a todos os serviços disponíveis, respectivamente.

Por fim, os níveis de confiança de serviços são diferenciados pela necessidade ou não de autenticação e autorização. No nível um de serviço há necessidade de autenticação e autorização, por esse motivo somente dispositivos confiáveis se conectam automaticamente. Já no nível dois de serviço de segurança somente a autenticação é requerida e no nível três o acesso é permitido a todos os dispositivos, ou seja, não há necessidade de autorização e de autenticação.

Existem diferentes processos que podem ser utilizados para autenticação de dispositivos com tecnologia *Bluetooth*, são eles:

- Baseada em chave secreta compartilhada:

Parte-se do princípio que os usuários dos dois dispositivos definiram a chave de autenticação previamente. Este tipo de autenticação é baseado no método conhecido como desafio-resposta (*challenge-response*) que consiste no envio de um número aleatório, o desafio, de um dispositivo ao outro com o objetivo de obter uma resposta do outro dispositivo envolvido.

A figura 8 abaixo demonstra o funcionamento que consiste no envio da identidade do dispositivo A. O dispositivo B, para garantir a autenticidade da identidade recebida, envia o desafio, um número aleatório e extenso para que, em seguida, o dispositivo A criptografe o desafio recebido com a chave previamente definida. O mesmo processo ocorre do dispositivo B para o A, isto é, para garantir a identidade de B o dispositivo A envia outro desafio para B, que ao receber a mensagem, a criptografa com a chave compartilhada.

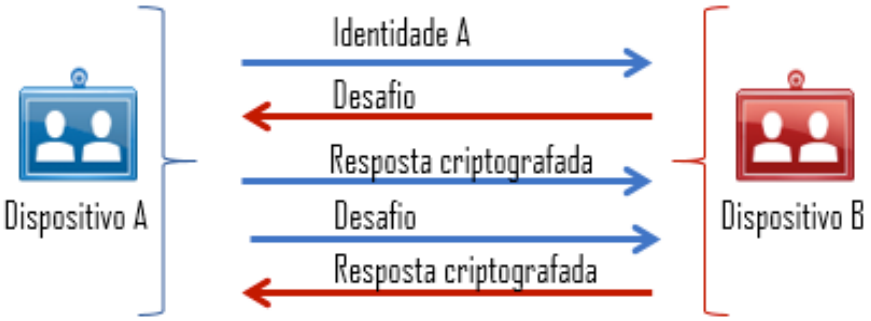

Figura 8 – Autenticação por chave compartilhada.
Fonte: Adaptado de TANEMBAUM, 2003.

- Autenticação com centro de distribuição de chaves:

Este tipo de autenticação utiliza um centro de distribuição de chaves que consiste no armazenamento de chave que o próprio usuário compartilha e, por esse motivo, o gerenciamento sessão e de autenticação sempre passa pelo centro de distribuição.

Utilizando o protocolo Otway-Rees como exemplo, a figura 9 abaixo demonstra de forma simplificada o processo de autenticação neste caso. O dispositivo A gera um par de números aleatórios, o desafio, para o dispositivo B que, ao receber a mensagem, gera duas novas mensagens e criptografadas: a primeira com uma parte da mensagem recebida e a segunda criada pelo próprio dispositivo B, análoga à mensagem recebida.

Em seguida, o centro de distribuição de chave verifica se o número gerado é igual e, em caso afirmativo, gera uma chave de sessão criptografada duas vezes, uma para o dispositivo A e outra para o dispositivo B, cada uma com o desafio do receptor para garantir a autenticidade da identidade de A e B. No momento em que os dois dispositivos possuem a chave de sessão, a comunicação pode ser estabelecida e na primeira troca de dados, cada um tem acesso às mensagens que haviam sido criptografadas no início do processo.

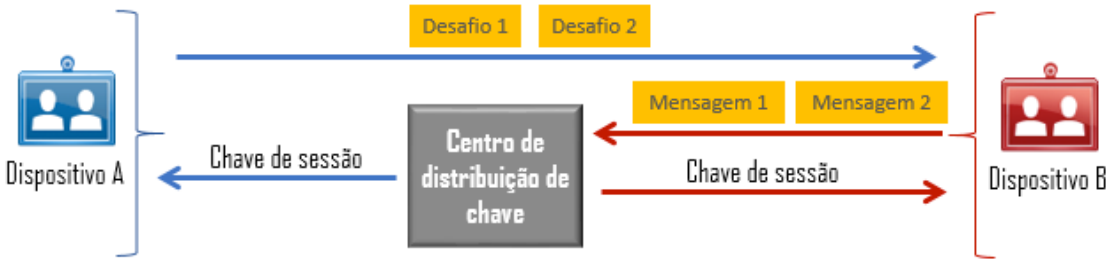

Figura 9 – Autenticação com centro de distribuição de chaves.
Fonte: Adaptado de TANENBAUM, 2003.

2.3.4 A comunicação

Conforme a especificação do padrão IEEE 802.15, a comunicação é baseada no esquema de divisão de tempo (*Time Division Duplexing* - TDD) e divisão de tempo com múltiplos acessos (*Time Division Multiple Access* - TDMA), isto é, transmissão de dados é realizada por meio de *slots* de tempos.

Na camada de *Baseband*, um pacote é constituído por um código de acesso seguido por um cabeçalho e pelos dados. O código de acesso contém o endereço da *piconet* e possui 72 bits de comprimento. O cabeçalho contém o endereço de 18 bits de um dispositivo escravo ativo na rede e o campo de dados é a parte do pacote onde trafegam os dados da aplicação e pode conter até 2745 bits.

Na comunicação baseada na tecnologia *Bluetooth* os dispositivos são identificados em um dos seguintes estados: espera, solicitação, página, conectado, transmissão, bloqueado, escuta e estacionado, que podem ser observados na figura 10 (SIQUEIRA, 2006).

Um dispositivo está no estado de espera quando está ligado, porém não está associado a uma rede *piconet*. Então, entra no estado de solicitação a partir do momento em que envia requisições de busca aos dispositivos com o objetivo de comunicação. Um dispositivo mestre está no estado de página quando envia constantemente mensagens à procura de dispositivos disponíveis para associar-se à rede.

No momento em que há comunicação consistente entre mestre e um novo dispositivo, este dispositivo assume o papel de escravo e passa ao estado de conectado para, assim, receber um endereço que o identifica. Enquanto conectado, um escravo pode transmitir dados sob condição de permissão do mestre e assume o estado de transmissão durante o envio dos dados. Ao finalizar a transmissão de dados, o dispositivo retorna ao estado de conectado.

O estado de escuta é caracterizado pelo baixo consumo de energia onde o escravo não realiza atividade por um número pré-definido de *slots.* Dessa forma, o dispositivo, em seu *slot* de tempo designado, realiza a transmissão de dados e, ao finalizá-lo, retorna para o estado de escuta até que o próximo *slot* reservado.

O estado de bloqueado também verifica o baixo consumo de energia, isto é, o escravo não está ativo por um período pré-determinado de tempo e, diferente do estado de escuta, não há transferência de dados dentro do estado bloqueado. Quando um dispositivo escravo não tem dados a serem enviados ou recebidos, o dispositivo mestre pode instruí-lo a entrar no estado de estacionado e o dispositivo perde seu endereço atual da rede, que será dado a outro escravo.

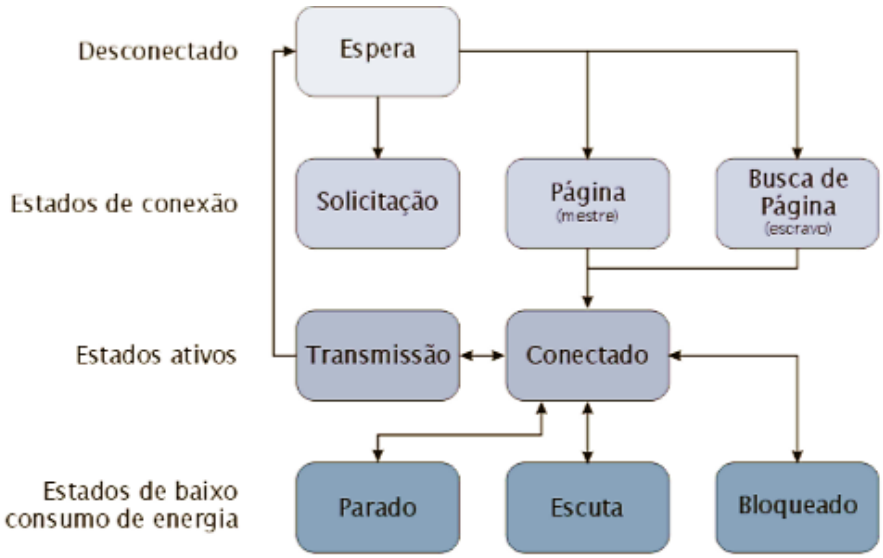

Figura 10 – Esquema de estados de dispositivos *Bluetooth*.
Fonte: SIQUEIRA, 2006.

2.3.4 Ataques *Bluetooth*

Ataques *Bluetooth* podem ocorrer com usuários que utilizam a tecnologia disponível e visível. Interceptação de dados, escaneamento da rede WPAN, acesso às configurações e informações dos dispositivos são tipos de ataques conhecidos ao padrão de comunicação. Existem diversos tipos de ferramentas de ataques conhecidos à tecnologia *Bluetooth*, geralmente motivados por benefícios financeiros e acesso a informações pessoais de usuários. Alguns exemplos são: *Bluetracking, Bluesnarfing, Bluebugging, Bluejacking* (LINS, 2010).

No ataque *Bluetracking,* o atacante obtém o endereço MAC de um dispositivo com o objetivo de rastrear a localização do usuário, bem como seus movimentos. Com o *Bluesnarfing,* geralmente o ataque é realizado sem que o usuário perceba. Este ataque é capaz de fazer cópia não autorizada de diversas informações contidas no dispositivo. A partir do momento em que há emparelhamento o atacante tem a capacidade para ter acesso a diversos dados do dispositivo mestre. Envolve o roubo de dados de um dispositivo, o que pode incluir informações de listas de contatos, calendários, e-mails ou mensagens de texto (FINJAN MOBILE, 2017). Nos dispositivos que utilizam as versões mais novas do *Bluetooth* esse ataque não é mais eficaz.

No ataque *Bluebugging,* o atacante envia comandos executáveis aos dispositivos *Bluetooth*. Este ataque permite acesso a comandos de configuração, fazendo se passar por usuário proprietário do dispositivo. O último ataque citado por Lins (2010), o *Bluejacking*,

consiste na transmissão de mensagens de texto não autorizadas durante o processo de pareamento de dispositivos localizados até dez metros da origem do ataque.

Outros tipos de ferramentas de ataques são descritos pela BlackArch (2013), que podem ser executados em dispositivos que utilizam o padrão IEEE 802.15. O *Btscanner* e *Bluescan* consistem no escaneamento de informações importantes restritas. O *Spooftooph* é baseado na falsificação ou cópia do nome, endereço MAC e classe do dispositivo com o objetivo de forjar uma identificação para garantir comunicação e acesso ao alvo.

Um importante e recente ataque foi identificado pela Armis, uma empresa cuja missão é de eliminar pontos cegos de segurança do IoT, permitindo que as empresas utilizem dispositivos IoT não gerenciados com segurança e de forma segura (ARMIS, 2017).A ArmisLabs revelou em setembro de 2017 um novo vetor de ataque que expõe os principais riscos de sistemas operacionais móveis, de *desktop* e IoT, incluindo Android, iOS, Windows e Linux e os dispositivos que os utilizam. O novo vetor é apelidado de *BlueBorne*, a medida que se espalha pela rede sem fio, ele ataca dispositivos via *Bluetooth*.

BlueBorne é um vetor de ataque pelo qual os hackers podem aproveitar as conexões *Bluetooth* para penetrar e controlar completamente os dispositivos direcionados. O *BlueBorne* afeta computadores comuns, telefones celulares e o domínio em expansão dos dispositivos IoT. O ataque não exige que o dispositivo segmentado seja emparelhado com o dispositivo do invasor, ou mesmo ser configurado no modo detectável. O ataque permite que os invasores assumam o controle de dispositivos, acessem dados corporativos e redes, penetrem em redes seguras de acesso sem fio e espalhem *malware* lateralmente para dispositivos adjacentes (ARMIS, 2017).

De acordo com Keljka (2017), o vetor de ataque *BlueBorne* pode potencialmente afetar todos os dispositivos com recursos *Bluetooth*, estimados em mais de 8,2 bilhões de dispositivos hoje. Entre os dispositivos vulneráveis estão os *smartphones* Google Pixel, os celulares e tablets Samsung Galaxy, todos os computadores Windows desde o Windows Vista, Samsung *smartwatches,* televisores e refrigeradores, todos os dispositivos iPhone, iPad e iPod touch com iOS 9.3.5 e inferior e dispositivos AppleTV com versão 7.2.2 e inferior.

O vetor de ataque do *BlueBorne* possui vários estágios. Primeiro, o atacante localiza conexões *Bluetooth* ativas ao redor dele. Os dispositivos podem ser identificados mesmo que não estejam configurados para o modo visível. Em seguida, o invasor obtém o endereço físico do dispositivo, que é um identificador exclusivo desse dispositivo específico. Ao testar o dispositivo, o atacante pode determinar qual o sistema operacional que sua vítima está usando

e ajustar o ataque de acordo com esta informação. O invasor irá, então, explorar uma vulnerabilidade na implementação do protocolo *Bluetooth* na plataforma relevante e obter o acesso que ele precisa para atuar em seu objetivo malicioso. Nesta fase, o atacante pode escolher criar um ataque do tipo *Man-in-The-Middle* e controlar a comunicação do dispositivo ou ter controle total sobre o dispositivo (ARMIS, 2017).

2.4 *ZigBee*

Fundada em 2002, a *ZigBee Alliance*, em conjunto ao IEEE, desenvolveu o padrão *ZigBee*. É uma associação de várias empresas, universidades e agências governamentais de todo o mundo que juntas, trabalham em conjunto para proporcionar e desenvolver tecnologias com o objetivo de criar um padrão de baixo consumo de energia, baixo custo, segurança, confiabilidade, interoperabilidade e com funcionamento em rede sem fios baseado em uma norma aberta global (ZIGBEE ALLIANCE, 2013).

A tecnologia *ZigBee*, descrita no padrão IEEE 802.15.4, é especificada para comunicação de equipamentos de comunicação de dados em redes sem fio de área pessoal (WPAN), que utilizam baixa taxa de dados, baixa potência e transmissões de rádio frequência de curto alcance e, também, de baixa complexidade (BLACKHAT, 2015). As características apresentadas são comuns em ambientes IoT, pois os domínios de aplicação da tecnologia são, principalmente, em automação de residências e prediais, dispositivos da área da saúde, serviços de telecomunicações, entre outros.

Conforme a ZigBee Alliance (2007), a camada de rede tem capacidade de suportar as topologias estrela, árvore e malha. Na topologia em estrela, a rede é controlada por um único dispositivo responsável por iniciar a topologia da rede, denominado coordenador *ZigBee* e os dispositivos finais comunicam-se diretamente com o coordenador. Em redes com topologias em malha a comunicação é do tipo *peer-to-peer* e na topologia em árvore, os roteadores movem dados e mensagens de controle através da rede utilizando de roteamento hierárquico. Em ambas as topologias, também há o papel de coordenador *ZigBee* possibilitando, assim, definir os parâmetros principais que serão utilizados na rede.

2.4.1 A arquitetura

A arquitetura *ZigBee*, figura 11, é constituída de um conjunto de blocos, chamados de camadas. Cada camada executa um conjunto específico de serviços para a camada superior. Uma entidade de dados fornece um serviço de transmissão de dados e uma entidade de

gerenciamento fornece todos os outros serviços. Cada entidade de serviço fornece uma interface para a camada superior por meio de um ponto de acesso de serviço e cada ponto suporta um número de primitivas de serviço para alcançar a funcionalidade necessária (ZIGBEE ALLIANCE, 2007).

O padrão 802.15.4 especifica as camadas física e de acesso ao meio, *Media Access Control* (MAC), para redes *wireless* que operam com baixa taxa de transmissão de dados, do inglês *Lower Rate Wireless Personal Area Network* (LR-WPAN).

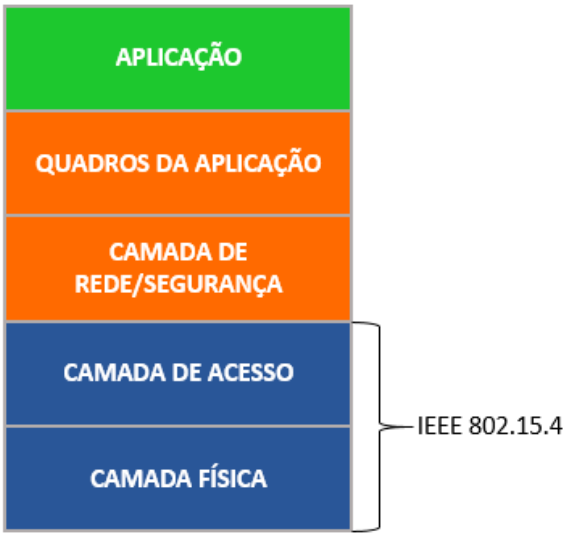

Figura 11 – Composição de camadas do IEEE 802.15.4.
Fonte: Adaptado de MELO, 2017.

De acordo com Melo (2017), a camada física tem função de transmitir e receber dados chamados de PPDUs (*PHY Protocol Data Unit*) e, logo depois, enviá-los no formato adequado à camada superior. É responsável também, segundo especificado pelo IEEE (2011), por detectar a energia do canal atual, indicar a qualidade do *link* para os pacotes recebidos, avaliar a disponibilidade do canal e, por fim, selecionar a frequência do canal.

Pode-se operar em três diferentes bandas de frequências não licenciadas, são elas: 868 MHz utilizada na região da Europa, 915 MHz para os Estados Unidos da Américas e 2,4GHz usualmente utilizada no Brasil, todas definidas sobre as bandas ISM. As três faixas de frequência utilizam o método de transmissão DSSS (*Direct Sequence Spread Spectrum*) utilizado devido à necessidade da alta interação e baixo custo (FILHO, 2016).

A camada de acesso tem como objetivo controlar a operação de acesso à camada física para recepção e transmissão de dados e, também, servir de interface à camada física com as camadas superiores do protocolo LR-WPAN (MELO, 2017). Conforme descrito por Vasques (2010), existem dois modos de operação de acesso ao meio: *Beaconing* e *Non-Beaconing*.

