CENTRO DE INSTRUÇÃO DE GUERRA ELETRÔNICA

1º Ten Com MATHEUS DE LIMA FERREIRA

O EMPREGO DO RÁDIO DEFINIDO POR SOFTWARE (RDS) COMO UMA PLATAFORMA ADAPTATIVA PARA CAPTURA E ANÁLISE DE DADOS EM SISTEMAS DE COMUNICAÇÃO SEM FIO

Brasília
2019

1º Ten MATHEUS DE LIMA FERREIRA

O EMPREGO DO RÁDIO DEFINIDO POR SOFTWARE (RDS) COMO UMA PLATAFORMA ADAPTATIVA PARA CAPTURA E ANÁLISE DE DADOS EM SISTEMAS DE COMUNICAÇÃO SEM FIO

Trabalho de Conclusão do Curso de Guerra Cibernética para Oficiais apresentado ao Centro de Instrução de Guerra Eletrônica como requisito para obtenção do Grau de Pós-Graduação *Lato Sensu*, nível de especialização em Guerra Cibernética.

Orientador: 2º Sgt GUILHERME LIMA PINTO

Coorientador: 2º Sgt VINÍCIUS EMILIANO DOS SANTOS

Brasília
2019

Ficha Catalográfica Elaborada pela Biblioteca
do Centro de lnstrução de Guerra Eletrônica (CIGE)
Bibliotecária Responsável: 2° Ten Thaís Moraes CRB1/1922

M314e

Ferreira, Matheus de Lima
O Emprego do Rádio Definido por Software (RDS) como uma Plataforma Adaptativa para Captura e Análise de Dados em Sistemas de Comunicação sem Fio / Matheus de Lima Ferreira – Brasília, 2019.
61f.; il.

Trabalho de conclusão apresentado ao Curso Básico de Guerra Cibernética para Oficiais – Centro de Instrução de Guerra Eletrônica, Brasília, 2019.

Bibliografia: f. 55-57.

1. Rádio Definido por Software. 2. Bluetooth. 3. Telefonia Celular. 4. RF 433 MHz. 5. Captura de dados. 6. Análise de dados. 7.Metodologia. I. Ferreira, Matheus de Lima. II. Centro de Instrução de Guerra Eletrônica. III. O Emprego do Rádio Definido por Software (RDS) como uma Plataforma Adaptativa para Captura e Análise de Dados em Sistemas de Comunicação sem Fio.

CDD355

1º Ten MATHEUS DE LIMA FERREIRA

O EMPREGO DO RÁDIO DEFINIDO POR SOFTWARE (RDS) COMO UMA PLATAFORMA ADAPTATIVA PARA CAPTURA E ANÁLISE DE DADOS EM SISTEMAS DE COMUNICAÇÃO SEM FIO

Trabalho de Conclusão do Curso de Guerra Cibernética para Oficiais apresentado ao Centro de Instrução de Guerra Eletrônica como requisito para obtenção do Grau de Pós-Graduação *Lato Sensu*, nível de especialização em Guerra Cibernética.

Aprovado em: ___ de novembro de 2019

_____
Guilherme Lima Pinto – 2º Sgt
Orientador

_____
Vinícius Emiliano dos Santos – 2º Sgt
Co-orientador

_____
Jorge Henrique Corga Rodrigues Jardim – 1° Ten
Membro da Comissão de Avaliação

_____
Adão dos Santos – 1° Sgt
Membro da Comissão de Avaliação

Brasília
2019

A Léia Moreira, minha mãe, pelo exemplo de vida ímpar, motivando-me com amor infindável para cada desafio apresentado pela vida.

AGRADECIMENTOS

A Deus, autor e consumador de minha fé, razão do meu viver. Agradeço por sua presença durante toda a minha caminhada, ensinando como devo proceder a cada passo.

Aos meus pais, Nelson e Léia, responsáveis pela minha formação como homem e por quem hoje eu sou. Sou fruto de sua constante abnegação para que eu pudesse viver o melhor futuro que poderiam me oferecer.

Ao meu irmão, Nathan, pelo sempiterno companheirismo e camaradagem. Seu constante esforço e dedicação na conquista de seus objetivos são fonte de motivação diária para mim.

Aos meus instrutores e companheiros de turma, excelentes profissionais com os quais tive a honra de ombrear no breve período deste curso. Agradeço a dedicação e a persistência na transmissão dos conhecimentos.

Confia ao Senhor as tuas obras, e os teus planos serão estabelecidos.

Provérbios 16:3

RESUMO

Referência**:** FERREIRA, Matheus de Lima. **O emprego do Rádio Definido por SOFTWARE (RDS) como uma plataforma adaptativa para captura e análise de dados em sistemas de comunicação sem fio**. 2019. 61 folhas. Monografia (Curso de Guerra Cibernética para Oficiais) - Centro de Instrução de Guerra Eletrônica, Brasília, 2019.

A constante evolução do campo informacional tem levado a uma intensificação no uso de mecanismos tecnológicos para atividades cotidianas. Dada a demanda por dispositivos que acompanhem a mobilidade do usuário atual, é crescente o uso de aparelhos que utilizam redes sem fio para o seu funcionamento. Constatado este fato, o objetivo deste trabalho consistiu em comprovar a viabilidade de utilização do Rádio Definido por *Software* (RDS) como uma ferramenta robusta e de baixo custo para a captura e análise de dados de diferentes tecnologias *wirelles.* Após pesquisas e testes, foram colhidas informações a partir do tráfego de transmissões *Bluetooth,* telefonia móvel e de dispositivos RF 433 MHz, comprovando a hipótese inicial. Com os resultados colhidos, foi elaborada uma proposta de metodologia para o emprego genérico do RDS em diferentes protocolos. Apresentando-se como um dispositivo com flexibilidade para atuar em uma ampla faixa do espectro eletromagnético, conclui-se que o Rádio Definido por *Software* atua como ponto de interseção entre os campos de atuação da Guerra Eletrônica e da Guerra Cibernética.

Palavras-chave: Rádio Definido por *Software*. *Bluetooth*. Telefonia Celular. RF 433 MHz. Captura de dados. Análise de dados. Metodologia.

ABSTRACT

The constant evolution of the informational field has led to an intensification in the use of technological mechanisms for daily activities. Given the demand for devices that keep up with today's user mobility, there is a growing use of gadgets that use wireless networks to function. Given this fact, the objective of this work was to prove the feasibility of using Software Defined Radio (SDR) as a robust and low cost tool for data capture and analysis of different wireless technologies. After research and testing, information was collected from the traffic of Bluetooth transmissions, mobile telephony and RF 433 MHz devices, confirming the initial hypothesis. With the results collected, a methodology proposal was elaborated for the generic use of SDR in different protocols. Presenting itself as a device with flexibility to operate in a wide range of the electromagnetic spectrum, it can be concluded that Software Defined Radio acts as an intersection point between the fields of Electronic Warfare and Cyber Warfare.

Keywords: Software Defined Radio. Bluetooth. Mobile Telephony. RF 433 MHz. Data capture. Data analysis. Methodology.

LISTA DE ILUSTRAÇÕES

LISTA DE SIGLAS

RDS	Rádio Definido por *Software*
RF	Rádio-Frequência
FPGA	*Field Programmable Gate Array*
DSP	*Digital Signal Processor*
GPP	*General Purpose Processor*
SoC	*System on a Chip*
IF	*Intermediate Frequency*
CAD	Conversor Analógico-Digital
DDC	*Digital Down Converter*
PSD	Processador de Sinal Digital
CDA	Conversor Digital-Analógico
CPU	*Central Processing Unit*
TCP/IP	*Transport Control Protocol/Internet Protocol*
USB	*Universal Serial Bus*
ISM	*Industrial, Scientific, Medical*
FH-CDMA	*Frequency Hopping – Code Division Multiple Access*
FH-TDD	*Frequency Hopping – Time-Division Duplex*
SCO	*Synchronous Connection-Oriented*
ACL	*Asynchronous Connection-Less*
FHS	*Frequency Hopping Synchronization*
3GPP	*3rd Generation Partnership Project*
SMS	*Short Message Service*
GSM	*Global System for Mobile Communications*
MS	*Mobile Station*
BSS	*Base Station System*
NSS	*Network Switching System*

OMS	*Operations and Maintenance System*
ME	*Mobile Equipment*
SIM	*Subscriber Identity Mobile*
MAC	*Media Access Control*
IMEI	*International Mobile Equipment Identity*
HLR	*Home Location Register*
IMSI	*International Mobile Subscriber Identity*
UMTS	*Universal Mobile Telecommunications System*
UTRA	*Universal Terrestrial Radio Access*
WCDMA	*Wide-Band Code-Division Multiple Access*
LTE	*Long Term Evoluton*
EPS	*Evolved Packet System*
E-UTRAN	*Evolved Universal Terrestrial Radio Access Network*
EPC	*Evolved Packet Core*
HSPA	*High Speed Packet Access*
EDGE	*Enhanced Data GSM Environment*
LPD	*Low Power Device*
CEPT	*European Conference of Postal and Telecommunications Administrations*
Wi-FI	*Wirelles Fidelity*
BTLE	*Bluetooth Low Energy*
AFH	*Adaptative Frequency Hopping*
BLE	*Bluetooth Low Energy*
CRC	*Cyclic Redundancy Check*
MCC	*Mobile Country Code*
MNC	*Mobile Network Code*
MSIN	*Mobile Subscriber Indetification Number*
ERB	Estação Rádio-Base

DCS	*Digital Connection System*
AM	Amplitude Modulada
SSB	*Single Side-Band*
CW	*Continuos Wave*
AGC	*Automatic Gain Control*
CCCH	*Common Control Channel*
APOP	Agente Perturbador da Ordem Pública
NFC	*Near Field Communication*
GPS	*Global Positioning System*

SUMÁRIO

1 INTRODUÇÃO

O período do último quarto do século passado até os dias atuais possui como característica evidente a ascensão exponencial do campo da informação. É correta a afirmação de que:

> As principais conquistas tecnológicas do Século XX se deram no campo da aquisição, do processamento e da distribuição de informações. Entre outros desenvolvimentos, vimos a instalação das redes de telefonia em escala mundial, a invenção do rádio e da televisão, o nascimento e o crescimento sem precedentes da indústria de informática e o lançamento dos satélites de comunicação. (TANNEMBAUM, 2003, p. 18).

A facilidade de acesso a novas tecnologias cresce concomitantemente à consolidação e manutenção na engenharia dos *hardwares* e *softwares* empregados, o que leva, deste modo, à redução dos custos de aquisição destes produtos, seja para uso pessoal ou corporativo.

Em consequência disto, o ser humano tem passado cada vez mais a delegar desde suas tarefas mais básicas, como abrir um portão ou acender uma luz, até as mais avançadas, como diagnosticar uma doença utilizando *Machine Learning*, para sistemas computadorizados, os quais realizam, de modo geral, a coleta, o transporte, o armazenamento, e o processamento das informações recebidas.

O campo das comunicações é um dos principais beneficiados nesta frente de avanços tecnológicos. Diferentes inovações como a telefonia celular, as redes sem fio 802.11 e os protocolos utilizados em destravamento de automóveis são demonstrações de como a modernização se tornou essencial no contexto do século XXI.