O modo *Beaconing* consiste na transmissão periódica de *beacon frames*, sinais que têm por objetivo sinalizar e confirmar sua presença na rede, são sinais de controle que delimitam quadros utilizados pelo coordenador para sincronização de dispositivos. Dada a sincronia, os nós da rede, com exceção do coordenador, podem permanecer inativos entre os *beacon frames* e, assim, economizar energia.

O controle de acesso ao meio é realizado através de mecanismos de prevenção e colisão, que garante a maior confiabilidade de que os dados serão recebidos sem que haja choque de pacotes. O protocolo utilizado de acordo com Nenoki (2013) é o CSMA-CA (*Carrier Sense Multiple Access with Collision Avoidance*). Seu funcionamento é baseado na escuta obrigatória do canal antes da transmissão de dados. Dessa forma, a prevenção de colisão é realizada, pois o dispositivo que deseja enviá-los verifica se o canal está inativo por pelo menos a duração aleatória, chamado DIFS (*Function Inter Frame Space*), um tempo aleatório, no qual uma estação pode acessar o meio imediatamente.

Caso o meio esteja ocupado, as estações esperam a duração de DIFS e, logo em seguida, iniciam na fase de contenção. Dessa forma, cada estação determina um *backoff time* aleatório, dentro de uma janela de contenção, para que seja realizada outra tentativa de acesso ao meio. Na situação em que passou esse intervalo de tempo e o meio ainda está ocupado, a estação perde o seu ciclo e deverá esperar até a próxima oportunidade de ciclo, ou seja, o dispositivo está condicionado ao estado do meio e há necessidade que esteja inativo novamente por um período de DIFS. Caso contrário, se passado o intervalo de tempo aleatório e o meio estiver livre, a estação pode acessá-lo. O protocolo também possui outro atraso apresentado entre a recepção de um quadro e a transmissão do quadro de reconhecimento é chamado *Short Inter Frame Spacing* (SIFS) (BONAVENTURE, 2010).

No modo *Non-Beaconing*, conforme descrito por Vasques (2010), não é utilizado o método de transmissão de *Beacons* como no modo anterior e o método de acesso ao meio é realizado baseado no *CSMA unslotted ALOHA*, um protocolo que não se utiliza de *slots* determinados de tempo para transmissão e, dessa forma, o dispositivo escuta o meio e, caso esteja ocupado, espera um tempo aleatório e exponencialmente decrescente para realizar outra tentativa de acesso ao canal.

2.4.3 Segurança *ZigBee*

Um protocolo de segurança de camada de *link* fornece quatro serviços de segurança: controle de acesso, integridade da mensagem, confidencialidade da mensagem e proteção de *replay* (SASTRY; WAGNER, 2004).

O controle de acesso significa que o protocolo da camada de segurança deve controlar a acessibilidade da rede, impedindo que dispositivos não autorizados sejam integrados a ela. Os nós legítimos devem ter a capacidade de detectar mensagens de nós não autorizados e rejeitá-los.

Além disso, uma rede segura deve prover proteção de integridade das mensagens, isto é, se um atacante consegue alterar uma mensagem de um remetente autorizado enquanto a mensagem está em trânsito, o receptor deve ser capaz de detectar a modificação. E, também, um código de autenticação de mensagens (em inglês, *Message Authentication Code*) que fornece a cada pacote autenticação e integridade de mensagens.

Em relação à confidencialidade, "a definição clássica de confidencialidade é a garantia do resguardo das informações dadas pessoalmente em confiança e a proteção contra a sua revelação não autorizada" (KENNEDY INSTITUTE OF ETHICS. BIOETHICS THESAURUS, 1995). Um esquema de criptografia deve não só impedir a recuperação da mensagem como também evitar que atacantes aprendam informações parciais sobre as mensagens que foram criptografadas, de acordo com Bellare (1997). Portanto, a confidencialidade de determinada mensagem significa que a mesma não será acessível e divulgada à pessoas não autorizadas, dadas as condições, a criptografia é uma forma de garantir confidencialidade de dados.

Para entender a proteção *Replay,* deve-se conceituar o ataque *replay*, que consiste na interceptação de pacotes transmitidos em uma comunicação entre duas entidades e que posteriormente, são reutilizados e retransmitidos na tentativa de forjar uma nova comunicação. Portanto, durante a comunicação, quando um dispositivo é autorizado a transmitir uma mensagem, haverá um código de autenticação de mensagem (*Message Authentication Code)* válido. Dessa forma, o receptor autorizará novamente a comunicação. Considerando esta situação, a proteção *replay* é capaz de evitar esse tipo de ataque, pois o dispositivo remetente atribui um número de sequência de ordem crescente a cada pacote, esse número é checado pelo destinatário que, ao receber um número menor da sequência, descarta o dado recebido (SASTRY; WAGNER, 2004).

A figura 11 acima mostra que no padrão IEEE 802.15.4 a segurança é implementada entre a camada de acesso e a camada de aplicação. Segundo Vasquez (2010), a especificação define quatro tipos de quadros para a camada de controle de acesso à mídia: *beacons*, *frames* de dados, *acknowledgement* (ACK) e frames de controle. A especificação não suporta segurança para pacotes ACK; outros tipos de pacotes podem opcionalmente suportar proteção de integridade e proteção de confidencialidade para o campo de dados do *frame*.

Sastry e Wagner (2004) discorrem sobre a definição do padrão em oito métodos de segurança diferentes disponíveis na aplicação. As classificações se diferem pelas propriedades oferecidas, são elas: sem segurança, somente criptografia (AES-CTR), somente autenticação (AES-CBC-MAC) e criptografia em conjunto com autenticação (AES-CCM), com variações no tamanho de código de autenticação de mensagens, como mostrado no quadro 2 a seguir:

Quadro 2 - Métodos de segurança *ZigBee*.
Fonte: Traduzido de SASTRY e WAGNER, 2004.

Nome	**Descrição**
Nula	Sem segurança
AES-CTR	Somente criptografia, Modo *Counter*
AES-CBC-MAC-128	Código de autenticação de mensagens de 128 bits
AES-CBC-MAC-64	Código de autenticação de mensagens de 64 bits
AES-CBC-MAC-32	Código de autenticação de mensagens de 32 bits
AES-CCM-128	Código de autenticação de mensagens de 128 bits
AES-CCM-64	Código de autenticação de mensagens de 64 bits
AES-CCM-32	Código de autenticação de mensagens de 32 bits

O tipo nulo de segurança, como o próprio nome demonstra, não garante qualquer tipo de criptografia, isto é, os dados não possuem proteção de confidencialidade ou de integridade. Ao utilizar o AES-CTR a confidencialidade dos dados é garantida por meio da codificação AES em conjunto ao modo contador, isto é, o remetente adiciona o contador de quadros, o contador de chaves e a carga útil criptografada no campo da carga útil de dados do pacote.

O AES-CBC-MAC garante a integridade dos dados ao utilizar o CBC-MAC. O tamanho do código de autenticação de mensagem varia conforme as informações do quadro e é processado utilizando a chave simétrica. Na transmissão de dados, o código de autenticação de mensagens é adicionado ao pacote e o outro dispositivo, ao recebê-lo, faz a comparação com o código incluído no pacote.

A segurança AES-CCM utiliza o CCM para encriptação e autenticação. Seu funcionamento é baseado na combinação de dois tipos de segurança já mencionados, o CBC-MAC e AES-CTR. Utiliza-se o CBC-MAC com o objetivo de garantir a integridade de dados para depois, com o AES-CTR, encriptá-los com o código de autenticação de mensagem. O tamanho do código pode variar de acordo com as informações do quadro.

Portanto, o nível de segurança oferecido pela tecnologia, de acordo com Sastry e Wagner (2004), varia entre a implementação de nenhum mecanismo de segurança, ou seja, a segurança chamada de nula, até a utilização de criptografia com necessidade de autenticação no dispositivo *ZigBee*.

2.4.4 Ataques *ZigBee*

Os ataques aos dispositivos que utilizam a tecnologia *ZigBee* como forma de comunicação inviabilizam o funcionamento normal da troca de dados. Ataques que ocorrem na camada MAC são classificados como ataques que seguem o protocolo MAC (*Attacks that follow the MAC protocol*) e ataques que modificam o protocolo MAC (*Attacks that use a modified MAC protocol*) (VOJISLAV,2006).

Em ataques que seguem o protocolo MAC existem diversos casos em que próprio atacante é associado à rede PAN e age como um usuário legítimo. O ataque de negação de serviço é comum e afeta a disponibilidade da rede, isto é, o atacante é capaz de degradar o desempenho da rede e reduzir consideravelmente a taxa de transferência de dados. Este ataque consiste em afetar a rede simplesmente transmitindo um grande número de pacotes ou pacotes com tamanho maior que permitido pelo padrão IEEE 802.15.4.

A segunda classificação descrita por Vojislav (2006) pode ocorrer modificando ou desconsiderando determinados recursos do protocolo MAC. A modificação é realizada no protocolo de acesso ao meio, CSMA, e consiste em não incrementar o expoente *backoff* após uma tentativa de transmissão malsucedida. O gerador de números aleatórios pode ser modificado para dar preferência a um menor tempo *backoff*, isso permite que o nó malicioso capture o canal com frequência maior e com vantagem injusta em relação aos outros nós da

rede, pois o nó atacante poderá transmitir um maior número de dados. Dessa forma, haverá mensagens enviadas com sucesso e colisões em muitos casos, além disso, colisões entre os pacotes enviados pelo atacante com os ACKs desperdiçam tanto a bateria dos dispositivos como também a largura de banda de toda a rede.

Ataques conhecidos como *Jamming* e *Exhaustion* são tipos de ataque de negação de serviço que ocorrem neste padrão de comunicação. O *Jamming* é eficaz especialmente em redes de frequência única, pode ser facilmente executado através do envio contínuo de sinais de rádio com uma potência de transmissão relativamente alta. As redes *ZigBee* são ainda mais vulneráveis aos ataques de interferência pelo fato de possuir potência de transmissão extremamente baixa. O ataque *Exhaustion*, comum de exaustão, é executado a partir da exploração dos processos de iniciação ou conexão, como procedimentos de associação, que exigem que ambos os nós envolvidos armazenem alguns valores de estado em sua memória. Um dispositivo é capaz de tentar associar-se a todos os coordenadores ao seu alcance, apesar da especificação descrita do protocolo, no qual exige que cada dispositivo seja associado apenas com um coordenador (VIDAL, 2011).

Na especificação IEEE 802.15.4, existe o mecanismo de proteção de repetição *(replay-protection)* que impede a recepção de mensagens repetidas. Isto é conseguido porque um receptor verifica o contador recente e rejeita *frames* com o valor do contador igual ou menor comparado ao recebido anteriormente.

Este mecanismo pode ser utilizado de forma maliciosa resultando no ataque de repetição de proteção (*Replay-protection attack)*. O atacante é capaz de enviar diversos quadros contendo diferentes e grandes contadores de quadros para um receptor, que executa proteção de repetição e eleva o contador de repetição como o contador de quadro maior no receptor. Dessa forma quando uma estação normal envia um quadro com um tamanho razoável do contador de quadros que é menor mantido no receptor, o quadro será descartado para a repetição e, consequentemente, o serviço será negado (VIDAL, 2011).

Ademais, outro ataque conhecido é de sincronização que tem a capacidade de influenciar todos os nós da rede simultaneamente. Consiste na representação e utilização da mesma identificação do coordenador PAN (PAN ID) por um nó malicioso. O atacante altera os parâmetros próprios de sincronização em relação ao nó coordenador legítimo, assim, os nós legítimos processam os *beacons* de ambos os nós e sincronizam seus intervalos de tempo de acordo com os parâmetros manipulados pelo nó malicioso (JUNG, 2011).

Na ferramenta de ataque Kali Linux, o pacote *killerbee* identifica formas de explorar o padrão IEEE 802.15.4.

> "O *KillerBee* é um *framework* e conjunto de ferramentas baseado em *Python* para explorar a segurança das redes *ZigBee* e IEEE 802.15.4. Usando as ferramentas do *KillerBee* e uma interface de rádio IEEE 802.15.4 compatível, você pode escapar às redes *ZigBee*, ao tráfego de repetição, aos criptos de ataque. Usando o *framework KillerBee*, você pode criar suas próprias ferramentas, implementar *ZigBee fuzzing*, imitar e atacar dispositivos finais, roteadores e coordenadores e muito mais" (KALITOOLS, 2014).

O Kalitools (2014) explica, ainda, os diversos comandos para execução do *KillerBee*, entre eles estão *zbreplay, zbfind, zbdump, zbstumbler*. O *zbreplay* é capaz de replicar o tráfego de um arquivo de extensão *pcap.* O *zbfind* possui finalidade de rastrear a localização de um transmissor *ZigBee*. *O zbdump* é similar ao *tcpdump* utilizado para captura de pacotes trafegados na rede. Por último, o *zbtumbler* realiza a transmissão de quadros *beacons* para o endereço de *broadcast.*

3. METODOLOGIA

Este capítulo aborda a metodologia utilizada na realização do trabalho e detalha as características principais como delimitação do tema, tipo de investigação e o limite do estudo. São expostas, também, as etapas de realização do trabalho, bem como os dispositivos e ferramentas necessários, os cenários de simulação dos ataques.

3.1 Delimitação do tema

Considerando o universo da IoT, os dispositivos presentes no mercado utilizam, normalmente, redes WLAN e WPAN com tecnologias de comunicação Wi-Fi, *Bluetooth* e *ZigBee*, descritas pelo IEEE nos padrões 802.11, 802.15 e 802.15.4, respectivamente. Visto que as arquiteturas utilizadas são padronizadas e acessíveis ao público, possuem vulnerabilidades que podem ser exploradas.

Este trabalho demonstra formas de exploração de dados que podem ser realizadas neste universo, no qual são considerados que os dispositivos se comunicam por meio dos padrões citados.

3.2 Tipo de investigação

Para classificação da pesquisa, toma-se como base a taxionomia apresentada por Vergara (2014), que a qualifica em relação a dois aspectos: quanto aos fins e quanto aos meios.

Quanto aos fins, a pesquisa será metodológica e aplicada. Metodológica porque o estudo se refere à manipulação dos cenários domésticos e o laboratório UIoT da Universidade através da execução de procedimentos e simulações para alcançar o objetivo do trabalho. Aplicada porque é motivada pela necessidade de solucionar problemas concretos, no caso, as vulnerabilidades dos protocolos utilizados em ambientes IoT.

Quanto aos meios, a pesquisa será de laboratório e bibliográfica. De laboratório, pois é realizada em ambiente circunscrito, no laboratório localizado na Universidade de Brasília, UIoT. Bibliográfica porque para a fundamentação teórico-metodológica do trabalho será realizada a investigação sobre os seguintes assuntos: ataques que exploram as vulnerabilidades associadas aos protocolos utilizados em ambientes IoT e as possíveis defesas.

3.3 Coleta e tratamento de dados

A coleta de dados será realizada por pesquisa bibliográfica dado que livros, teses e dissertações são as principais fontes de informações. Estudos sobre os padrões de comunicação utilizados em dispositivos IoT e suas vulnerabilidades, bem como os ataques realizados e meios possíveis de defesa serão tratados no projeto.

3.4 Limites do estudo

Um aspecto importante é a limitação bibliográfica, devido à dificuldade de acesso às fontes que abordam os ataques possíveis às tecnologias de comunicação estudadas, dado que os procedimentos para execução dos mesmos não são amplamente divulgados. As principais fontes bibliográficas são teses, dissertações e artigos, o que dificulta, por vezes, o acesso às informações pesquisadas.

Os ataques realizados são simulados em cenários controlados que possuem determinados tipos dispositivos, portanto, o ambiente IoT restringe-se aos equipamentos disponibilizados pelo LATITUDE e de aquisição própria. Vale ressaltar que para a execução dos ataques há necessidade de autorização e que existem ataques que podem comprometer o correto funcionamento da rede e, dessa forma, suscetível à não autorização por parte dos responsáveis pelo cenário.

3.5 Etapas do projeto

Como forma de organização do trabalho, pode ser resumido em seis etapas principais, mostradas no fluxograma:

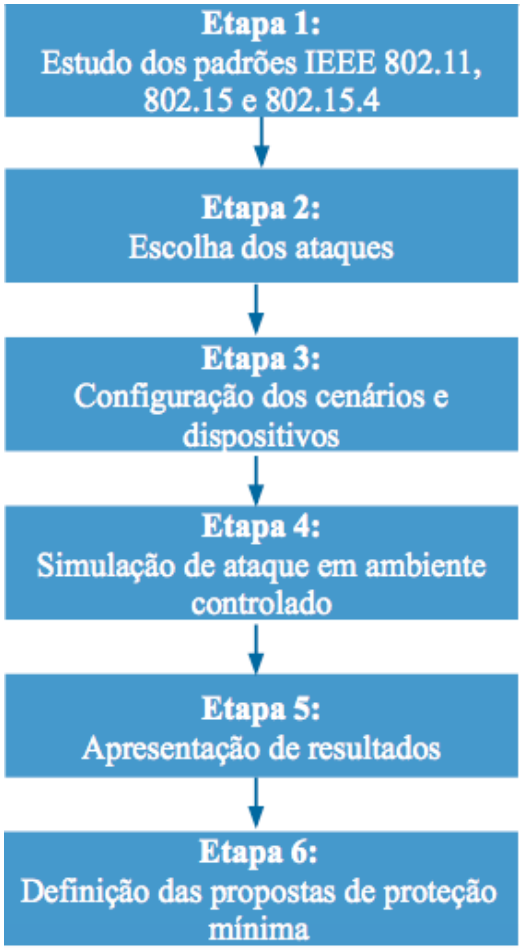

Figura 12 – Fluxograma das etapas do trabalho.
Fonte: Autoral.

- Etapa 1: Estudo dos padrões IEEE 802.11, 802.15 e 802.15.4.

  Esta etapa consiste no estudo das definições descritas pelo Instituto dos Engenheiros Eletrônicos e Eletricistas como forma de identificar e conhecer as arquiteturas, versões, vulnerabilidades e mecanismos de segurança implementados em cada tecnologia.

- Etapa 2: Escolha dos ataques.

  Esta etapa compreende a escolha e pesquisa, baseada na etapa 1, das possibilidades de ataques conhecidos para cada padrão, Wi-Fi, *Bluetooth* para que, posteriormente, sejam executadas em ambientes controlados. Os ataques escolhidos são adaptados às características de cada cenário disponível, dessa forma, limitam-se às regras e autorizações para execução.

- Etapa 3: Configuração dos cenários e dispositivos.