Observa-se que as melhorias abordadas tratam-se de sistemas de comunicação sem fio, os quais utilizam-se do espaço aberto para a transmissão e recepção de dados, através da propagação de ondas eletromagnéticas sintonizadas em frequências, modulações e protocolos específicos.

É por esta característica de disponibilidade em *broadcast* que estes sistemas apresentam uma particularidade: toda a informação trafegada pode ser capturada, valendo-se de instrumentos e aplicações compatíveis com a onda interceptada.

1.1 DELIMITAÇÃO DO TEMA

O emprego do Rádio Definido por *Software* (RDS) como uma plataforma adaptativa para captura e análise de dados em sistemas de comunicação sem fio.

1.2 PROBLEMA

A diversidade de sistemas exige o uso de um variado número de equipamentos para se alcançar a liberdade de ação na análise do espectro, o que leva a um elevado custo na aquisição de materiais, e a um emprego maior em recursos humanos, resultando em perda na eficiência deste processo.

Baseado nestes fatos, como desenvolver uma metodologia para flexibilizar a alternância de escaneamento em diferentes tecnologias de comunicação sem fio, de forma a reduzir a quantidade de equipamentos e os custos necessários para a consecução das ações?

1.3 HIPÓTESE

O Rádio Definido por *Software* (RDS) pode ser empregado como uma estação capaz de capturar e analisar dados de diferentes tecnologias de redes sem fio.

1.4 JUSTIFICATIVA

Comprovada a potencialidade de emprego no âmbito da Guerra Cibernética, o Rádio Definido por *Software* (RDS) suprirá amplas demandas informacionais, apresentando-se como uma solução robusta e de baixo custo.

1.5 OBJETIVOS

A seguir têm-se os objetivos, geral e específicos, cujo presente trabalho se propôs atingir:

1.5.1 Objetivo Geral

O objetivo geral é apresentar o Rádio Definido por *Software* (RDS) como

solução multiplataforma para captura e análise de dados.

1.5.2 Objetivos Específicos

A fim de atingir o objetivo geral, os seguintes objetivos específicos serão buscados:

a) Apresentar o Rádio Definido por *Software* (RDS) como uma plataforma adaptativa para análise de diferentes tecnologias;

b) Realizar a captura e análise de dados em tecnologias *Bluetooth*, telefonia celular e em transmissões RF 433 MHz;

c) Comparar resultados e produzir os dados para a solução do problema; e

d) Apresentar uma proposta de metodologia para o emprego do Rádio Definido por *Software* (RDS) no escaneamento de tecnologias de comunicação sem fio distintas.

1.6 MÉTODO DE PESQUISA

Quanto aos objetivos, foi realizada uma pesquisa exploratória, através de pesquisas bibliográficas.

Quanto à abordagem, foi realizada uma pesquisa qualitativa dos dados apresentados pelas bibliografias referenciadas.

Quanto à finalidade, baseou-se em pesquisa aplicada, buscando gerar conhecimento para a aplicação prática e dirigida à solução dos problemas definidos anteriormente nos objetivos deste trabalho.

Quanto ao método, empregou-se o hipotético-dedutivo a fim de testar uma possível solução para o problema apresentado.

Por fim, quanto aos procedimentos, utilizaram-se pesquisas bibliográficas, documentais e experimentais.

1.7 ESTRUTURA DO TRABALHO

No primeiro capítulo serão abordados os tópicos referentes à introdução, à delimitação do tema, ao problema de pesquisa, à hipótese, aos objetivos gerais e

específicos, bem como aos métodos de pesquisa e à estrutura do trabalho.

No segundo capítulo serão descritos os principais conceitos e definições de assuntos atinentes ao Rádio Definido por *Software* – RDS.

No terceiro capítulo serão descritos os principais conceitos e definições de assuntos atinentes às tecnologias de comunicação sem fio.

No quarto capítulo serão descritas as técnicas, táticas e procedimentos empregadas para a captura dos dados.

No quinto capítulo será apresentada a análise dos resultados e uma proposta de metodologia para o emprego do Rádio Definido por *Software* – RDS como uma multiplataforma de escaneamento de redes.

No sexto capítulo será feita a conclusão do presente trabalho.

2 RÁDIO DEFINIDO POR SOFTWARE

Segundo [*Wirelles Innovation Forum*], um rádio é qualquer tipo de dispositivo que transmite ou recebe sinais em alguma radiofrequência (RF) do espectro eletromagnético, permitindo a transferência facilitada de informações. Atualmente, rádios existem em uma variedade de equipamentos como celulares, computadores, chaves de carro, veículos e televisões.

Os equipamentos rádio tradicionais baseados em *hardware* possuem funcionalidades para interoperação limitadas e somente podem ser modificados através de intervenção física. Isto resulta em altos custos de produção e mínima flexibilidade no suporte a diferentes padrões de forma de onda. Por outro lado, o Rádio Definido por Software (RDS) oferece uma solução eficiente e econômica a este problema, permitindo o aumento de suas capacidades nativas através de atualizações de *software* [*Wirelles Innovation Forum*].

Conforme se explica em [*Wirelles Innovation Forum]:*

> O RDS define um conjunto de tecnologias de *hardware* e *software* onde algumas ou todas as funções operativas do rádio (inclusive o processamento de camada física) são implementadas através de *software* modificável ou alteração de *firmware* em tecnologias de processamento programáveis. Estes dispositivos incluem arranjo de portas programáveis em campo (FPGA), processadores de sinais digitais (DSP), processadores de uso geral (GPP), sistema em um chip (SoC) ou outra aplicação específica de processadores programáveis. A utilização destas tecnologias permite novos recursos sem fio e capacidades a serem agregadas a um sistema de rádio existente sem a necessidade de novo *hardware*.