  Esta etapa baseia-se na preparação dos requisitos necessários para a execução dos ataques escolhidos na etapa 2. A configuração das redes WLAN e WPAN foi

essencial. No ataque Wi-Fi, a configuração do *Access Point* disponibilizado, aquisição do adaptador *wireless*, configuração da máquina virtual Kali Linux e configuração do dispositivo alvo. No ataque *Bluetooth,* aquisição do adaptador *Bluetooth* utilizado na máquina virtual atacante, configuração do Kali Linux, aquisição de dispositivos utilizados como alvo do ataque.

- Etapa 4: Simulação de ataque em ambiente controlado.
  Esta etapa consiste na execução dos ataques escolhidos aos padrões abordados e escolhidos na etapa 2, bem como a sua documentação e análise.
- Etapa 5: Apresentação dos resultados.
  Esta etapa consiste na obtenção e apresentação dos resultados a partir da execução dos ataques nos cenários Wi-Fi e Bluetooth. Nesta etapa são elaboradas as análises para caracterização dos padrões abordados.
- Etapa 6: Definição das propostas de recomendações mínimas.
  Nesta etapa são elaboradas as propostas e recomendações mínimas para proteção contra ataques. Estas propostas são apresentadas como forma de minimizar a chance de sucesso dos ataques expostos no trabalho.

3.6 Ferramentas e dispositivos

- Kali Linux:

O Kali Linux é um sistema operacional baseado no Debian que contém ferramentas nativas para testes de invasão, penetração, força bruta, dentre outras. Ele já vem pronto para uso e é amplamente utilizado por *hackers*, analistas e profissionais de segurança (FRAGA, 2017). Neste trabalho, foi utilizado para replicar ataques em ambientes Wi-Fi e *Bluetooth*. A ferramenta foi instalada como uma máquina virtual no Virtual Box.

- *Wireshark:*

> *"Wireshark* é o analisador de protocolo de rede mais utilizado no mundo. Ele permite que você veja o que está acontecendo em sua rede em um nível microscópico e é o padrão de fato em muitas empresas comerciais e sem fins lucrativos, agências governamentais e instituições educacionais" (WIRESHARK).

No projeto, a ferramenta foi utilizada em conjunto ao Kali Linux como forma de interceptação de pacotes do dispositivo alvo de ataques nos diferentes cenários.

- Computador:

  O computador foi utilizado no cenário WLAN e WPAN configurado como atacante através do Virtual Box que tornou possível a instalação da máquina virtual Kali Linux. Também foi alvo de ataque no cenário Wi-Fi. O sistema operacional utilizado é o Windows 10.

- *Access Point*:

  O AP utilizado para realizar as simulações de ataque Wi-Fi foi o Roteador *Linksys* WRT54G v2.2, figura 13. Ele possui suporte para WEP, WPA e WPA2 e sua configuração foi feita através do DD-WRT, que adiciona mais funções e capacidade para o roteador.

Figura 13 – Access Point Linksys WRT54G v2.2.
Fonte: Autoral.

- DD-WRT:

> "A DD-WRT é um *firmware Open Source* baseado em Linux, adequado para uma grande variedade de roteadores WLAN e sistemas embarcados. A ênfase principal reside no fornecimento da manipulação mais fácil possível, ao mesmo tempo em que suporta um grande número de funcionalidades dentro do quadro da respectiva plataforma de hardware utilizada.
> A interface gráfica do usuário está logicamente estruturada e é operada por meio de um navegador da Web padrão, portanto, até mesmo os não técnicos podem configurar o sistema em apenas alguns passos simples" (DD-WRT).

O DD-WRT é utilizado no cenário Wi-Fi em conjunto ao AP *Linksys* WRT54G.

- Adaptador *Wireless*

Para execução do ataque através do Kali Linux, foi necessária a utilização de um adaptador *wireless* para captar o sinal Wi-Fi. O dispositivo utilizado foi do modelo TP-Link TL-WN722N, figura 14.

Figura 14 – Antena wireless *TP-Link TL-WN722N.*
Fonte: SANTOS, 2015.

- Adaptador *Bluetooth*:

O adaptador *Bluetooth*, figura 15, utilizado em conjunto ao Kali Linux possibilita a identificação dos dispositivos *Bluetooth* disponíveis para realização do ataque.

Figura 15 – Adaptador *Bluetooth*.
Fonte: autoral.

- Os dispositivos utilizados no cenário *Bluetooth* são mostrados no quadro 3.

Quadro 3 – Dispositivos utilizados nos cenários Wi-Fi e *Bluetooth*.
Fonte: Autoral.

Dispositivo	**Descrição**
Caixa de som	A caixa de som é caracterizada como um dispositivo IoT e utiliza a tecnologia *Bluetooth* para comunicação.
Televisão	A televisão foi utilizada como dispositivo IoT e tem a opção de comunicação por meio da tecnologia *Bluetooth*.
Fone de ouvido	O fone de ouvido é um dispositivo IoT. A comunicação é realizada pela tecnologia *Bluetooth* e é do modelo *Level Active* da Samsung.
Celulares	Celulares foram utilizados como dispositivos IoT. Foram utilizados dois modelos de celulares: o primeiro é da marca Sony Ericsson Xperia, modelo U20a com sistema operacional Android 1.6. O segundo é da marca SAMSUNG, modelo Galaxy J7 Prime com Android 6.0.1.
Smartwatch	O *smartwatch* é um dispositivo IoT. Foi utilizado o GEAR S3 é um relógio da marca Samsung que, por possuir conectividade *Bluetooth*, possui funcionalidades equivalentes à de um celular.

3.7 Cenário Wi-Fi

O cenário de ataque utilizando o padrão IEEE 802.11 consiste em um dispositivo cliente que será alvo do ataque, um ponto de acesso *Linksys* WRT54G v2.2 configurado com determinado protocolo de segurança e o atacante hospedado em uma máquina virtual Kali Linux.

Os protocolos de segurança utilizados para quebra de chave são WEP, WPA e WPA2, com as opções de algoritmo de criptografia TKIP e CCMP (AES). As configurações são modificadas através da interface do DD-WRT, observada na figura 16.

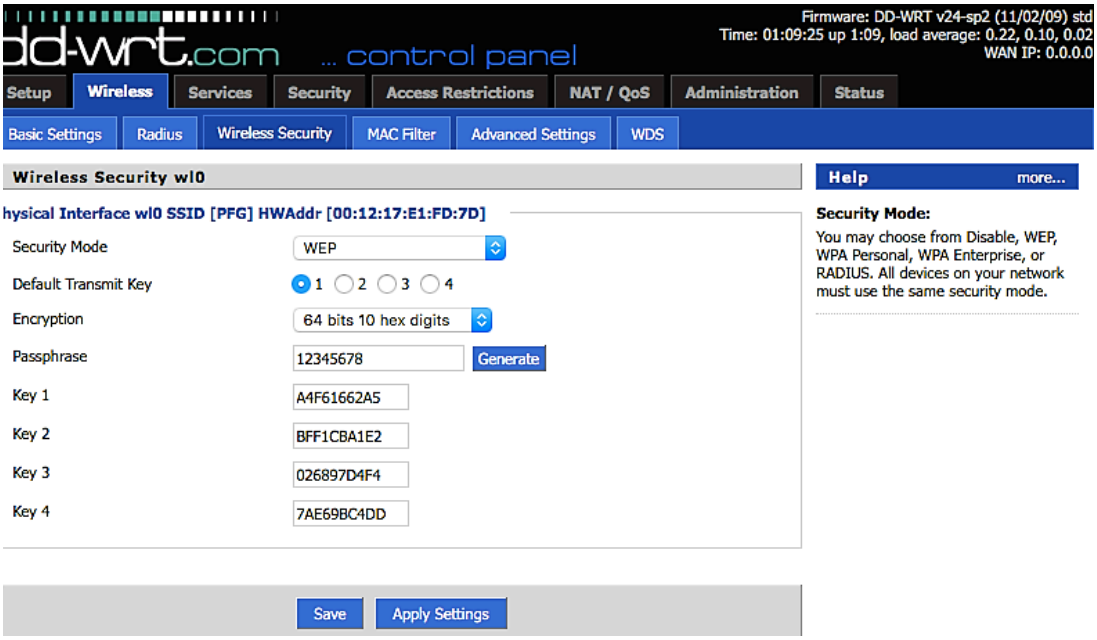

Figura 16 – Interface de configuração do AP.
Fonte: Autoral.

O cenário foi configurado em ambiente doméstico controlado como forma de simulação para os ataques realizados no laboratório UIoT da Universidade de Brasília. Os ataques escolhidos consistem na quebra de senha que garante acesso à rede WLAN, a interceptação de pacotes de dispositivo usuário da rede e negação de serviço como forma de afetar e impossibilitar a comunicação entre dispositivos participantes. Na figura 17 é mostrado o cenário.

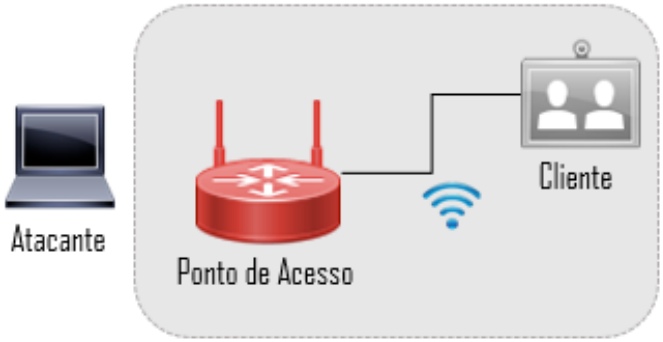

Figura 17 - Cenário Wi-Fi.
Fonte: Autoral.

3.8 Cenário UIoT

A arquitetura implementada no laboratório UIoT da Universidade de Brasília consiste nos dispositivos IoT, um concentrador de dados chamado de *gateway* que possui redundância com o objetivo de manutenção da disponibilidade da rede e o servidor em nuvem, RAISe (figura 22).

A comunicação entre os dispositivos consiste na transmissão dos dados coletados para o servidor através do concentrador de dados que realiza o tratamento de dados antes de encaminhá-los ao RAISe. Ocorre principalmente por tecnologias sem fio, como Wi-Fi, *ZigBee* e MQTT (*Message Queue Telemetry Transport*). Atualmente, a tecnologia *Bluetooth* não é utilizada por dispositivos do laboratório.

Dentre os dispositivos IoT documentados estão sensores de temperatura e umidade do ar e do solo, luminosidade, sensores de gases, de corrente e de nível de água de acordo com a tabela. Cada dispositivo envia ao RAISe as informações de medição em intervalos de tempo que variam desde 30 segundos até 3 minutos. As medições nos dispositivos não são armazenadas nos mesmos, são transmitidas ao servidor e, por este motivo, a escolha do ataque consistiu na quebra de chave e interceptação de dados. O ataque de negação de serviço não foi autorizado, pois a indisponibilidade da rede causa perdas de grande quantidade de dados, isto é, afeta o andamento de estudos desenvolvidos no laboratório UIoT.

Quadro 4 - Dispositivos IoT do laboratório UIoT.
Fonte: Autoral.

Dispositivo	Descrição	Protocolos	IP	Comunicação
Arduíno Uno	Sensor de temperatura, umidade e luminosidade	TCP	-	Ethernet

Dispositivo	Descrição	Protocolos	IP	Comunicação
Gateway – Raspberry (2 dispositivos para redundância)	Concentrador de dados	TCP, UDP, MQTT, ZigBee	172.16.9.61 172.16.9.64	Wi-Fi ou Ethernet
Arduíno nano	Sensor de temperatura, umidade, de gás (Carbônico e Butano) e de volume	ZigBee	-	Adaptador ZigBee
Arduíno nano	Sensor de temperatura, umidade	ZigBee	-	Adaptador ZigBee
Arduínomega	Sensor de umidade, temperatura, luminosidade e corrente	MQTT	172.16.9.92	Ethernet
Arduíno mini + ESP 8266	Sensor de corrente	TCP	172.16.9.81	Wi-Fi
Arduíno Uno	Sensor de umidade de solo, umidade e temperatura do ar e sensor do nível de água	MQTT	172.16.9.101	Ethernet

Figura 18 - Cenário UIoT.
Fonte: Autoral.

3.9 Cenário *Bluetooth*

O cenário de ataques *Bluetooth*, figura 19, foi configurado em ambiente doméstico controlado. Para realizar o ataque, é necessário o Kali Linux e um dispositivo *Bluetooth*. Através da ferramenta *scan*, o atacante consegue identificar o dispositivo e suas principais informações.

Figura 19 – Cenário Bluetooth.
Fonte: Autoral.

4. RESULTADOS E ANÁLISES

Este capítulo apresenta os ataques executados como forma de análise às tecnologias de comunicação abordadas neste trabalho. São expostos os procedimentos realizados nos cenários Wi-Fi (doméstico e laboratório UIoT) e no cenário *Bluetooth*, os resultados obtidos, a análise dos mesmos e, também, as recomendações de proteção a fim de minimizar a possibilidade de sucesso dos ataques abordados.

4.1 Cenário Wi-Fi

4.1.1 Quebra de chave

No cenário Wi-Fi, foram realizados ataques de quebra de chave. Primeiramente, foi realizado ataque para quebra de chave WEP. Com o comando *iwlist wlan0 scanning* é possível observar todas as redes disponíveis. O endereço MAC do AP alvo é 00:12:17:E1:FD:7D e se encontra no canal 6, como observado na figura 20 abaixo.

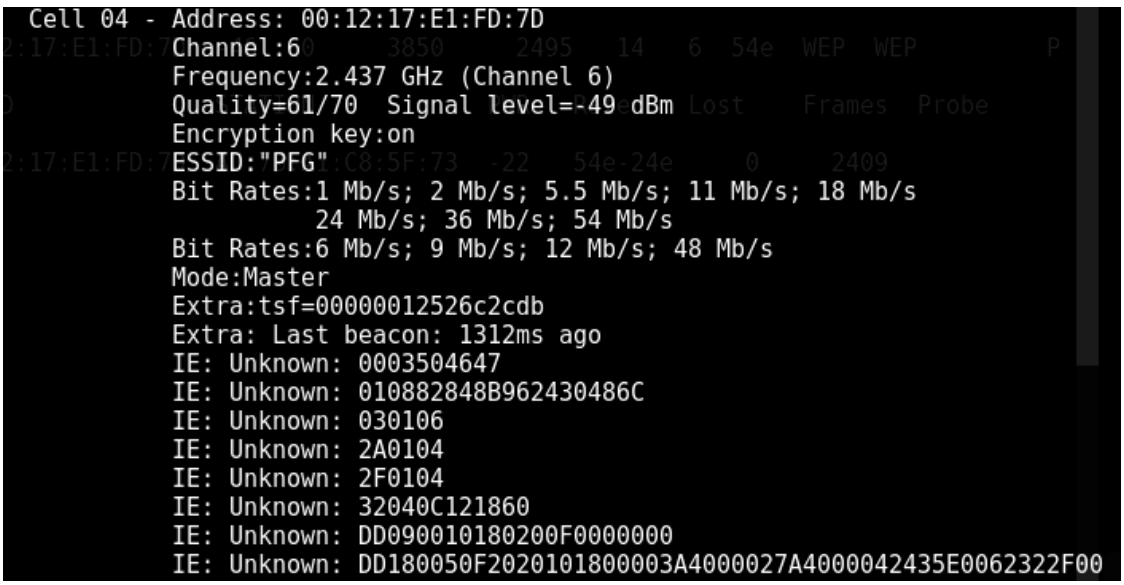

Figura 20 – Mapeamento da rede.
Fonte: autoral.

Após a identificação do endereço MAC do AP, a interface *wlan0* do Kali Linux é colocada em modo monitor ou promíscuo para realizar a escuta e ela passa a ser chamada de *wlan0mon*. Para isto, é utilizado o comando *airmon-ng start wlan0*.

A captura de pacotes é feita utilizando o comando *airodump-ng –bssid 00:12:17:E1:FD:7D -c 6 -w <nome do arquivo>wlan0mon*. O principal objetivo dele é

capturar IVs que posteriormente são utilizados pelo *aircrack-ng* para quebrar a chave (SALGUEIRO, 2008). O resultado pode ser observado na figura 21.

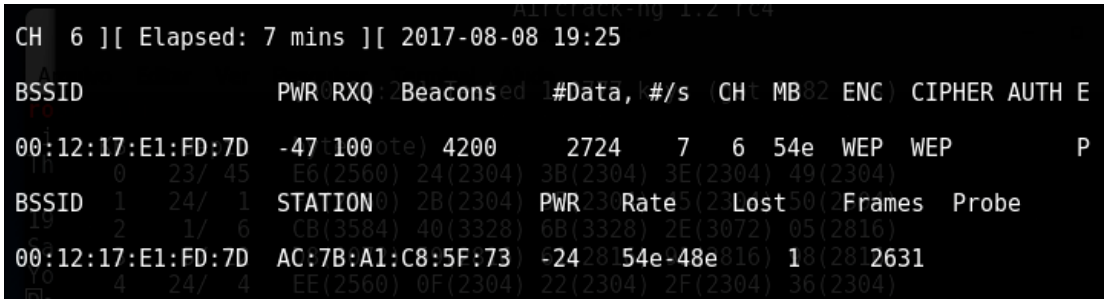

Figura 21 – *airodump-ng.*
Fonte: autoral.

Com o *airodump-ng*, é possível descobrir qual é a estação conectada ao AP, no caso, a máquina que foi utilizada como cliente. A estação é AC:7B:A1:C8:5F:73. Em seguida, é utilizado o comando *aireplay-ng -3 -b 00:12:17:E1:FD:7D -h AC:7B:A1:C8:5F:73 wlan0mon.* Com este comando, é possível realizar a injeção de requisições ARP que é identificada com o modo 3. Este ataque é eficiente para gerar IVs. Na figura 22, é obtido o resultado.

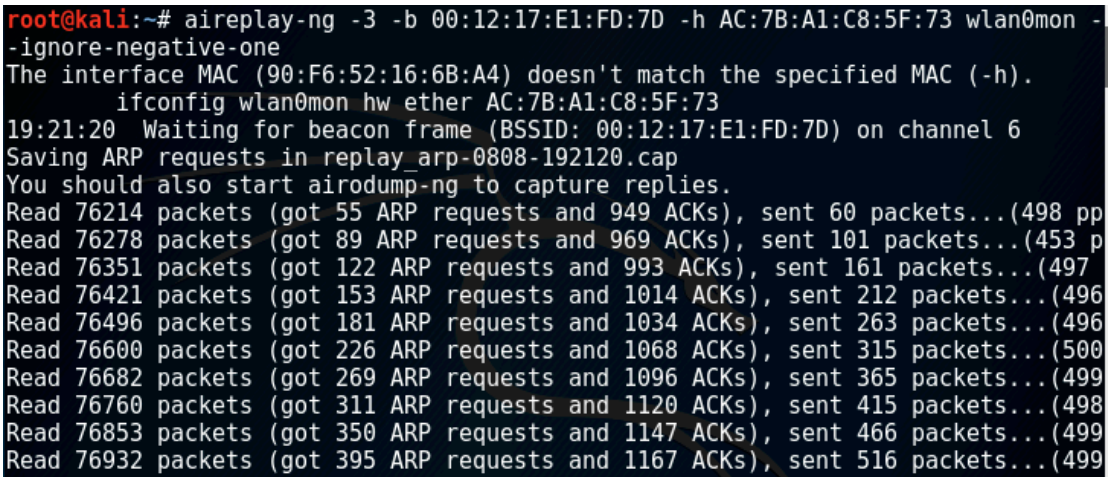

Figura 22 – *aireplay-ng.*
Fonte: autoral.

Por fim, é utilizado o comando *aircrack-ng <nome do arquivo>.cap* para encontrar a chave WEP. O resultado encontra-se na figura 23.

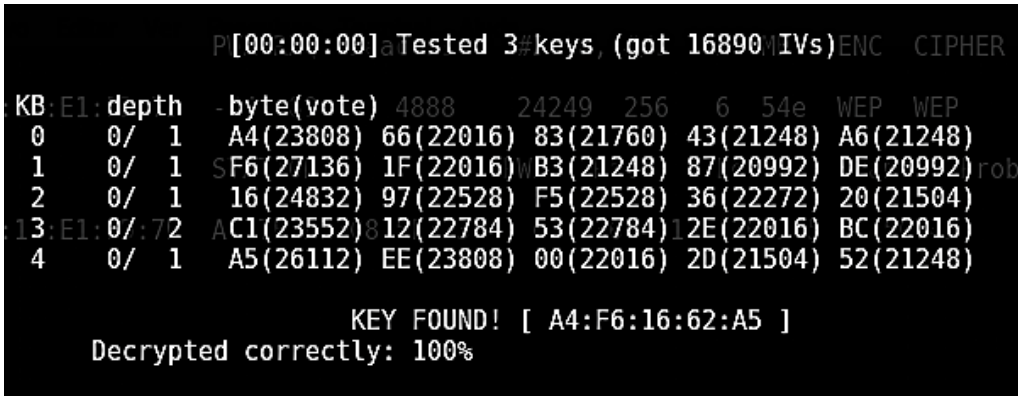

Figura 23 – Quebra de senha WEP.
Fonte: autoral.

Utilizando o mesmo cenário do ataque WEP, é, também, realizada a quebra de chave WPA. Para isto, o procedimento é o mesmo. A interface *wlan0* é colocada em modo monitor com o comando *airmon-ng start wlan0*. Com o comando *airodump-ng wlan0mon* é possível identificar informações das redes. O MAC do AP, o canal utilizado, o tipo de cifra e o SSID são respectivamente, 00:12:17:E1:FD:7D, 6, TKIP, PFG, mostrado na figura 24.

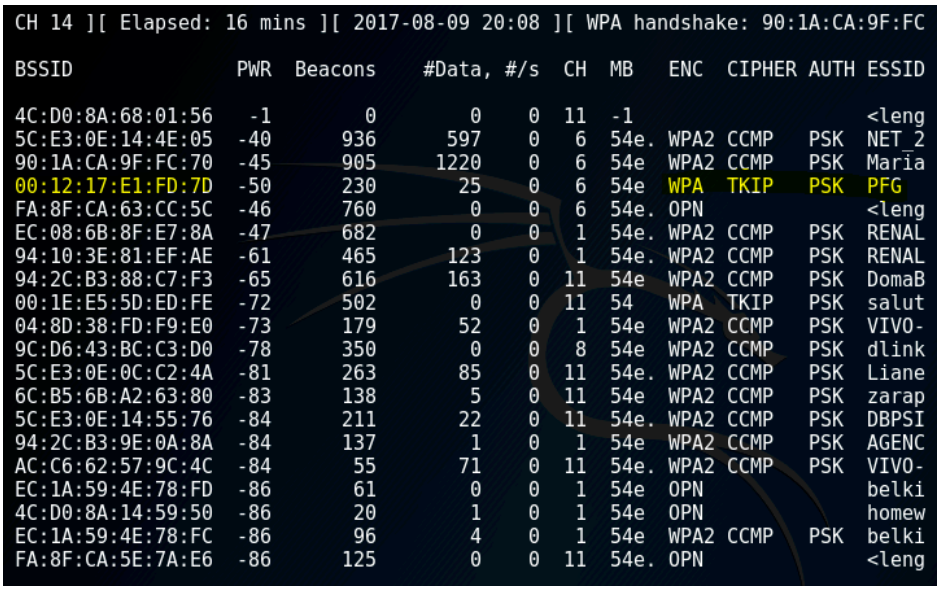

Figura 24 – *airodump-ng*.
Fonte: autoral.

Ao monitorar apenas a rede escolhida para o ataque, quando o cliente se conecta ao AP, a informação sobre MAC da estação, D0:33:11:6C:3A:86 (figura 25), é utilizada para o comando *aireplay-ng* (figura 26), que força a desautenticação do cliente para que, assim, haja uma nova conexão e o processo de *handshake* seja interceptado (figura 27).

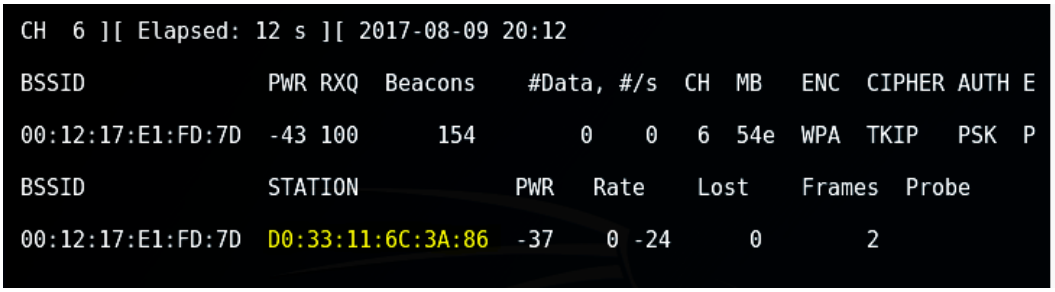

Figura 25 – Monitoramento da vítima.
Fonte: autoral.

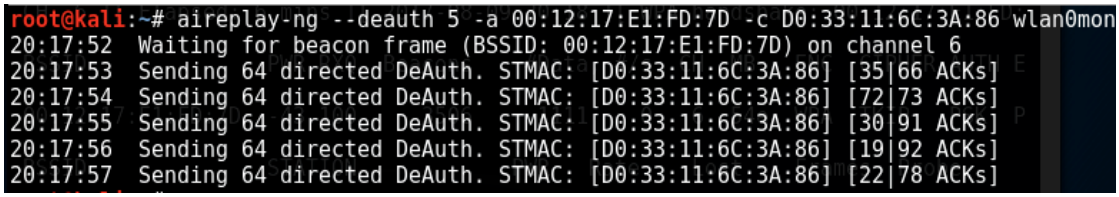

Figura 26 – Desautenticação forçada.
Fonte: autoral.

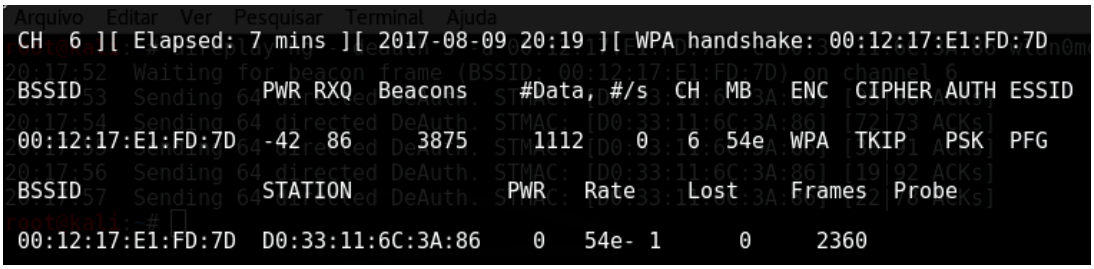

Figura 27 – Interceptação do *handshake*.
Fonte: autoral.

Ao interceptar o *handshake* entre o cliente e o AP, cria-se o dicionário do tipo mp64 com o objetivo de utilizá-lo na quebra da senha. O comando utilizado é mostrado na figura 28.

Figura 28 – Dicionário para quebra de chave WPA.
Fonte: autoral.

Para finalizar o ataque, o comando *aircrack-ng <arquivo.cap> -w <dicionário.lst>,* onde o arquivo.cap foi criado anteriormente no comando *airodump-ng,* é utilizado juntamente com o dicionário criado para o ataque de força bruta, onde as possibilidades de senhas contidas no arquivo serão testadas. O resultado é mostrado na figura 29 abaixo.

Figura 29 – Quebra de senha WPA.
Fonte: autoral.

Os mesmos procedimentos foram executados para o caso WPA2. O AP foi configurado com nome PFG com criptografia TKIP. Ao interceptar o *handshake* (figura 30) entre o cliente e o AP, foi utilizado o dicionário disponível na pasta *wordlist* do Kali Linux para quebra de senha, como mostrado na figura 35. O resultado é obtido na figura 32.

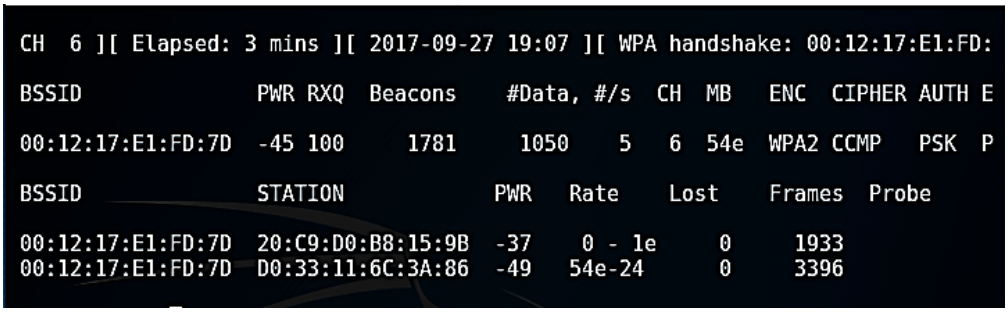

Figura 30– Interceptação do *handshake* WPA2.
Fonte: autoral.

Figura 31 – Dicionário para quebra de senha WPA2.
Fonte: autoral.

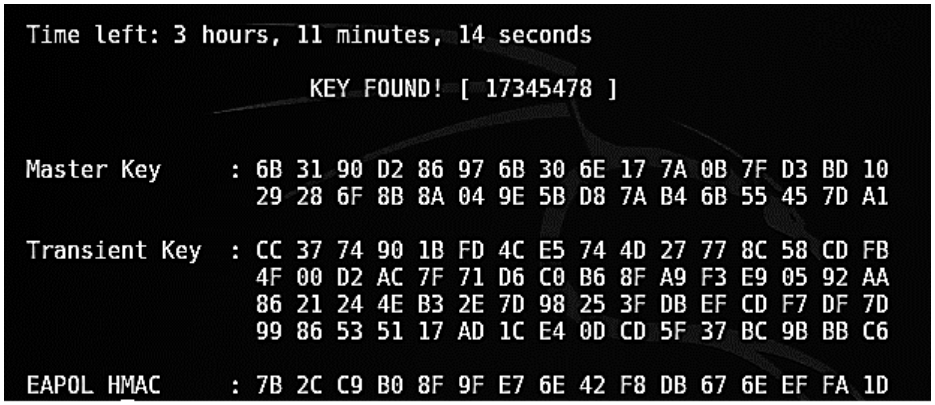

Figura 32 – Quebra de senha WPA2.
Fonte: autoral.

Nota-se a diferença entre o tempo utilizado para quebra de chaves WEP, WPA e WPA2. No caso WEP, foram necessários apenas três testes de chave e menos de um minuto para atingir o objetivo. No caso WPA com encriptação TKIP, 45.676 chaves foram testadas e um minuto e trinta e três segundos para encontrá-la. E, por último, o WPA2 com encriptação CCMP exigiu 198.597 chaves em três horas, onze minutos e catorze segundos para quebra de senha. O quadro abaixo demonstra a comparação entre as seguranças de chave e encriptação utilizadas.

Quadro 5 - Comparação da segurança de chaves.
Fonte: autoral.

Segurança de chave	Número de chaves testadas	Tempo de quebra de chave
WEP	3	11 segundos
WPA	45.676	1 minuto e 33 segundos
WPA2	198.597	3 horas, 11 minutos e 14 segundos

Dessa forma, as chaves com segurança WPA2 garantem a maior dificuldade de quebra quando comparada ao WEP e WPA, considerando que as senhas utilizadas possuem apenas números, ou seja, senhas de complexidade pequena e normalmente utilizadas em ambientes domésticos.

4.1.2 Interceptação de pacotes

A segunda parte do ataque consiste em interceptar os pacotes trafegados em uma rede onde o atacante logrou acesso. O *Wireshark* é uma ferramenta de captura de pacotes que, inicialmente, é capaz de capturá-los apenas do próprio dispositivo, isto é, não são capturados dados de outros clientes da rede, como pode ser visto na figura 33.

Para interceptação de pacotes de outro usuário em uma rede WLAN foi utilizado o ataque ARP- *Poisoning*. "O ataque do tipo *ARP-Poisoning* (ou *ARP-Spoofing*) é o meio mais eficiente de executar o ataque conhecido por *Man-In-The-Middle,* que permite que o atacante

intercepte informações confidenciais posicionando-se no meio de uma conexão entre duas ou mais máquinas" (PIFFARETTI, 2011).

Dessa forma, com o objetivo de interceptar as informações de um alvo pertencente à rede, o ataque *arpspoofing* é realizado por meio do Kali Linux que envia mensagens ARP (*Address Resolution Protocol*) com o intuito de associar o endereço MAC do dispositivo atacante ao endereço IP do alvo, dessa forma, o tráfego é enviado para o endereço IP do atacante ao invés de passar pelo endereço IP do *gateway* da rede.

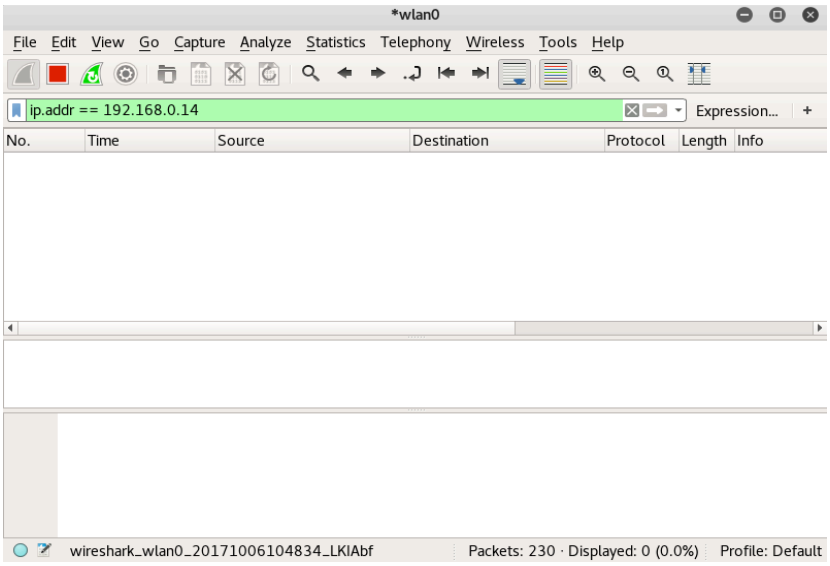

Figura 33 – Captura de pacotes do alvo sem a utilização de *arpspoofing*.
Fonte: Autoral.

Utilizando os comandos *arpspoof -i wlan0 192.168.0.1* e *arpspoof -i wlan0 -c host -t 192.168.0.14 192.168.0.1* alcançou-se o objetivo do ataque. A interface *wireless* wlan0 e o dispositivo alvo possui IP 192.168.0.14 em cenário doméstico com endereço de *gateway* 192.168.0.1 (figura 34).

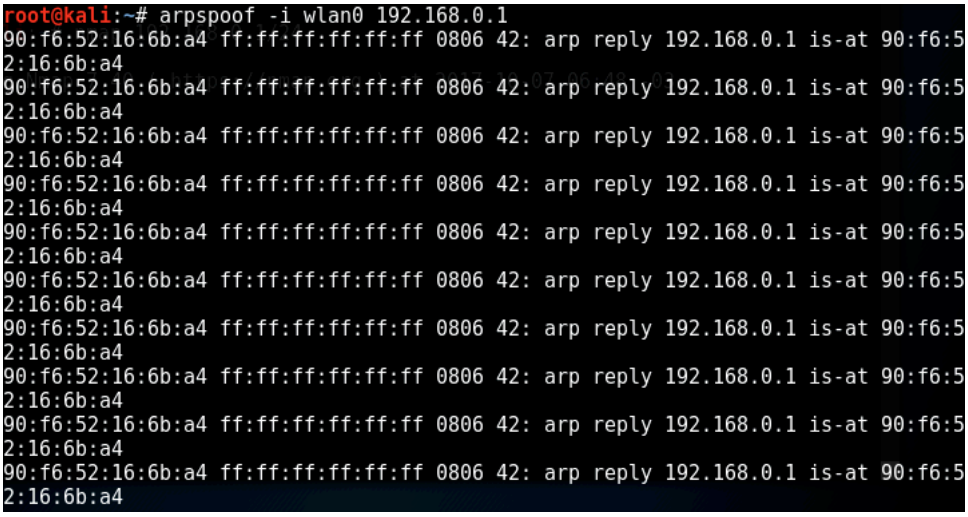

Figura 34 – Ataque *arpspoofing*.
Fonte: Autoral.

O *Wireshark* possibilitou capturar pacotes ARP enviados aos dispositivos conectados à rede. O pacote da figura 35, cujo sinal foi identificado pelo adaptador Wi-Fi instalado na máquina virtual, identifica o momento em que é anunciado o novo endereço MAC assumido pela rede 192.168.0.1.

A mensagem broadcast é enviada a partir do AP para anunciar que a rede 192.168.0.1 está associada ao MAC 90:F6:52:16:6B:A4, isto é, o dispositivo atacante está entre a comunicação do usuário e, por este motivo, o tráfego gerado pelo alvo é capturado. Nas figuras 36 e 37é possível observar estas informações.