Segundo Dillinger et al. (2003), as Forças Armadas americanas foram as primeiras instituições a empregar o RDS. Cada órgão governamental realiza a compra de seus sistemas de forma independente, o que leva a uma multiplicidade de tecnologias frequentemente não compatíveis entre si, devido a fatores como frequência utilizada, tecnologia na modulação dos sinais, dentre outros. Assim, os militares dos EUA, empregaram o RDS com o objetivo de garantir a interoperabilidade dos seus equipamentos com o de outras agências governamentais.

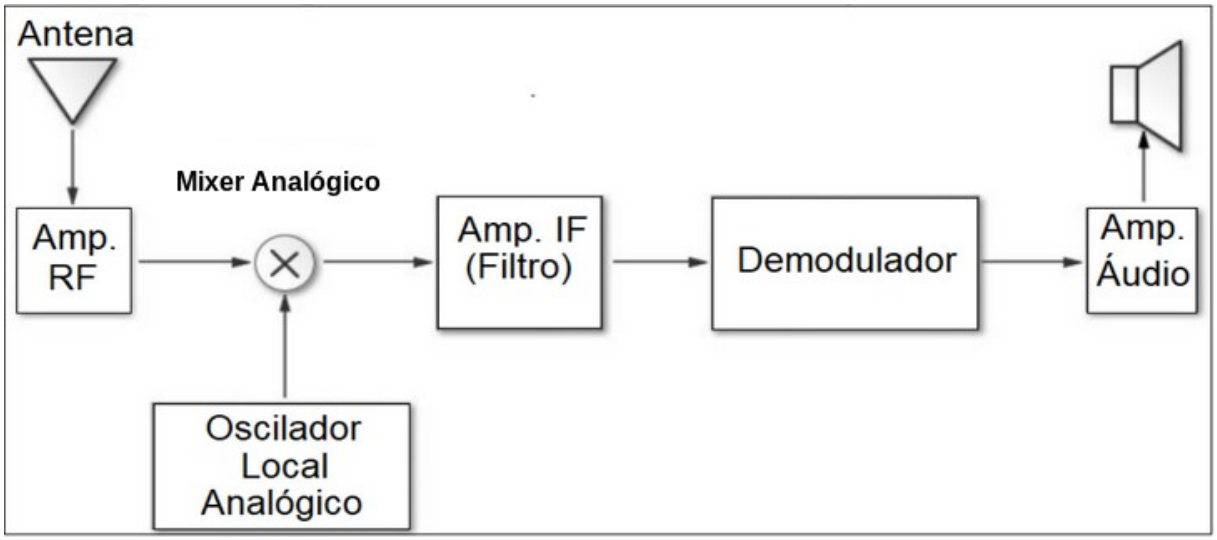
2.1 ARQUITETURA DO RÁDIO DEFINIDO POR *SOFTWARE*

Segundo Machado-Fernández (2015), em um rádio receptor convencional, o

sinal de radiofrequência (RF) recebido pela antena é repassado para um amplificador. Este só atua dentro de sua finalidade para a sua faixa de frequência de operação. Este sinal recebido e aumentado é então mesclado a um outro sinal gerado por um oscilador digital analógico, que gera uma onda de acordo com a sintonização do rádio. A união dos dois sinais é realizada por um *mixer* analógico, que gera a frequência intermediária de interesse. Após, um amplificador de IF que, dentro de suas especificações, atua como um filtro de passagem de banda, delimita a largura que será repassada ao demodulador para processamento.

O demodulador, recebendo o sinal, irá recuperar o sinal modulado original. Deste ponto em diante, diversos processamentos podem ser realizados de acordo com a necessidade e a finalidade do equipamento rádio em utilização. Na figura 1, é apresentado o esquema de um rádio receptor convencional, onde a onda em RF é recebida e transformada ao final do encadeamento em audiofrequência (MACHADO-FERNÁNDEZ, 2015).

Figura 1 – Diagrama de blocos básico de um receptor convencional



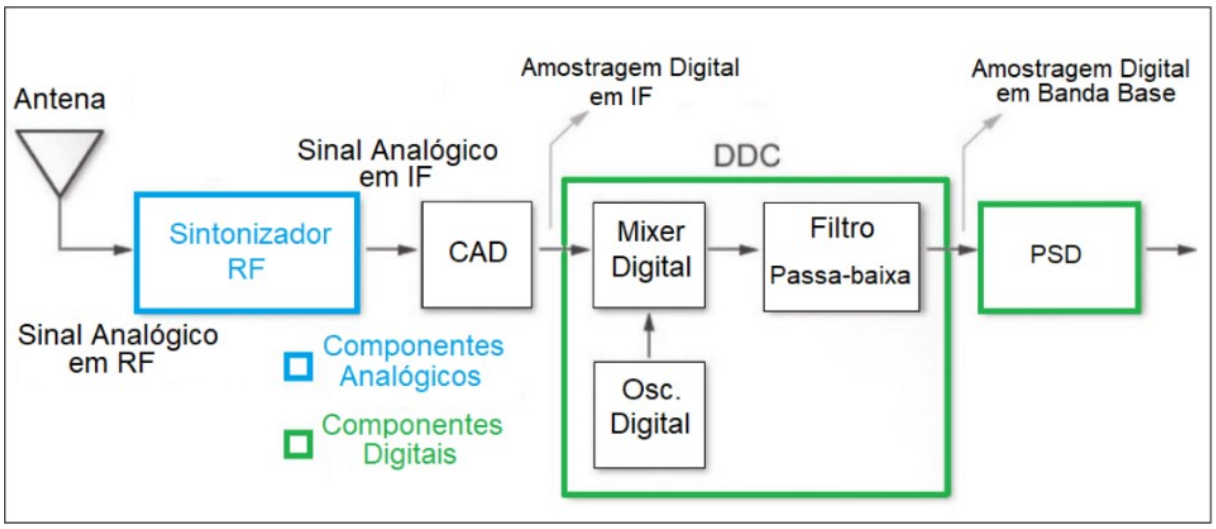
FONTE: Machado-Fernández (2015), p. 83, tradução nossa