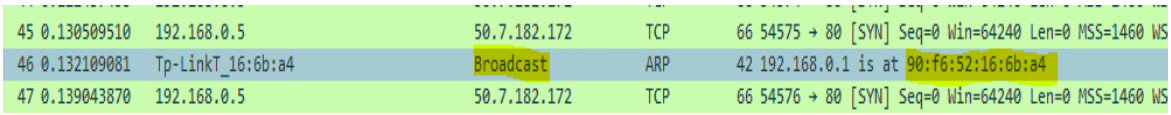

Figura 35 – Análise de pacotes *Wireshark*.
Fonte: Autoral.

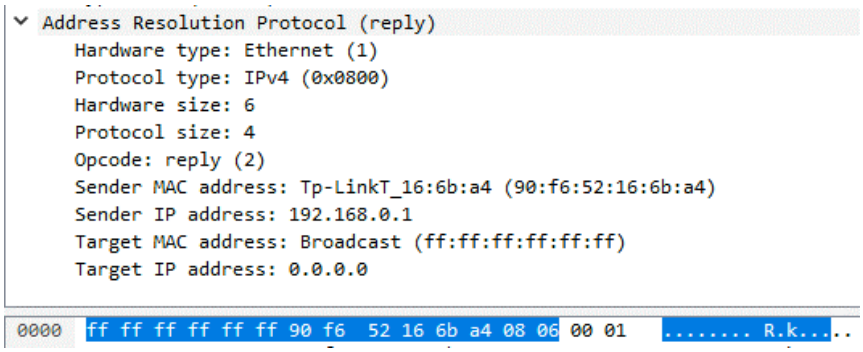

Figura 36 – Pacote ARP.
Fonte: Autoral.

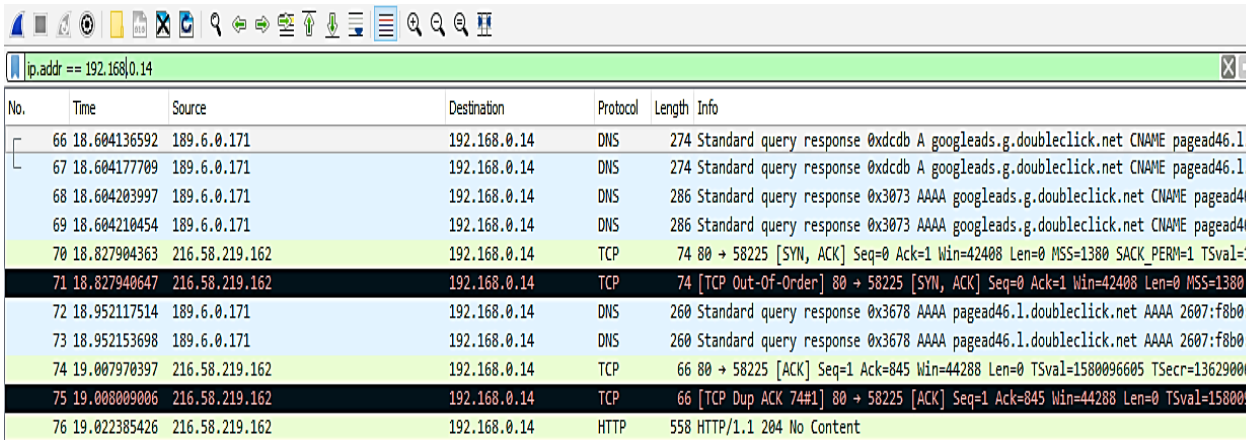

Figura 37 – Tráfego de usuário da rede.
Fonte: Autoral.

Como forma de analisar os pacotes, selecionou-se um pacote com o protocolo TLS *(Transporte Layer Security)* que, segundo Costa (2006), possui a função de garantir a segurança provendo a privacidade e a integridade de dados entre duas aplicações que se comuniquem pela internet e isto se torna possível por meio da autenticação das partes envolvidas e da criptografia dos dados transmitidos entre as mesmas.

Na interface gráfica do *Wireshark* é possível visualizar as camadas do pacote escolhido – figuras 38, 39 e 40 - para análise que contém dados importantes como endereços da fonte e destino, portas utilizadas para comunicação, a carga de dados TCP e, na camada *Secure Sockets Layer,* na qual são visualizadas as informações sobre o TLS, está disponível a informação contida no pacote, porém criptografado.

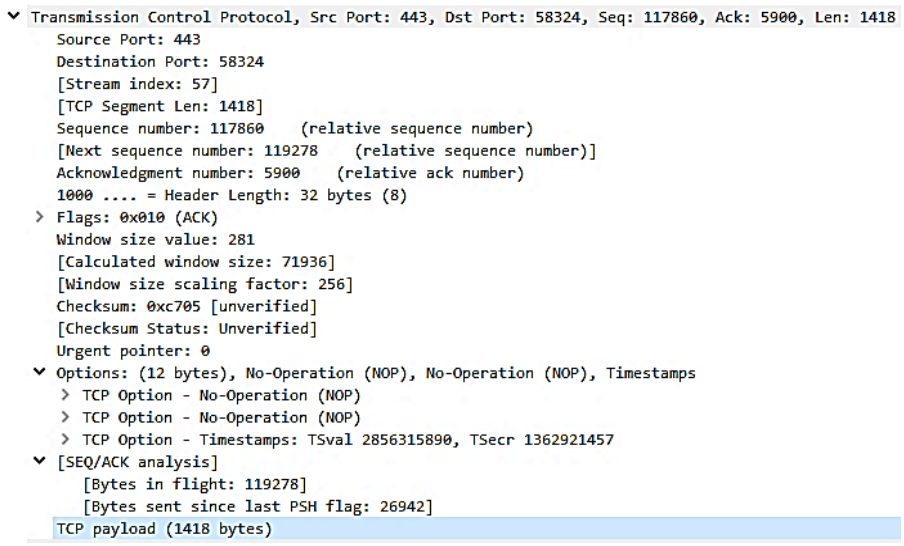

Figura 38 – Camada de transporte do pacote TLSv1.2.
Fonte: Autoral.

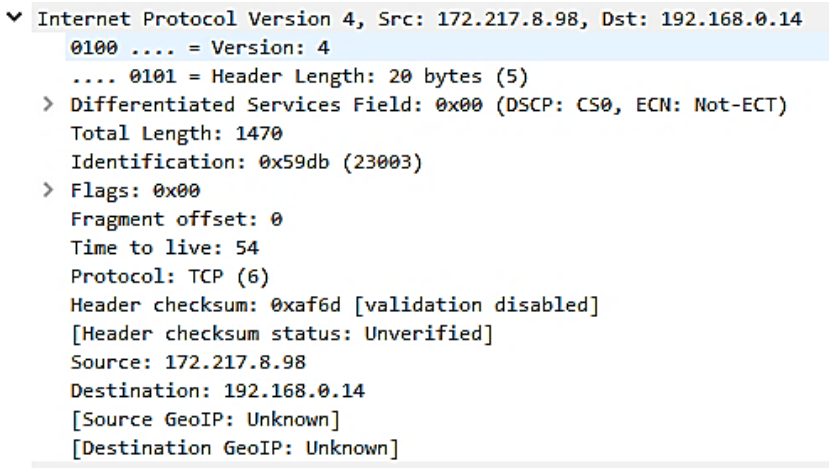

Figura 39 – Camada de rede do pacote TLSv1.2.
Fonte: Autoral.

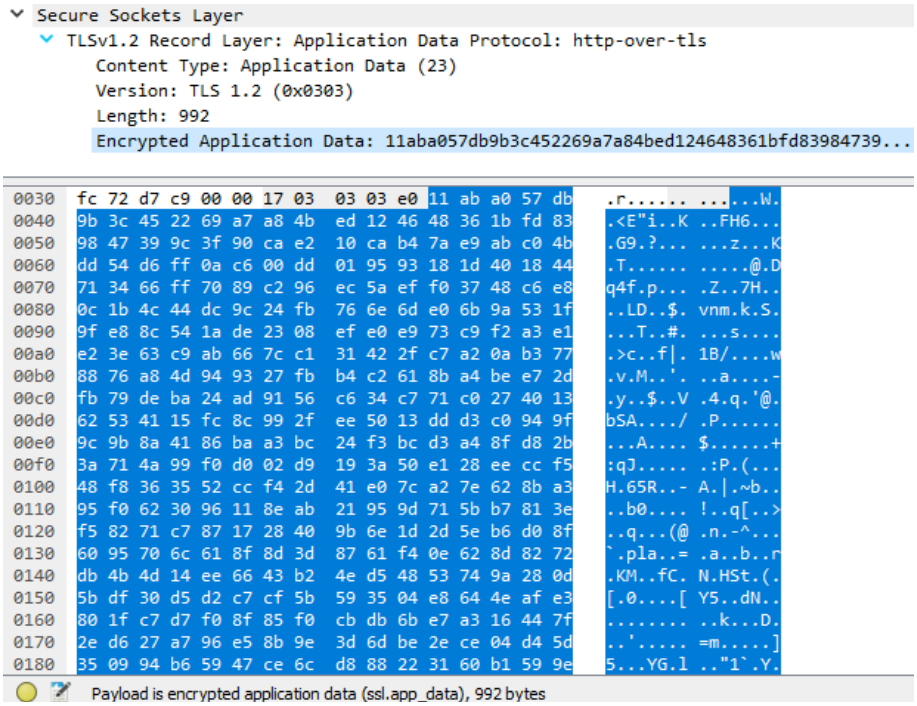

Figura 40 – Dados do pacote TLSv1.2.
Fonte: Autoral.

É importante observar que a interceptação de dados permite conhecer quais tipos são trafegados a partir e para o dispositivo alvo. Neste cenário, a utilização da criptografia TLS garantiu a confidencialidade das informações não permitindo o acesso às mensagens originais, porém afetou a privacidade do usuário permitindo identificar endereços, portas utilizadas, protocolos mais utilizados, bem como os acessos à Internet pelo usuário.

4.1.3 Negação de serviço

A parte final do ataque ao cenário Wi-Fi é baseada na indisponibilidade da rede, com o objetivo de garantir que os dispositivos conectados à WLAN seriam afetados e teriam comunicação interrompida. Com posse das informações obtidas sobre o AP nos procedimentos iniciais do ataque como BSSID, canal, ESSID, é possível realizar o ataque de negação de serviço no ambiente controlado em questão.

O MDK é uma ferramenta de prova de conceito para explorar fragilidades comuns do protocolo IEEE 802.11 (KALI TOOLS, 2014). Executou-se na interface *wlan0* em modo monitor e na figura 41 o comando *mdk3 wlan0mon a -i 00:12:17:E1:FD:7D*. A opção *a* transmite diversas requisições de autenticação à vítima e, como consequência, é capaz de indisponibilizar a rede WLAN. Estes resultados podem ser visualizados nas figuras 42 e 43.

Figura 41 – Envio de requisições para indisponibilidade da rede.
Fonte: Autoral.

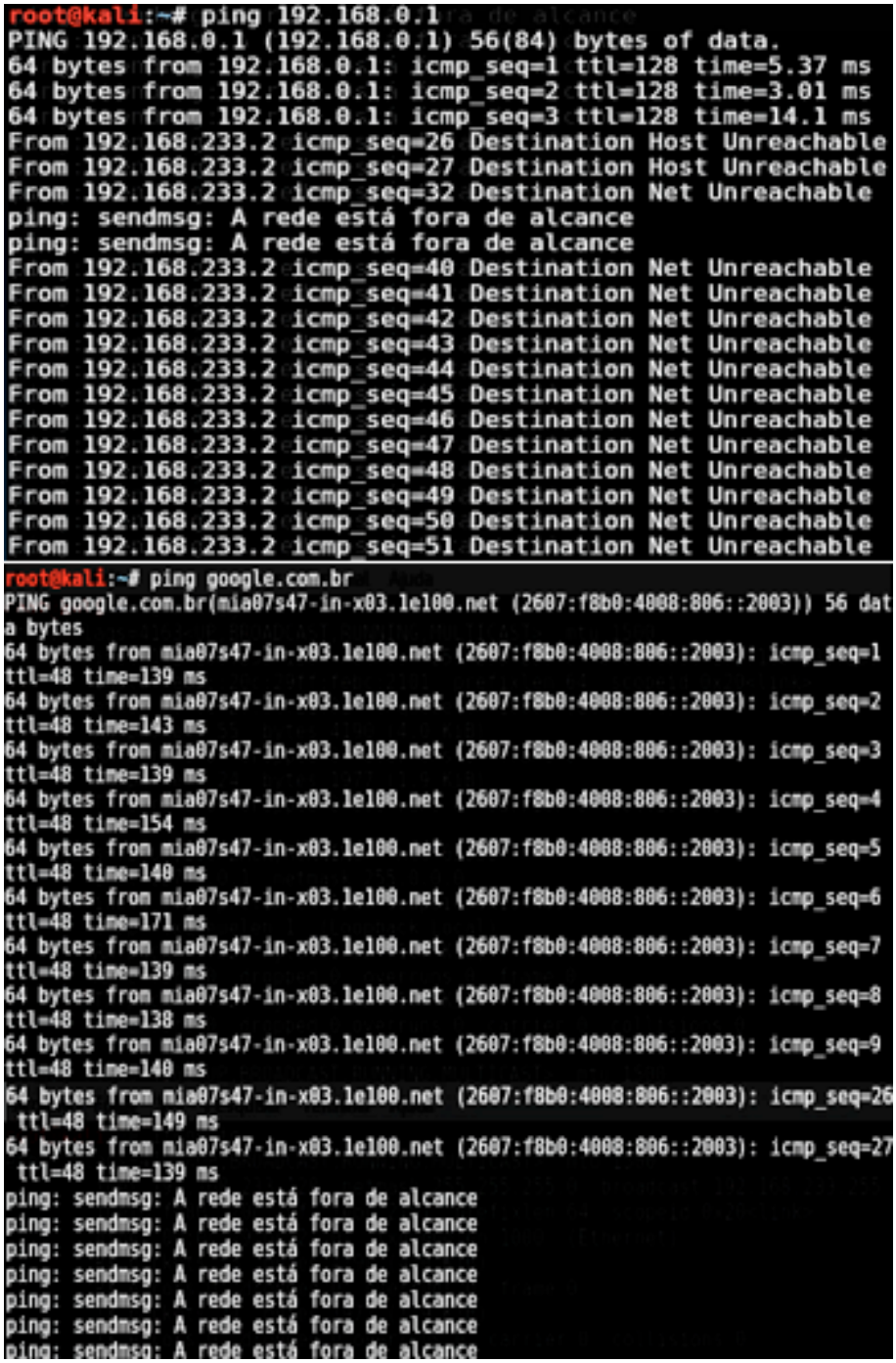

Figura 42 – Verificação da disponibilidade da rede durante o ataque.
Fonte: Autoral.

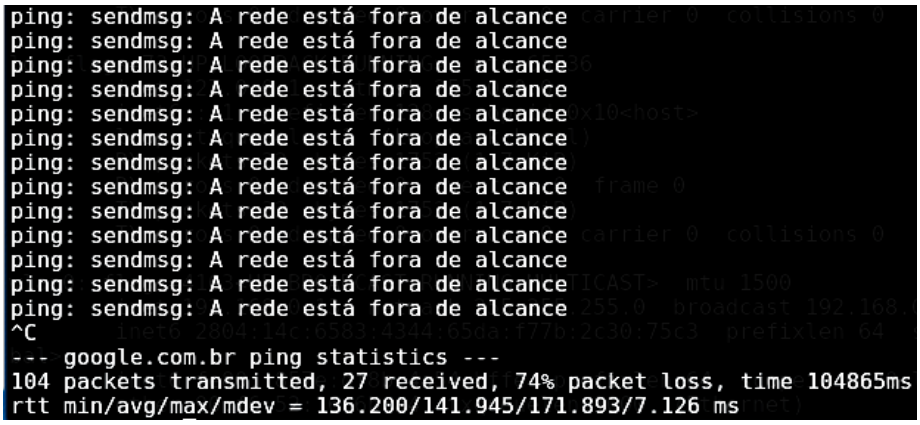

Figura 43 – Verificação da disponibilidade da rede durante o ataque.(continuação)
Fonte: Autoral.

Como forma de verificação da disponibilidade da rede, testou-se a comunicação com o AP cujo IP é 192.168.0.1 e, da mesma forma, a comunicação com o site google.com.br. Observa-se que de 104 pacotes transmitidos, apenas 27 foram recebidos, resultando em 74% de perda dos pacotes. Portanto com a utilização do ataque de inundação de requisições de autenticação, logrou-se a indisponibilidade da rede WLAN PFG.

Após a execução do comando, a partir da tentativa de autenticação de 70 clientes, a rede apresentou lentidão até indisponibilizar a rede com número máximo de requisição e, mesmo com a interrupção do ataque, a rede não foi reestabelecida, sendo necessário resetar o AP. O ataque de negação de serviço durou aproximadamente 1 minuto e 15 segundos registrando lentidão a partir de 40 segundos.

Como resultado, foram criados 807 clientes dos quais 73 lograram autenticação e 366 associaram-se à rede e nenhum dos clientes foi rejeitado, caracterizando, assim, nenhum mecanismo de defesa para inundação de requisições de autenticação.

Dessa forma, nota-se que uma rede WLAN doméstica, que possui diversos tipos de dispositivos conectados, possui vulnerabilidade considerável, pois com a utilização de uma ferramenta, no caso, o Kali Linux, é capaz de sobrecarregar o AP. É possível, em pouco tempo, afetar a comunicação entre dispositivos conectados e obter sucesso no ataque de negação de serviço.

Este tipo de ataque acarreta consequências que podem, por vezes, ser simples como uma indisponibilidade momentânea da rede, mas, também, a perda de dados importantes como, por exemplo, câmeras de monitoramento capazes de enviar alertas ao centro de vigilância ou dispositivos IoT que, em ambientes domésticos, realizam funções médicas como medições e envio de alertas aos familiares e ao serviço médico.

4.2 Cenário UIoT

4.2.1 Quebra de chave

O ataque realizado em laboratório consistiu primeiramente na quebra de chave da rede Wi-Fi com o objetivo de inserir-se na rede, executada da mesma forma que no cenário Wi-Fi. Para identificação do *Access Point* disponível, utilizou-se a interface *wireless* em modo monitor, com o comando *airmon-ng start wlan0.* Em seguida, para listar as redes visíveis no ambiente utiliza-se *airodump-ng wlan0mon*, cuja interface era nomeada como wlan0.

O AP utilizado no laboratório possui ESSID identificado como UIoT, com MAC 98:DE:D0:87:A2:34, que utiliza o canal 2 e é protegido pelo protocolo de segurança WPA2. Estas informações podem ser mostradas na figura 44.

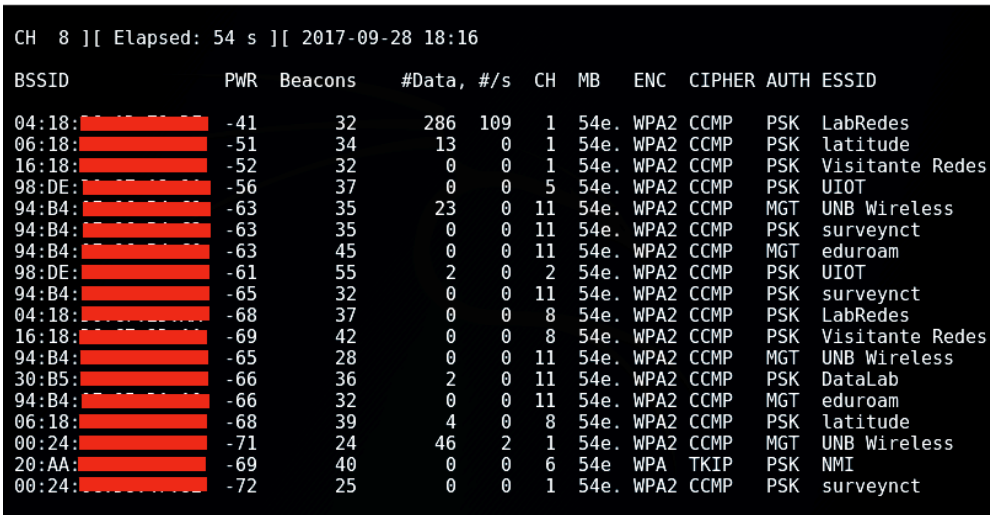

Figura 44 - Redes visíveis no ambiente UIoT.
Fonte: Autoral.

Este ataque envia pacotes a fim de forçar a desautenticação dos usuários associados a um ponto de acesso específico utilizando o comando *aireplay-ng deauth* que seleciona o MAC e interface alvo (figura 53). A desassociação de clientes é realizada com objetivo de capturar WPA ou WPA2 *handshake*, pois, dessa forma, há necessidade de autenticação dos clientes novamente. Este resultado é mostrado na figura 54.

Após a captura do *handshake*, cria-se um dicionário – figura 55 - que será utilizado para quebra de senha com o comando *crunch* a partir de um padrão de caracteres e com as possibilidades de números e letras especificados no comando. O arquivo texto utilizado possui nome de *newdicionario.txt*.

É importante ressaltar que houve necessidade de engenharia social para definição do tipo do dicionário criado. A UFRGS (2016) define engenharia social como práticas utilizadas com o objetivo de garantir acesso às informações importantes e confidenciais em organizações ou sistemas por meio da exploração da confiança das pessoas.

No caso, com a utilização de engenharia social foi possível determinar que a chave contém 10 caracteres entre letras (maiúsculas) e números, portanto o comando utilizado é *crunch 10 10 [possibilidade de caracteres na chave] -t A@@@@@@@@@@. -o /root/newdicionario.txt.* Nas figuras 45 e 46 é possível observar os procedimentos da quebra de chave.

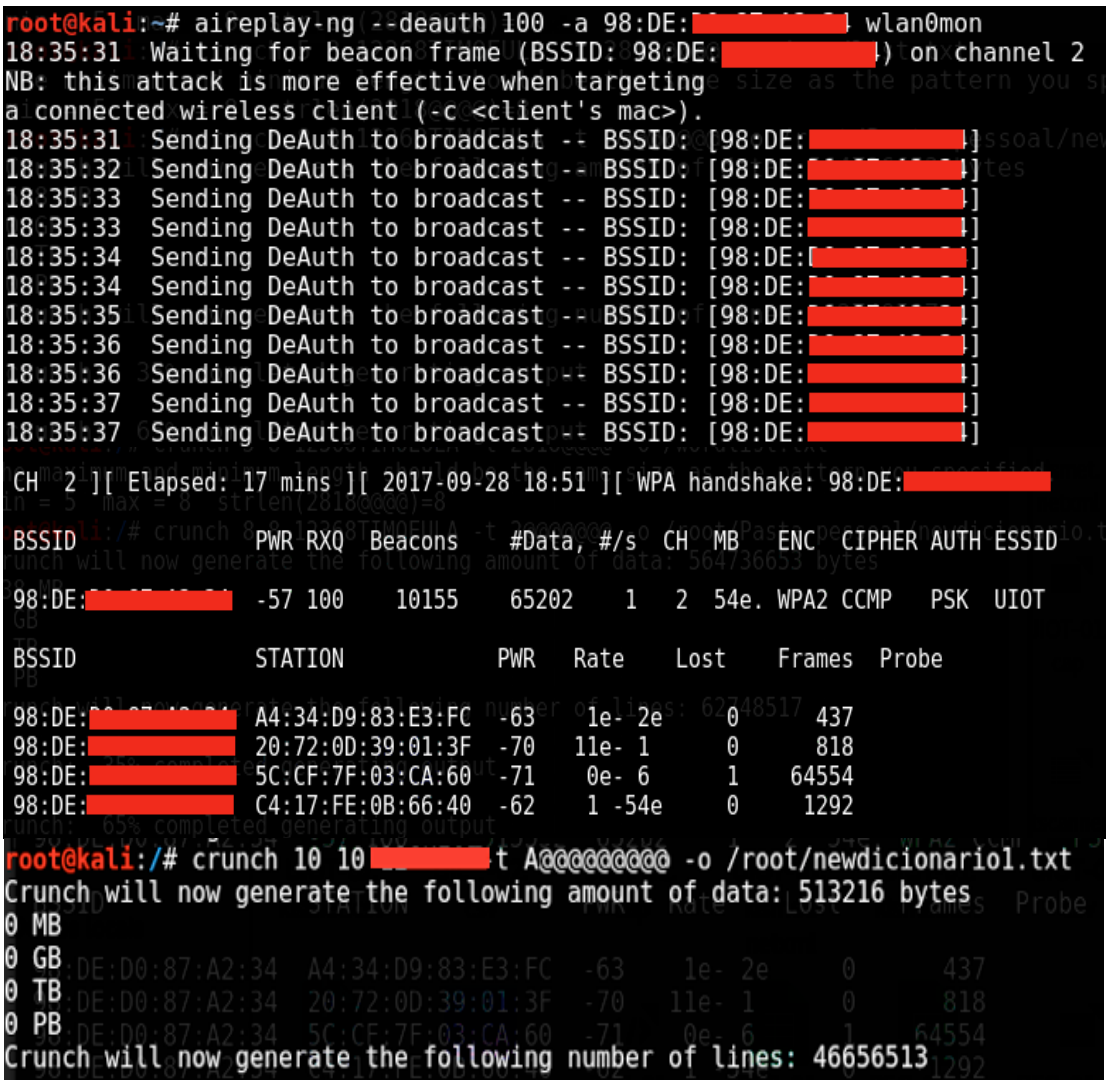

Figura 45 – Quebra de chave WPA2.
Fonte: Autoral.

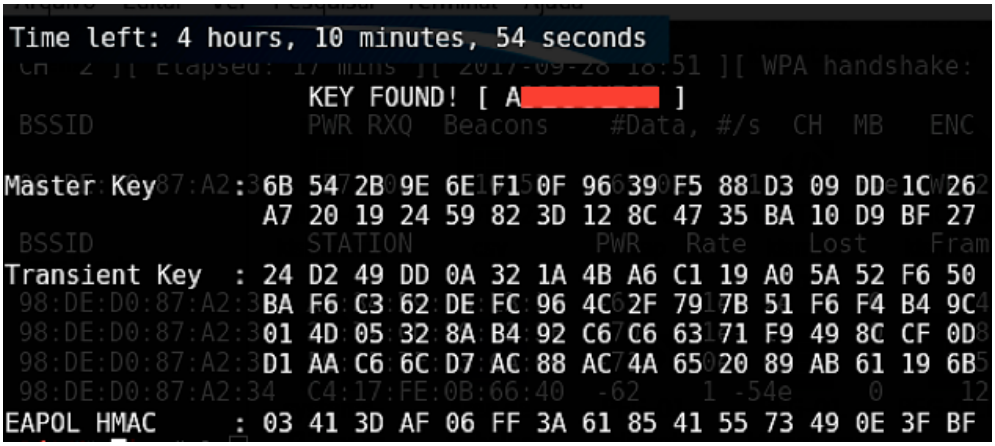

Figura 46– Quebra de chave WPA2.
Fonte: Autoral.

Nota-se na figura 46 que o tempo necessário para quebra de senha foi de, aproximadamente, 4 horas, 10 minutos e 54 segundos. Quando comparado com as senhas testadas no ambiente doméstico anterior, a senha do laboratório possui complexidade maior (letras maiúsculas, minúsculas e números) e, por este motivo, houve a necessidade da utilização de engenharia social para determinar o padrão da chave, que inicia com o caractere A, bem como seu tamanho.

4.2.2 Interceptação de dados

Para realizar a interceptação de dados no UIoT, foram utilizados os mesmos passos que no ambiente controlado. Com o comando *arpspoof–i wlan0 172.16.9.200* foi possível realizar o ARP *spoofing* da rede. O IP 172.16.9.200 é o endereço da rede Wi-Fi do laboratório UIoT.

Nas figuras 47 e 48, é possível observar os pacotes ARP capturados no *Wireshark* após o *spoofing* da rede, além de mostrar o pacote que assume o novo endereço MAC. A rede 172.16.9.200 é associada ao endereço MAC 90:F6:52:16:6B:A4, no caso, o do atacante.

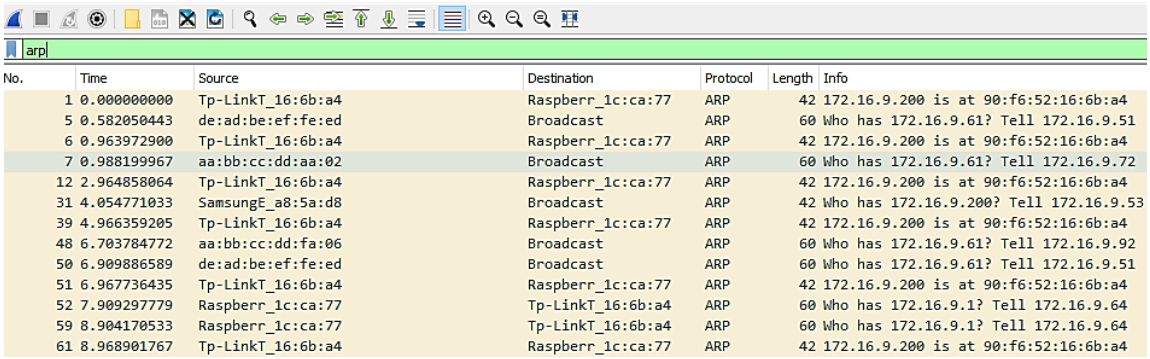

Figura 47 – Pacotes ARP capturados.
Fonte: autoral.

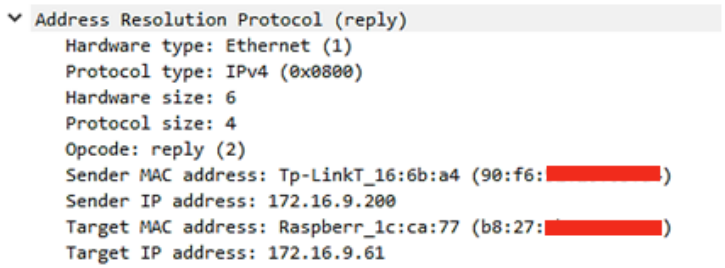

Figura 48– Pacote ARP no UIoT.
Fonte: autoral.

Com o *sniffer*, foi possível identificar o dispositivo chamado *gateway* na rede, que possui IP 172.16.9.61 e é do fabricante *RaspberryPi Foundation*. Ele é responsável por receber dados dos dispositivos IoT e enviá-los para o servidor RAISe.

O protocolo de transporte predominante nesta rede é o UDP. Ele é utilizado em conjunto com o XML (*eXtensible Markup Language*), que é uma linguagem de marcação utilizada para criação de documentos com dados organizados hierarquicamente (PEREIRA, 2009).

Os pacotes interceptados nas figuras 49 a 54 consistem em mensagens de descoberta de dispositivo pela rede, isto é, um dispositivo IoT anunciou à rede UIoT que faz parte da rede 172.16.9.0. Neste caso, o endereço é fe80::5ee:fe65:efc9:d28e para o IPv6 e 172.16.9.55 para o IPv4 identificados nas figuras abaixo. Trata-se do mesmo dispositivo configurado com as duas versões de endereço IP e tem como destino endereços identificados como *Group Address (multicast/broadcast) IPv4mcast_[endereçoMAC]* e *IPv6mcast_[endereçoMAC]*, conforme as figuras 50 e 52, o que comprova o anúncio do dispositivo à rede UIoT ao endereço multicast.

111	15.931368737	172.16.9.55	239.255.255.250	UDP/XML	58057 → 3702 Len=1078
112	16.237794470	fe80::5ee:fe65:efc9:d28e	ff02::c	UDP/XML	58058 → 3702 Len=1093
114	16.341299468	172.16.9.55	239.255.255.250	UDP/XML	58057 → 3702 Len=1078
116	16.646164067	fe80::5ee:fe65:efc9:d28e	ff02::c	UDP/XML	49774 → 3702 Len=624
117	16.646924034	172.16.9.55	239.255.255.250	UDP/XML	49773 → 3702 Len=624
118	16.748127277	fe80::5ee:fe65:efc9:d28e	ff02::c	UDP/XML	49774 → 3702 Len=624
119	16.749331932	172.16.9.55	239.255.255.250	UDP/XML	49773 → 3702 Len=624
120	16.850596608	fe80::5ee:fe65:efc9:d28e	ff02::c	UDP/XML	49774 → 3702 Len=624

Figura 49 – Pacote UDP/XML.
Fonte: autoral.

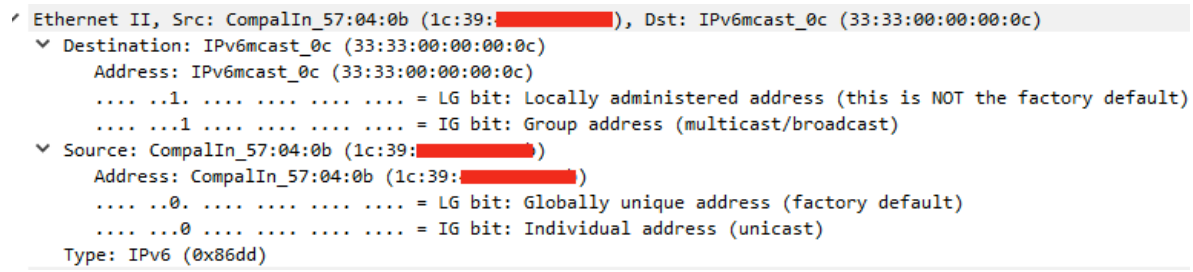

Figura 50 - Camada Ethernet pacote IPv6 UDP/XML.
Fonte: autoral.

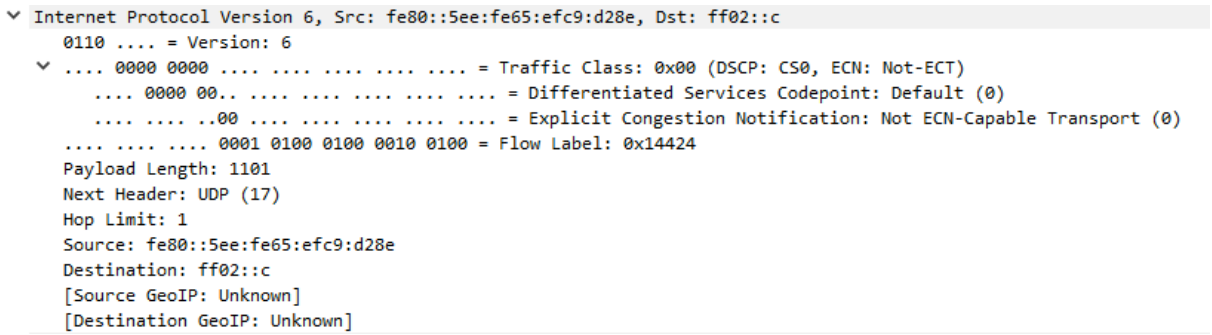

Figura 51 - Camada IP pacote IPv6 UDP/XML.
Fonte: autoral.

Figura 52 – Camada Ethernet pacote IPv4 UDP/XML.
Fonte: autoral.

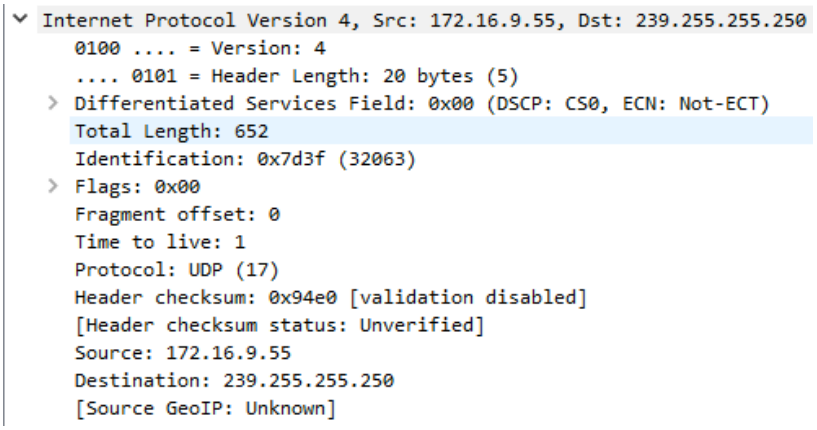

Figura 53 – Camada IP pacote IPv4 UDP/XML.
Fonte: autoral.

Figura 54 – Pacote XML (continuação).
Fonte: autoral.

O envelope SOAP *(Simple Object Access Protocol)* é um protocolo para comunicação entre dispositivos que possuem diferentes sistemas operacionais, tecnologias e linguagens de programação (W3SCHOOLS, 2017), o endereço utilizado no *xmlns:soap* é o padrão *http://www.w3.org/2003/05/soap-envelope*.

O cabeçalho SOAP é composto pelo WSA *(Web Services Addressing)* cuja função é selecionar determinado serviço e porta em um servidor SOAP (W3C, 2006). Os campos WSA habilitados neste pacote são*:* destinatário, ação e identificação da mensagem, respectivamente, *wsa:To,*wsa:*Action*e*wsa: MessageID.*Todos os campos são compostos pelo

Schemas. O WSD (*Web Services Dynamic Discovery*) também compõe o cabeçalho SOAP. Ele possui número de identificação de instância, sequência e número da mensagem.