Os diagramas de blocos de outros equipamentos rádio, transceptores ou não, possuem características semelhantes, integrando filtros, osciladores, (de)moduladores, amplificadores, dentre outros. Em suas implementações, as capacidades intrínsecas ao seu emprego são limitadas pelo *hardware* implantado no processo de fabricação (MACHADO-FERNÁNDEZ, 2015).

No caso dos rádios definidos por *software*, os componentes supracitados também estão presentes, entretanto de forma flexível e modular, pois são implantados de forma lógica. A figura 2 apresenta o diagrama de blocos de um RDS

receptor, em contraste ao de um rádio receptor comum:

Figura 2 – Diagrama de blocos básico de um RDS receptor



FONTE: Machado-Fernández (2015), p. 84, tradução nossa

No RDS, o sinal recebido passa pelo sintonizador RF, que equivale aos três primeiros blocos do diagrama anterior, sendo enviado, em sua saída, para o conversor analógico-digital (CAD). A saída gerada, em formato digital, é enviada ao Conversor Digital Baixo (DDC) (MACHADO-FERNÁNDEZ, 2015).

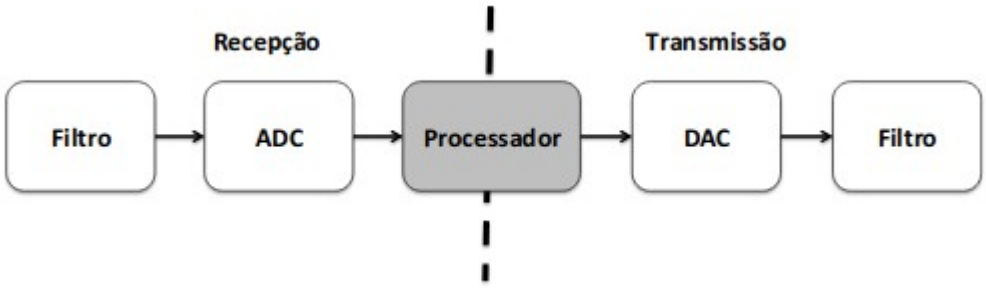
Conforme diz Machado-Fernández (2015), o DDC é um chip monolítico, cuja função é primordial para o funcionamento do RDS. Em sua estrutura, ele possui um *mixer* digital, um filtro passa-baixa e um oscilador digital. Estes componentes são implementados de maneira semelhante ao do rádio receptor comum, entretanto suas características e capacidades são definidas em *software*. Após passar pelo DDC, o sinal é repassado ao Processador de Sinal Digital (PSD), que fará a demodulação e decodificação do sinal, de acordo com a finalidade do rádio receptor, sendo suas funcionalidades programadas por *software*.

Nos exemplos supracitados, caracterizou-se o aspecto de recepção a fim de exemplificar a flexibilidade oferecida por um sistema que possui a viabilidade de implementar funcionalidades via *software*. Entretanto, dentro do escopo da atividade cibernética, a recepção fica limitada à proteção e à exploração. Para o exercício do ataque, é importante que a capacidade de transmissão seja também oportunizada.

De forma simplificada, a Figura 3 mostra a arquitetura básica de um RDS que opera tanto como um transmissor ou receptor. Os conversores analógico-digital, digital-analógico (ADC e DAC) e os filtros de frequência devem ser implementados em hardware (bloco de cor branca), enquanto as outras funcionalidades podem ser

inseridas no transceptor através de unidades de processamento programáveis (bloco cinza escuro). Dependendo da finalidade e da onda recebida, o sinal de banda base é digital, de forma que os blocos de DAC, ADC e filtros não precisem ser instalados, direcionando-se os sinais digitais diretamente para a unidade de processamento.

Figura 3 – Arquitetura simplificada de um Rádio Definido por Software (Transceptor)



FONTE: SILVA, W. *et al*., 2015

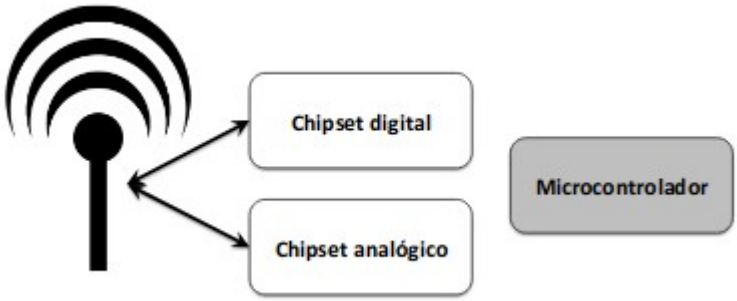
Conforme Silva, W. *et al* (2015), existem duas arquiteturas principais para a implementação do RDS. A forma mais simples é a disposição de vários *chipsets* configurados para diferentes padrões de comunicação em um mesmo rádio onde, de acordo com a necessidade, alterna-se entre eles. Outras formas de emprego consistem na configuração de *hardwares* especializados. O RDS Modal e o Reconfigurável são as duas constituições mais utilizadas.