No corpo da mensagem mostra-se o pacote *Hello* que possui campos WSA e WSD que anuncia a descoberta de um dispositivo. Neste caso, nota-se que o dispositivo é identificado como um computador, o endereço é mostrado no campo *wsa: Address*, o endereço é fe80::5ee:fe65:efc9:d28e para o IPv6 e 172.16.9.55 para o IPv4 identificados no *link* HTTP e a porta utilizada é 5357.

Em posse das informações interceptadas como os endereços, os identificadores do dispositivo e a porta (5357) utilizada, possibilitam um ataque em que um dispositivo malicioso anuncia à rede, transmitindo pacotes com o padrão encontrado e é capaz de forjar uma identidade ou passar-se por outro dispositivo conhecido. Dessa forma, possibilita o envio de dados falsos ao servidor (RAISe) com o objetivo de prejudicar a coleta de informações e medições realizadas no laboratório.

4.3 Cenário *Bluetooth*

Como etapa inicial do ataque, o adaptador *Bluetooth* foi utilizado na máquina virtual Kali-Linux, e para iniciar o serviço *Bluetooth* na máquina virtual, foi utilizado o comando */etc/init.d/Bluetooth start*, como pode ser observado na figura 55. O primeiro procedimento realizado no cenário *Bluetooth* consistiu na verificação da interface utilizada no Kali Linux e identificação de dispositivos IoT disponíveis por meio dos comandos e *hciconfig* e *hcitoolscan.*

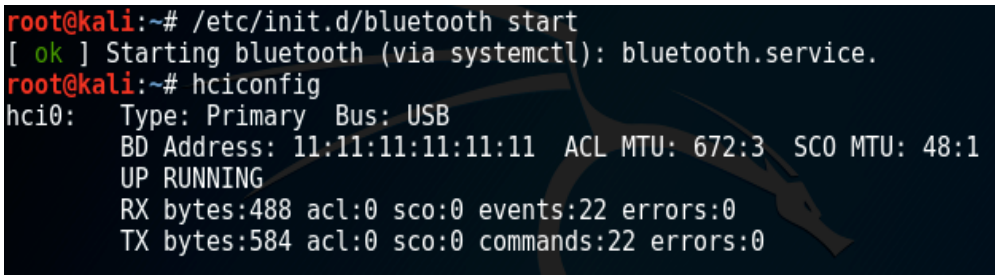

Figura 55 – Início do serviço *Bluetooth* e interface do Kali Linux.
Fonte: Autoral.

Os dispositivos visíveis para ser alvo do ataque de escaneamento são: uma caixa de som, uma televisão, dois celulares, um relógio e um fone de ouvido que possuem nome e MAC, respectivamente, SB510 e FC:58:FA:79:C0:CD,TV*Bluetooth* e 1C:5A:3E:0D:AF:5F, Galaxy J7 Prime e 98:39:8E:91:51:6E, X10 mini pro e 58:17:0C:B7:D7:EB, Gear S3 e

40:D3:AE:A7:DD:33, Samsung *Level Active* e 28:83:35:50:F6:63, conforme as figuras 56 e 57 a seguir.

Figura 56 – Dispositivos IoT visíveis.
Fonte: Autoral.

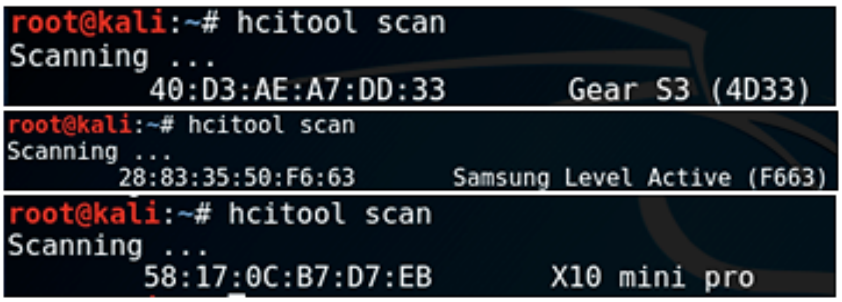

Figura 57 – Dispositivos IoT visíveis (contiuação).
Fonte: Autoral.

O comando *btscanner* – figura 58 - permite a realização do *scanner* da rede e, ao selecionar opção *inquiry scan* – figura 59 - no dispositivo encontrado, tem-se como resultado a descrição completa dos dispositivos alvos, sem a necessidade de pareamento, basta que o dispositivo *Bluetooth* esteja visível na rede. É possível obter informações como endereço MAC, nome do dispositivo, classe, serviços, dentre outras.

O procedimento foi repetido para todos os dispositivos citados. Foi feito o pareamento entre o celular Samsung e a televisão e foi possível observar alterações no campo *Features* nos dois dispositivos. Para os demais dispositivos que não foram pareados, este campo não é mostrado. As figuras 60 a 62 mostram os resultados do escaneamento da rede WPAN, são identificados o endereço MAC do dispositivo alvo e atacante, o nome, a classe, os serviços, a versão do *Bluetooth* e seu fabricante. Os resultados obtidos constam no quadro 4.

```
root@kali:~# btscanner
Opening the OUI database
Reading the OUI database
Finished reading the OUI database
```

Figura 58 - Início do processo *scan.*
Fonte: Autoral.

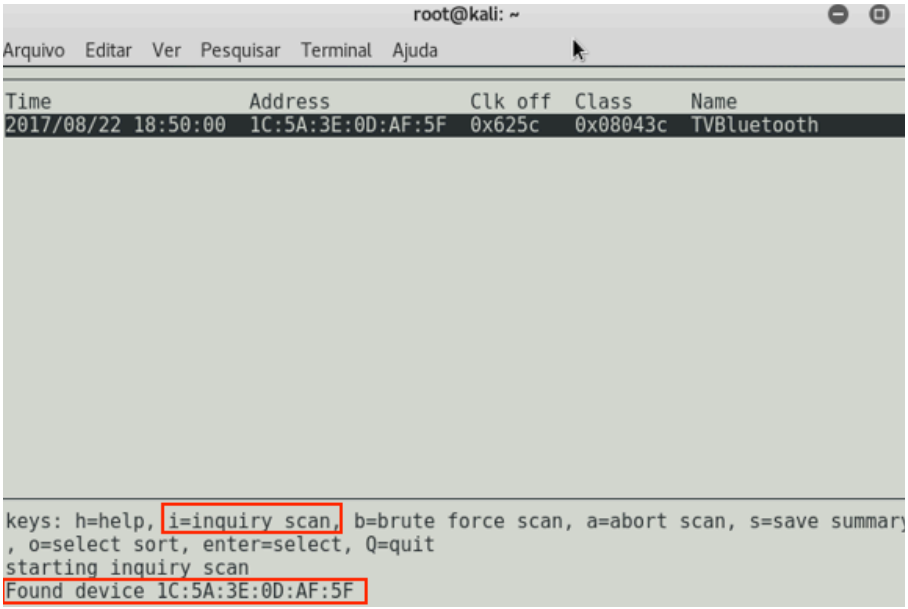

Figura 59 – *Inquiry scan*.
Fonte: Autoral.

```
RSSI:    +0   LQ:  000    TXPWR:  Cur   +0
Address:       FC:58:FA:79:C0:CD
Found by:      11:11:11:11:11:11
OUI owner:
First seen:    2017/08/22 18:59:59
Last seen:     2017/08/22 19:00:13
Name:          SB510
Vulnerable to:
Clk off:       0x1262
Class:         0x240410
               Audio-Video/Microphone
Services:      Rendering,Audio

HCI Version
-----------
LMP Version:  n/a (n/a) LMP Subversion: n/a
Manufacturer: n/a (n/a)

HCI Features
RSSI:    +0   LQ:  000    TXPWR:  Cur   +0
Address:       58:17:0C:B7:D7:EB
Found by:      11:11:11:11:11:11
OUI owner:
First seen:    2017/10/30 12:37:39
Last seen:     2017/10/30 12:38:17
Name:          X10 mini pro
Vulnerable to:
Clk off:       0x5b64
Class:         0x58020c
               Phone/Smart phone
Services:      Capturing,Object Transfer,Telephony

HCI Version
-----------
LMP Version: 2.1 (0x4) LMP Subversion: 0x1d1f
Manufacturer: Texas Instruments Inc. (13)
```

Figura 60– *Scan* dos dispositivos IoT.
Fonte: Autoral.

```
RSSI:    +0   LQ:  000    TXPWR:  Cur   +0
Address:       40:D3:AE:A7:DD:33
Found by:      11:11:11:11:11:11
OUI owner:
First seen:    2017/10/29 16:00:50
Last seen:     2017/10/29 16:00:50
Name:          unknown
Vulnerable to:
Clk off:       0x3eea
Class:         0x280704
               Uncategorised
Services:      Capturing,Audio

HCI Version
-----------
LMP Version:  n/a (n/a) LMP Subversion: n/a
Manufacturer: n/a (n/a)

RSSI:    +0   LQ:  000    TXPWR:  Cur   +0
Address:       28:83:35:50:F6:63
Found by:      11:11:11:11:11:11
OUI owner:
First seen:    2017/10/29 16:12:01
Last seen:     2017/10/29 16:12:38
Name:          Samsung Level Active (F663)
Vulnerable to:
Clk off:       0x1e24
Class:         0x240404
               Audio-Video/Headset
Services:      Rendering,Audio

HCI Version
-----------
LMP Version: 4.1 (0x7) LMP Subversion: 0x21c8
Manufacturer: Cambridge Silicon Radio (10)
```

Figura 61 – *Scan* dos dispositivos IoT.
Fonte: Autoral.

```
RSSI:    +0   LQ:  000    TXPWR:  Cur   +0
Address:       1C:5A:3E:0D:AF:5F
Found by:      11:11:11:11:11:11
OUI owner:
First seen:    2017/08/22 18:50:00
Last seen:     2017/08/22 18:53:29
Name:          TVBluetooth
Vulnerable to:
Clk off:       0x6259
Class:         0x08043c
               Audio-Video/Video Display and Loudspeaker
Services:      Capturing

HCI Version
-----------
LMP Version: 4.0 (0x6) LMP Subversion: 0x220e
Manufacturer: Broadcom Corporation (15)
-----------
Features:     0xbf 0xfe 0xcf 0xfe
    <3-slot packets> <5-slot packets> <encryption> <slot offset>
    <timing accuracy> <role switch> <sniff mode> <RSSI> <channel quality>
    <SCO link> <HV2 packets> <HV3 packets> <u-law log> <A-law log> <CVSD>
    <paging scheme> <power control> <transparent SCO> <broadcast encrypt>
    <EDR ACL 2 Mbps> <EDR ACL 3 Mbps> <enhanced iscan> <interlaced iscan>
    <interlaced pscan> <inquiry with RSSI> <extended SCO> <EV4 packets>
    <EV5 packets> <AFH cap. slave> <AFH class. slave> <LE support>
    <3-slot EDR ACL> <5-slot EDR ACL> <sniff subrating>
    <pause encryption> <AFH cap. master> <AFH class. master>
    <EDR eSCO 2 Mbps> <EDR eSCO 3 Mbps> <3-slot EDR eSCO>
    <extended inquiry> <LE and BR/EDR> <simple pairing>
    <encapsulated PDU> <err. data report> <non-flush flag> <LSTO>
    <inquiry TX power> <EPC> <extended features>
```

```
Found device 1C:5A:3E:0D:AF:5F
Found device 1C:5A:3E:0D:AF:5F
Found device 1C:5A:3E:0D:AF:5F
Found device 1C:5A:3E:0D:AF:5F

RSSI:    +0   LQ:  000    TXPWR:  Cur   +0
Last seen:     2017/08/22 19:21:31
Name:          Galaxy J7 Prime
Vulnerable to:
Clk off:       0x6735
Class:         0x5a020c
               Phone/Smart phone
Services:      Networking,Capturing,Object Transfer,Telephony

HCI Version
-----------
LMP Version: 4.1 (0x7) LMP Subversion: 0x2209
Manufacturer: Broadcom Corporation (15)

HCI Features
------------
Features:     0xbf 0xfe 0xcf 0xfe
    <3-slot packets> <5-slot packets> <encryption> <slot offset>
    <timing accuracy> <role switch> <sniff mode> <RSSI> <channel quality>
    <SCO link> <HV2 packets> <HV3 packets> <u-law log> <A-law log> <CVSD>
    <paging scheme> <power control> <transparent SCO> <broadcast encrypt>
    <EDR ACL 2 Mbps> <EDR ACL 3 Mbps> <enhanced iscan> <interlaced iscan>
    <interlaced pscan> <inquiry with RSSI> <extended SCO> <EV4 packets>
    <EV5 packets> <AFH cap. slave> <AFH class. slave> <LE support>
    <3-slot EDR ACL> <5-slot EDR ACL> <sniff subrating>
    <pause encryption> <AFH cap. master> <AFH class. master>
    <EDR eSCO 2 Mbps> <EDR eSCO 3 Mbps> <3-slot EDR eSCO>
    <extended inquiry> <LE and BR/EDR> <simple pairing>
    <encapsulated PDU> <err. data report> <non-flush flag> <LSTO>
    <inquiry TX power> <EPC> <extended features>
```

Figura 62 – *Scan* do celular Samsung e da televisão.
Fonte: Autoral.

Este ataque pode ocorrer em ambientes nos quais os dispositivos utilizam a comunicação *Bluetooth* e demonstra a vulnerabilidade de acesso aos dados sobre dispositivos como classe, fabricante e serviços oferecidos e pode ser utilizado como forma de mapeamento do ambiente. É possível identificar os dispositivos disponíveis e visíveis com o objetivo de conhecer detalhes importantes da rede WPAN alvo. Como resultado do escaneamento da rede obteve-se os seguintes dados dos dispositivos *Bluetooth* visíveis:

Quadro 6 - Dispositivos *Bluetooth*.
Fonte: autoral.

Dispositivo	**MAC**	**Serviço**	**Classe**	**Fabricante**	**Versão**
Televisão	1C:5A:3E:0D:AF:5F	*Capturing*	0x08043c	Samsung ElectronicsCo.Ltd	4.0
Galaxy J7 prime	98:39:8E:91:51:6E	*Networking, capturing, object transfer, telephony*	0x5a020c	Samsung ElectronicsCo.Ltd	4.1
Sony Ericsson U20a	58:17:0C:B7:D7:EB	*Capturing, object transfer, telephony*	0x58020c	Sony Mobile Communications AB	2.1
Caixa de som	5C:58:FA:79:C0:CD	*Rendering, áudio*	0x240410	No Vendor	3.0
Relógio	40:D3:AE:A7:DD:33	*Capturing, áudio*	0x280704	Samsung ElectronicsCo.Ltd	4.2
Fone de ouvido	28:83:35:50:F6:63	*Rendering, áudio*	0x240404	Samsung ElectronicsCo.Ltd	4.1

As versões mais novas do *Bluetooth* são menos vulneráveis e possuem melhores mecanismos de segurança e, por esse motivo, não foi possível efetuar um ataque mais invasivo. Foi possível observar que o *Bluetooth,* a partir das versões 4.0, possui um mecanismo de defesa próprio que consiste em ficar visível apenas durante um intervalo

específico de tempo e, por isso, torna-se mais vulnerável à captura das informações dos dispositivos no momento em que o *Bluetooth* dos mesmos é ativado.

Pode-se afirmar que o escaneamento da rede é uma forma eficaz para o mapeamento dos dispositivos inseridos em uma rede WPAN e, com a posse deste conhecimento, dá ao atacante a possibilidade de gerar um outro ataque direcionado à determinada classe ou fabricante, ou também, selecionar versões de *Bluetooth* que possuem vulnerabilidades conhecidas.

4.4 Recomendações para proteção mínima contra ataques

Considerando as duas amostras de cenários utilizadas, Wi-Fi e *Bluetooth*, existem formas de minimizar a possibilidade de sucesso dos ataques realizados.

No caso da quebra de chaves, considerando as opções de segurança WEP, WPA e WPA2, de acordo com resultados obtidos, deve-se utilizar o WPA2, pois se obteve maior dificuldade de quebra. Porém, dado que os ataques de quebra de senha foram realizados por meio da criação de dicionários, ou *wordlists,* que devem, necessariamente, conter a chave de acesso à rede WLAN, deve-se, então, configurar a senha de forma a aumentar a sua complexidade. Portanto, a escolha de chaves com o número de caracteres maior que o mínimo estabelecido (normalmente oito caracteres), a utilização de letras maiúsculas e minúsculas em conjunto com números e caracteres especiais diminuirá, portanto, a possibilidade de que o dicionário criado contenha a chave configurada no AP.

Considerando ataques *man-in-the-middle,* o ARP *Spoofing* (ou ARP *Poisoning),* segundo Vieira (2008) é um tipo de ataque em que o roteador de uma rede WLAN passa a enviar dados de um cliente legítimo ao atacante e este redireciona os dados recebidos ao destinatário original. Dessa forma, o ataque não é percebido pelo usuário. O autor lista sistemas operacionais vulneráveis a este tipo de ataque como Windows NT, XP, 95/98/2000, Linux, Netgeare AIX *4.3* e o único sistema operacional não vulnerável listado é o *Sun Solaris Systems*. Portanto, uma forma de defesa contra ARP *Spoofing* é a utilização de sistemas operacionais não vulneráveis. Vale ressaltar que o artigo foi escrito no ano de 2008, portanto a lista não está atualizada e o ataque foi realizado no Windows 10.

Outra forma de defesa exposta por Vieira (2008) é a utilização de rotas estáticas a serem configuradas no roteador ou *switch* de camada 3, pois as requisições ARPs ilegítimas seriam ignoradas, porém em redes domésticas onde usuários não possuem conhecimento de configuração de rotas, torna-se um desafio a sua implementação.

Existe também uma ferramenta chamada *Arpwatch*, descrito por Vieira (2008), que possui capacidade de detecção ataques ARP. A ferramenta monitora a atividade da rede e mantém uma base de dados dos pareamentos Ethernet/IP e reporta alterações via e-mail. O usuário com perfil de administrador é informado no momento em que uma nova máquina adquire um endereço da rede e envia um relatório com endereço IP e o MAC da nova máquina na rede.

Dado que para realização ataque de negação de serviço a única informação necessária é o endereço MAC do AP alvo, uma forma de prevenir-se, inicialmente, é a implantação de uma rede separada para os dispositivos IOT e configuração de uma rede WLAN oculta com o objetivo de torná-la invisível e, assim, garantir a integridade e manutenção da mesma.