2.1.1 RDS Modal

Segundo Silva, W. *et al* (2015), arquiteturas de RDS Modal funcionam como um rádio com N implementações, que são alternadas sob demanda. Esta arquitetura surgiu na década de 1990, quando as tecnologias celulares analógicas foram gradualmente substituídas por redes digitais. Nesta situação, eram necessários telefones que pudessem oferecer serviços digitais a maior parte do tempo, mas estes deveriam suportar o modo analógico em áreas onde a cobertura digital ainda não existia. A solução óbvia era combinar os *chipsets* analógicos e digitais no interior do aparelho, com *software* realizando o chaveamento entre eles.

Na Figura 4, observa-se o exemplo de um rádio modal, com duas tecnologias diferentes de comunicação implementadas. A cor cinza indica componentes definidos em *software* e a branca indica componentes de *hardware*.

Figura 4 – Exemplo de um RDS modal



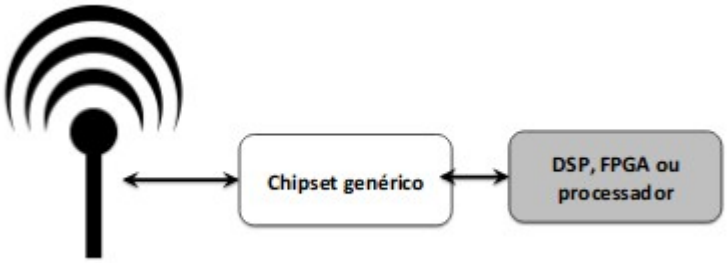
FONTE: Silva, W. *et al*., 2015

2.1.2 RDS Reconfigurável

Conforme Silva, W. *et al* (2015), existem aplicações em que o RDS modal é uma solução inadequada, por exemplo na pesquisa experimental. Além disso, sistemas de RDS Modal não escalam bem com a quantidade de formas de onda a serem suportadas. Nestas situações, é mais interessante empregar *hardware* programável para fazer o processamento dos sinais. Desta forma, o *hardware* pode suportar uma quantidade ilimitada de padrões e técnicas de transmissão. Esta abordagem é frequentemente denominada RDS reconfigurável, porque o *hardware* pode ser configurado para qualquer aplicação.

Na Figura 5, o componente em cinza pode ser uma CPU, um DSP ou um FPGA, o qual é programável para que se cumpra a tarefa desejada.

Figura 5 – Exemplo de um RDS Reconfigurável



FONTE: Silva, W. *et al*., 2015

2.2 FATORES DE EMPREGO DO RÁDIO DEFINIDO POR SOFTWARE

Derivada da crescente evolução no cenário tecnológico, especificamente no campo das comunicações sem fio, o RDS vem a suprir diversas demandas de forma

flexível e a baixo custo, em contraste aos sistemas tradicionais (SILVA, W. *et al,* 2015).

Todo ativo tecnológico (*hardware, software,* protocolos etc) possui características intrínsecas que devem ser avaliadas a fim de definir se sua utilização é válida ou não, dependendo do contexto e finalidade do objetivo desejado. Esta sentença é válida da mesma forma para o RDS. Assim, serão descritos abaixo os coeficientes que devem ser observados no emprego desta ferramenta (SILVA, W. *et al,* 2015).

2.2.1 Fatores Positivos

Como primeiro fator, Silva, W. *et al* (2015) considera que o RDS é um dispositivo evolutivo. Suas funcionalidades são implementadas por *software* e, por isso, pode receber novas capacidades e correções de *bugs* mesmo após a sua distribuição e entrada em operação. Se um novo padrão for liberado (Ex.: evolução da tecnologia celular 4G para 5G) o mesmo dispositivo pode ser atualizado. É de se considerar que uma variedade de dispositivos atuais já possuem esta possibilidade nativa por utilizarem *firmwares*, porém a camada física dos mesmos ainda é implementada por hardware.

O reuso de *hardware* é o segundo fator fundamental no fomento ao emprego do rádio definido por *software*. Para uma mesma tecnologia, como a telefonia celular por exemplo, existem inúmeros padrões, cuja variedade se deve a nuances como localização geográfica e regulamentação específica de cada Estado-nação. O mesmo caso se aplica a padrões de TV digital. Com o RDS este fato não se torna um impeditivo, tornando um equipamento capaz de atender a um amplo campo de possibilidades (SILVA, W. *et al,* 2015).

Para o terceiro fator, os protocolos de comunicação são cientes da aplicação. Redes sem fio comumente são utilizadas para um propósito muito específico, de forma que são interligadas com redes tradicionais por nós ou pontes. Conforme diz Silva, W. *et al* (2015):

> A comunicação TCP/IP, padrão nas redes, opera em um sistema de caixa preta. O uso de abstrações como mensagens, circuitos ou pacotes limita o conjunto de ações possíveis para operações básicas, como a filtragem por endereço e porta, estabelecendo parâmetros de QoS etc. Como a implementação tem de suportar fluxos genéricos, o equipamento deve utilizar soluções muito abrangentes, e desta forma

> não é possível empregar otimizações específicas de cada aplicação. (…) Assim, o trecho que é utilizado para um propósito específico poderia empregar protocolos otimizados para a aplicação, por exemplo utilizando os conceitos de redes centradas em dados ou identificadores geolocalizados.

Por último, o RDS oferece níveis mais altos de abstração. Com isso, compiladores e sistemas operacionais podem implementar otimizações e verificações mais complexas, de forma que é gerado um código mais eficaz, com tempo de desenvolvimento reduzido (SILVA, W. *et al,* 2015).

2.2.2 Fatores Negativos

Silva, W. *et al* (2015) afirma que o desempenho inferior é caracterizado como primeira desvantagem. Sistemas baseados em configuração de *hardware* apresentam desempenho mais veloz, dado que sua infraestrutura é desenvolvida de acordo com a finalidade e com a aplicação em execução, se for o caso. Um sistema que baseia-se em *software* faz proveito de componentes genéricos, a fim de prover a compatibilidade com diversos padrões.

O emprego de RDS também tem como revés o uso de energia superior e uma maior área de circuito, dado que uma conformação programável exige uma maior quantidade de portas lógicas em detrimento de um sistema baseado em *hardware*. Para projetos onde o tamanho dos componentes e o consumo de energia são fatores relevantes, esta característica pode apresentar-se como um fator de limitação (SILVA, W. *et al,* 2015).

Um fator muito importante a ser considerado, segundo Silva, W. *et al* (2015), é a segurança. A despeito da menor flexibilidade e interoperabilidade dos sistemas implementados em *hardware*, para que se faça qualquer alteração no equipamento, é necessário acesso físico ao mesmo e, dependendo do caso, uma vasto arcabouço de conhecimento técnico para a ação. Sendo baseado em *software*, inclusive configurações de baixas camadas de comunicação podem ser modificadas sem a necessidade de acesso físico.

2.2.3 Avaliação dos Fatores

No contexto deste trabalho, o Rádio Definido por *Software* (RDS) foi

empregado com a finalidade de exploração, a partir da coleta e análise de dados, e eventualmente empregado para o ataque.

Dado este fato, o equipamento em questão foi utilizado de forma distinta da sua finalidade comum, não sendo instalado como parte vital de uma infraestrutura de comunicações ou para processamento de grande quantidade de dados.

O reuso de *hardware* e sua capacidade evolutiva são pontos importantes em se tratando do seu aproveitamento no escaneamento de diferentes tecnologias que utilizam o espectro eletromagnético.

Quanto aos fatores negativos, fato é que os mesmos se aplicam para a atividade em questão. O RDS é capaz de cumprir seu propósito, mas está sujeito às limitações de *hardware* genérico, maior área de circuito e questões inerentes à segurança. Todavia, os ganhos adquiridos com a flexibilidade e a economia nos custos de aquisição compensam as condicionantes negativas, visto o elevado valor de manutenção de contratos ao adquirir tecnologias proprietárias.

O RDS é uma solução de *hardware* e de fonte abertas, de forma que o emprego, manutenção e evolução do sistema pode ser realizado pela própria instituição que o emprega.

2.3 HACKRF ONE

A gama de opções de Rádios Definidos por *Software* disponíveis no mercado é bastante ampla. No Quadro 1 é feita uma amostra dos modelos mais empregados:

Quadro 1 – RDS mais comuns

RDS	Frequência Minima (MHZ)	Frequência Máxima (MHZ)	Largura de Banda RX (MHz)	Resolução do CAD (Bits)	Transmite? S/N	Preço ($USD)
RTL-SDR (R820T)	24	1766	3.2	8	Não	~20
Funcube Pro+	0.15 410	260 2050	0.192	16	Não	~200
Airspy	24	1800	10	12	Não	199
SDRPlay	0.1	2000	8	12	Não	149
HackRF	30	6000	20	8	Sim	299
BladeRF	300	3800	40	12	Sim	400 & 650
USRP 1	DC	6000	64	12	Sim	700

FONTE: Teixeira (2016)

Considerando-se que o RDS a ser utilizado deve ser capaz de explorar e eventualmente realizar ataques, aspectos técnicos devem ser considerados para a escolha do equipamento empregado:

a) Tecnologias de comunicação sem fio utilizam diferentes frequências. O RDS que se destina a ser flexível para coletar informações deve ser capaz de escanear uma ampla faixa do espectro eletromagnético;

b) A fim de realizar o monitoramento de transmissões, o RDS deve ser capaz de escanear simultaneamente uma largura de banda considerável; e

c) Para replicar sinais capturados, realizar ataques de *jammings,* interferência eletromagnética, dentre outros, o RDS precisa ter a capacidade de atuar não somente como receptor, mas também como transceptor; e

d) O RDS deve ser uma solução de baixo custo correspondente às suas potencialidades.

Avaliados os equipamentos à luz das condicionantes supracitadas, o Rádio Definido por *Software* HackRF One foi escolhido como solução para as atividades de captura e análise a serem desenvolvidas. Na figura 6, temos o modelo de RDS escolhido:

Figura 6 – HackRF One

FONTE: Great Scott Gadgets (2016)

O HackRF One apresenta como características operacionais [*Great Scott Gadgets*]:

- *Hardware Open-Source;*
- Transceptor *Half-Duplex*;

- Alimentação via USB;
- Opera na faixa de 1 MHz a 6 GHz;
- Ganho de transmissão, recepção e filtro de banda-base configuráveis via *software*;
- Potência de saída da antena controlada via *software* (50mA a 3,3V);
- Até 20 milhões de *samples* por segundo; e
- Compatível com GNU Radio.

3 TECNOLOGIAS SEM FIO

Dentro do escopo deste trabalho, será explorada a captura e análise de informações das seguintes tecnologias sem fio através do HackRF One:

a) *Bluetooth*;

b) Redes Celulares; e

c) *Low Power Device* 433 MHz.

O objetivo do presente capítulo é apresentar os padrões e parâmetros de cada tecnologia a ser explorada.