Para ataques *Bluetooth* uma forma simples de evitar que dados sobre o dispositivo sejam acessíveis ao atacante é mantê-lo desabilitado nos momentos em que o usuário não esteja realmente utilizando esta tecnologia de comunicação, pois basta que esteja visível para que o ataque ocorra. Outra forma é manter atualizada a versão *Bluetooth*, pois versões antigas possuem maior vulnerabilidade de exposição de dados.

5. CONCLUSÕES E TRABALHOS FUTUROS

A execução dos ataques de quebra de chave, interceptação de pacotes e negação de serviço foi importante para o entendimento prático do cenário Wi-Fi tanto em ambientes domésticos quanto no laboratório UIoT da Universidade de Brasília. É possível concluir que em cenários onde existe a possibilidade de segurança de chave WEP, WPA e WPA2, a maior forma de segurança, atualmente, é o WPA2, mesmo com suas vulnerabilidades.

A interceptação de pacotes, com a utilização da ferramenta *Wireshark* e ataque ARP *Spoofing*, possibilitou o acesso aos protocolos, dados e endereços utilizados nas redes PFG e UIoT. No primeiro caso, a disponibilidade dos pacotes foi quebrada, porém a criptografia TLS garantiu a confidencialidade dos dados. No segundo caso, foi possível mapear o tipo de tráfego utilizado no cenário como também o pacote XML que apresentou dados importantes sobre a configuração de endereços IPv4 e IPv6 de um mesmo dispositivo interceptado e sua composição.

O ataque de negação de serviço realizado em ambiente doméstico controlado garantiu a indisponibilidade da rede por meio da ferramenta disponível no Kali Linux chamada *mdk3* que é capaz de enviar múltiplas requisições de autenticação ao AP da rede. Dessa forma, garante um ataque por inundação que resulta na queda permanente da rede fazendo-se necessário resetar o AP. Este ataque possibilitou demonstrar a fragilidade de uma rede doméstica que pode tornar-se indisponível em poucos minutos.

Em se tratando do cenário *Bluetooth*, o escaneamento com a utilização da ferramenta *scan* permitiu o mapeamento do ambiente e dispositivos disponíveis visíveis, sendo possível classificá-los por fabricante a partir do endereço MAC, classe e serviços oferecidos. Estas informações podem ser utilizadas de forma maliciosa por atacantes que desejam, de alguma forma, comprometer a rede e os dispositivos.

Como trabalhos futuros, pode-se realizar a execução de ataques aos cenários Wi-Fi e *Bluetooth* com a implementação das propostas de solução. Deve-se, também, implementar ataques *ZigBee*, padrão que foi estudado no projeto e definir mecanismos de defesa para o mesmo.

REFERÊNCIAS BLIBLIOGRÁFICAS

1 CERVANTES, Christian. **Um sistema de detecção de ataques Sinkhole sobre 6LoWPAN para Internet das Coisas.**Curitiba, 2014.

2 KUSHALNAGAR; MONTENEGRO; SCHUMACHER. **IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs):** Overview, Assumptions, Problem Statement and Goals, IETF RFC 4919, 2007.

3 IEEE 802 Working Group. **Standard for part 15.4:** Wireless médium Access control (MAC) and physical layer (PHY) specifications for low rate wireless personal area networks (LR-WPANs). Technical report, ANSI/IEEE 802.15, USA, Dezembro, 2008.

4 HADDAD; CHAKRABARTI; LAGANIER; PARK; KIM. **IPv6 over low Power wpan security analysis.** Technical report, USA, Março 2011.

5 PINZON, Alexandre. **Vulnerabilidade da segurança em redes sem fio.** 2009. Monografia – Faculdade de Informática, curso de Bacharel em Sistemas da informação, Porto Alegre, 2009.

6 GARCIA, Luis Guilherme Uzeda. **Redes locais sem fio que atendem ao padrão IEEE 802.11.** Disponível em: <https://www.gta.ufrj.br/grad/01_2/802-mac/>. Acesso em: 05 de maio de 2007

7 DUARTE, Carlos Anderson Andrade Duarte. **A evolução dos protocolos de segurança das redes sem fio:** do WEP ao WPA2 passando pelo WPA. 2010. Monografia – Escola Superior Aberta do Brasil, curso de Pós-Graduação em Redes de Computadores, Vila Velha 2010;

8 TANENBAUM, A. S. **Redes de Computadores.** Rio de Janeiro. Editora Campus. 4ª edição. 2003.

9 D-LINK. **D-Support for Wireless LAN. D-PR:** D-Linker Professional Resellers for wireless. 2002.

10 WI-FI ALLIANCE. **Deploying WPA and WPA2 in the Enterprise.** Disponível em: <http://www.Wi-Fi.org/white_papers/whitepaper 022705deployingwpawpa2enterprise> Acesso em: 20 de junho de 2017.

11 IEEE 802.15 WPAN Task Group 1 (TG1). Disponível em: <http://www.ieee802.org/15/pub/TG1.html>. Acesso em: 22 de junho de 2017.

12 SIG, Bluetooth. **What is *Bluetooth* technology?** Disponível em: <https://www.Bluetooth.com/what-is-Bluetooth-technology/how-it-works>. Acesso em: 22 de junho de 2017.

13 DIAS, Jaime. **Segurança em redes sem fios:** *Bluetooth***.** Disponível em: <https://web.fe.up.pt/~jaime/0506/SSR/seg_sem_fios_Bluetooth_v3.pdf>. Acesso em: 22 de junho de 2017.

14 BASTIEN, Jorge; BASTIEN, Lacroix; PROUX, Alexander. **Les protocoles réseau de I'Internet dês objets:** vulnérabilités connues.

15 NAKAMURA, Emilio Tissato; GEUS, Paulo Lício. **Segurança de Redes em Ambientes Cooperativos.** São Paulo: Editora Novatec. 2007.

16 SHENG, Zhengguo; YANG, Shusen; YU, Yifan; VASILAKOS, Athanasios V.; MCCANN, Julie A.; LEUNG, Kin K. **A survey on the IETF Protocol suíte for the Internet-of-things:** Standards, Challenges and Opportunities.

17 GASCÓN, David. **Security in 802.15.4 and *ZigBee* networks.**2009. Disponível em: <http://www.libelium.com/security-802-15-4-ZigBee/>. Acesso em: 22 de junho de 2017.

18 **Internet of Things:** Strategic, Research, Roadmap. 2009. Disponível em: <http://www.internet-of-things-research.eu/pdf/IoT_Cluster_Strategic_Research_Agenda_2009.pdf>. Acesso em: 28 de junho de 2017.

19 McDermott-Wells, P. Bluetooth Overview. **IEEE Potencials Magazine**. Dezembro de 2004, páginas 33-35.

20 SIG, Bluetooth. **Specification of the Bluetooth System**. 2017. Disponível em: <www.bluetooth.com>. Acesso em: 30 de Agosto de 2017.

21 SIQUEIRA, Thiago. **Bluetooth:** Características, protocolos e funcionamento. São Paulo, 2006.

22 Black Hat. **ZigBee Exploited**. 2015. Disponível em: <https://www.blackhat.com/docs/us-15/materials/us-15-Zillner-ZigBee-Exploited-The-Good-The-Bad-And-The-Ugly.pdf> Acesso em: 30 de Agosto de 2017

23 ZIGBEE ALLIANCE. **ZigBee Alliance.** 2013. Disponível em: <http://www.ZigBee.org/Home.aspx>. Acesso em: 9 de Setembro de 2017.

24 ZIGBEE ALLIANCE. **ZigBee Specification**. 2007. Disponível em: <http://www.ZigBee.org/Specifications/ZigBee/download.aspx>. Acesso em: 9 de Setembro de 2017.

25 MELO, Pablo. **Padrão IEEE 802.15.4:** a base para as especificações ZigBee, Wireless Hart e MiWi. 2017. Disponível em: <https://www.embarcados.com.br/padrao-ieee-802-15-4/>. Acesso em: 9 de Setembro de 2017.

26 FILHO, José Gonçalves Pereira. **A Camada Física do Padrão IEEE 802.15.4**. 2016. Disponível em: <https://inf.ufes.br/~zegonc/material/Redes%20de%20Sensores%20sem%20Fio/Resumo_Zegonc_Camada_Fisica.pdf>. Acesso em: 10 de Setembro de 2017.

27 VASQUES, Bruna Luisa, COUTINHO, Igor, LIMA, Manuela, CARNEVAL, Vitor. **ZigBee.** Rio de janeiro, 2010.

28 NENOKI, Eduardo. **ZIGBEE:** estudo da tecnologia e aplicação no sistema elétrico de potência. Curitiba, 2013.

29 BONAVENTURE, O. **Carrier-Sense-Multiple-Access-with-Collision.** Disponível em <https://scm.info.ucl.ac.be/release/cnp3/Book/0.2/html/lan/lan.html.>. Acesso em: 10 de Setembro de 2017.

30 **DFWMAC-DCF básico (CSMA/CA)**. Disponível em: <https://www.gta.ufrj.br/grad/00_2/ieee/CSMA.htm>. Acesso em: 10 de Setembro de 2017.

31 SASTRY, Naveen; WAGNER David. **Security Considerations for IEEE 802.15.4 Networks**. California, 2004.

32 Kennedy Institute of Ethics. Bioethics Thesaurus. Washington: KIE, 1995:9. Disponível em:<https://www.ufrgs.br/bioetica/confiden.htm>. Acesso em: 13 de Setembro de 2017.

33 M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. **A concrete security treatment of symmetric encryption:** analys is of the DES modes of operation. 1997.

34 ALECRIM, Emerson. **O que é Wi-Fi? (IEEE 802.11).** 2013. Disponível em: <https://www.infowester.com/wifi.php#funcionamento>. Acesso em: 30 de agosto de 2017.

35 EDUARDO, Carlos. **Topologias 802.11**. 2011. Disponível em: <https://www.wlan.com.br/?p=453>. Acesso em: 30 de agosto de 2017.

36 FERNANDES, Ivo. **Wi-Fi.** 2006. Disponível em: <https://paginas.fe.up.pt/~ee99207/Tecnologias/WLAN/Wi-Fib.html>. Acesso em: 30 de agosto de 2017.

37 JOBSTRAIBIZER, Flávia. **Desvendando as redes sem fio:** passo a passo como montar, configurar e usar uma rede sem fio. São Paulo: Digerati Books. 2010.

38 PAIM, Rodrigo R. **WEP, WPA E EAP**. 2011. Disponível em: <https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2011_2/rodrigo_paim/downloads/trabalho.pdf>. Acesso em: 31 de agosto de 2017.

39 CONTRACTI TI. **Tipos de ataques Wireless.** 2016. Disponível em: <http://www.contractti.com.br/tipos-de-ataque-wireless/>. Acesso em: 31 de agosto de 2017.

40 REIS, Lucas Nelson Ribeiro. **IEEE 802.11 e o *Wired Equivalent Privacy* (WEP).** 2008. Disponível em: <https://www.gta.ufrj.br/grad/08_1/ieee802-11/wep.html> Acesso em: 1 de setembro de 2017.

41 KUROSE, James F.; ROSS, Keith W. **Redes de computadores e a Internet: uma abordagem top-down.** 5. Ed. São Paulo: Addison Wesley, 2010.

42 SANTOS, Bruno P. et al. 2016. Disponível em: <http://homepages.dcc.ufmg.br/~mmvieira/cc/papers/internet-das-coisas.pdf>. Acesso em: 14 de setembro de 2017.

43 MAGALHAES, Gabriel G. M. S. de. **Estudo de segurança nos principais protocolos da Internet das Coisas.** 2016. Monografia - Departamento de Ciência da Computação, curso de graduação em Engenharia da Computação, Brasília, 2016.

44 ZAMPERLINI, Paulo Roberto Tercio; SANTOS, Guilherme Rezende dos. **Protocolos de segurança Wireless (WEP, WPA e WPA2):** um estudo comparativo. 2016. Seminário de Redes e Sistemas de Telecomunicações, Instituto Nacional de Telecomunicações – INATEL. 2016.

45 SANTOS, Pedro Paulo Martins dos. **Análise de segurança em Redes sem Fio e proposta de solução para o Laboratório da Engenharia de Redes de Comunicação.** 2015. Trabalho de Graduação – Departamento de Engenharia Elétrica, Faculdade de Tecnologia, UnB. Curso de graduação em Engenharia de Redes de Comunicação, Brasília, 2015.

46 FRAGA, Bruno. **Você sabe o que é Kali Linux?** 2016. Disponível em: <https://tecnicasdeinvasao.com/linux/kali-linux/voce-sabe-o-que-e-o-kali-linux/>. Acesso em: 18 de setembro de 2017.

47 Finjan Mobile. Disponível em: <https://www.finjanmobile.com/what-is-bluesnarfing/>. Acesso em: 21 de Setembro de 2017.

48 ZORZ, Zeljka.**Billions of *Bluetooth*-enabled devices vulnerable to new airborne attacks**. Setembro de 2017. Disponível em: <https://www.helpnetsecurity.com/2017/09/13/blueborne/>- Acesso em: 21 de Setembro de 2017.

49 **The Attack Vector "BlueBorne Exposes Almost Every Connected Device**". <https://www.armis.com/blueborne/> - ARMIS 13 de Setembro de 2017.

50 LINS Patrícia. **Aspectos de Segurança em Comunicações utilizando a tecnologia Bluetooth.** 2010. Disponível em: <https://patricialins.org/2010/04/24/aspectos-de-seguranca-em-comunicacoes-utilizando-a-tecnologia-Bluetooth/>. Acesso em: 26 de Setembro de 2017.

51 BlackArch. **Bluetooth tools**. 2013. Disponível em: <https://blackarch.org/Bluetooth.html>. Acesso em: 28 de Setembro de 2017.

52 GRÉGIO, André Ricardo. **Tecnologia Bluetooth e aspectos de segurança.** Disponível em: http://www.ic.unicamp.br/~ducatte/mo401/1s2009/T2/079779-t2.pdf>.Acesso em: 28 de Setembro de 2017.

53 VOJISLAV B. Miˇsi´c, JunFung, and Jelena Miˇsi´c. **MAC Layer Attacks in 802.15.4 Sensor Networks.** 2006. Disponível em: <http://www.scs.ryerson.ca/~jmisic/papers/chapterMisicFungMisicMAClayerAttacks.pdf>. Acesso em: 28 de Setembro de 2017.

54 JUNG, SANG SHIN**. Attacking and Securing Beacon-Enabled 802.15.4 Networks.** Thesis, Georgia State University, 2011. Disponível em: <http://scholarworks.gsu.edu/cs_theses/74>. Acesso em: 28 de Setembro de 2017.

55 VIDAL, Albert. **Anàlisi teòric de lês debilitats de seguret at Dels estàndards per a La Medició Intel ligent.** 2011. Disponível em: <http://upcommons.upc.edu/bitstream/handle/2099.1/16014/mem_PFC.pdf?sequence=4>. Acesso em: 02 de Outubro de 2017.

56 Wireshark. **About Wireshark**. Disponível em: <https://www.wireshark.org/#learnWS>. Acesso em: 09 de Outubro de 2017.

57 PIFFARETTI, Diego. 2011. Disponível em: <https://mundotecnologico.net/2011/11/01/entendendo-a-tecnica-de-arp-poisoning/>. Acesso em: 09 de Outubro de 2017.

58 KALI TOOLS. **MDK3**. 2014. Disponível em: <https://tools.kali.org/wireless-attacks/mdk3>. Acesso em: 09 de Outubro de 2017.

59 KALI TOOLS. **KillerBee**. 2014. Disponível em: <https://tools.kali.org/wireless-attacks/killerbee>. Acesso em: 10 de Outubro de 2017.

60 DD-WRT. Disponível em: <http://www.dd-wrt.com/site/content/about>. Acesso em: 10 de Outubro de 2017.

61 SALGUEIRO, Marcello Bontempo. **Aircrack-ng e sua família para quebrar WEP e WPA1.** 2008. Disponível em: <https://www.vivaolinux.com.br/artigo/Aircrackng-e-sua-familia-para-quebrar-WEP-e-WPA1>. Acesso em: 10 de Agosto de 2017.

62 COSTA, Bernardo. **Segurança na Internet. 2006.** Rio de Janeiro. Disponível em:

https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2010_2/bernardo/tls.htmlAcessoem: 17 de Outubro de 2017.

63 VIEIRA, Luiz. **ARP Poisoning**. Setembro de 2008. Disponível em: <https://imasters.com.br/artigo/10117/seguranca/arp-poisoning/?trace=1519021197&source=single>. Acesso em: 23 de Outubro de 2017.

64 VANHOEF, Mathy. **Key Reinstallation Attacks:**breaking WPA2 by forcing nonce reuse. 2017. Disponível em:<https://www.krackattacks.com>. Acesso em: 24 de Outubro de 2017.

65 PEREIRA, Ana Paula. **O que é XML?** 2009. Disponível em: <https://www.tecmundo.com.br/programacao/1762-o-que-e-xml-.htm>. Acesso em: 25 de Outubro de 2017.

66 W3Schools. **XML SOAP**. 2017. Disponível em:<https://www.w3schools.com/xml/xml_soap.asp>. Acesso em: 27 de Outubro de 2017.

67 W3C. **WEB SERVICES ADDRESSING 1.0- Core.** 2006. Disponível em: <https://www.w3org/TR/2006/REC-ws-addr-core-20060509/ >. Acesso em: 27 de Outubro de 2017.

68 IEEE. **WEP:** The “Wired Equivalent Privacy” Algoritm. 1994

69 Wi-Fi Alliance. **Wi-Fi Protect Access**: Strong, standards-based, interoperable security for today’s Wi-Fi networks. 2003.

70 Wi-Fi Alliance. **The State of Wi-Fi Security:** Wi-Fi Certified WPA2 Delivers Advanced Security to Homes, Enterprises and Mobile Devices. 2012. Disponível em: <https://www.wi-fi.org/downloads-registered-guest/20120229_State_of_Wi-

Fi_Security_09May2012_updated_cert.pdf/7600>. Acesso em: 27 de Outubro de 2017.

71 FIGUEIRA, Vitor Pinheiro. **"Internet das coisas":** um estudo sobre questões de segurança, privacidade e infraestrutura / Vitor Pinheiro Figueira. – Niterói, RJ: [s.n.], 2016.

72 UFRGS. 2016. **Engenharia social** (segurança da informação). Disponível em: < http://pt.wikipedia.org/w/index.php?oldid=19800887>. Acesso em: 27 de Outubro de 2017.

73 MICHIELINI Roziane. **Orientações para elaboração de trabalhos científicos:** Projeto de pesquisa, teses, dissertações, monografias e trabalhos acadêmicos, conforme a Associação Brasileira de Normas Técnicas (ABNT), a American Psychological Association (APA) e o Comitê Internacional de Editores de Revistas Médicas (VANCOUVER). 2015.