3.1 *BLUETOOTH*

Segundo Alecrim (2011), *Bluetooth* é um padrão global de comunicação sem fio e de baixo consumo de energia que permite a transmissão de dados entre dispositivos compatíveis com a tecnologia. Para isso, uma combinação de *hardware* e *software* é utilizada para permitir que essa comunicação ocorra entre os mais diferentes tipos de aparelhos. A transmissão de dados é feita através de radiofrequência, permitindo que um dispositivo detecte o outro independente de suas posições, desde que estejam dentro do limite de proximidade

Dentro de suas especificações, a tecnologia foi dividida em três categorias (ALECRIM, 2011):

- Classe 1: potência máxima de 100 mW, alcance de até 100 metros;
- Classe 2: potência máxima de 2,5 mW, alcance de até 10 metros; e
- Classe 3: potência máxima de 1 mW, alcance de até 1 metro.

A partir da existência de diferentes padrões de comunicação distribuídos no mundo, o *Bluetooth* desde sua criação foi desenvolvido para ser funcional

independente da localização geográfica. Para isso, a faixa ISM (*Industrial, Scientific, Medical*) é a que corresponde à necessidade apresentada, operando de 2,4 GHz a 2,5 GHz (ALECRIM, 2011).

Tratando-se de uma faixa aberta, qualquer sistema de comunicação existente pode se valer da mesma para estabelecer comunicação para diversos fins, conforme Alecrim (2011). Para evitar a geração ou o sofrimento de interferência, é utilizado o esquema de comunicação FH-CDMA (*Frequency Hopping-Code-Division Multiple Access*). Este processo divide a faixa de frequência utilizada em diversos canais. Estabelecida a conexão, a transferência de dados é realizada através de salto de frequência (*frequency hopping*). Tal configuração leva à redução da largura de banda utilizada e, consequentemente, a uma taxa de transferência menor.

Desta forma, podem ser utilizadas até 79 frequências, espaçadas a 1 MHz entre si. Utilizando a tecnologia FH/TDD (*Frequency Hopping/Time-Division Duplex*), a comunicação ocorre no modo *full-duplex*, onde os integrantes de uma rede pode transmitir e receber simultaneamente. Para que isto ocorra, as transmissões são divididas em *slots*, os quais são canais divididos em períodos de 625 µs (microssegundos). Sendo que cada salto de frequência corresponde a um *slot,* em um segundo ocorrem 1600 saltos (ALECRIM, 2011).

Existem dois padrões para o estabelecimento do enlace entre transmissores e receptores, de acordo com Alecrim (2011):

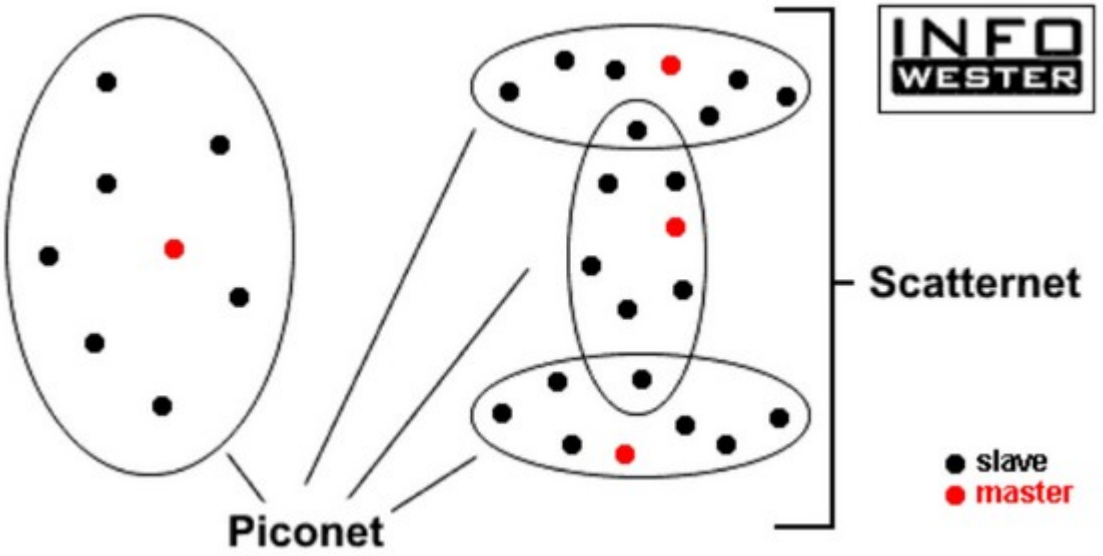
a) *Synchronous Connection-Oriented* (SCO): O dispositivo mestre e o dispositivo escravo estabelecem um *link* sincronizado entre si, reservando-se *slots* para cada um. Sua principal utilização é em aplicações com envio contínuo de dados, como conversa de voz, por exemplo. Por outro lado, não há o reenvio de dados perdidos. Sua taxa de transmissão é de 432 Kb/s. Para voz, a taxa é de 64 Kb/s.

b) *Asynchronous Connection-Less* (ACL): É estabelecido um *link* assíncrono entre o dispositivo mestre e os dispositivos escravos, sem se valer de reserva de *slots*. Neste caso, ocorre a recuperação de dados perdidos. Esta funcionalidade é importante em se tratando de transferência de arquivos, por exemplo. A taxa de transferência de dados neste modo vai até 721 Kb/s.

A tecnologia *Bluetooth* também funciona conectada em rede. Esta última, no contexto deste padrão, denomina-se *piconet,* constituída pelo *master* e pelos escravos. O dispositivo que inicia a conexão assume o primeiro papel, e os demais,

o segundo. Uma *piconet* suporta 8 dispositivos, porém, sobrepondo-se mais desta de forma que uma se comunique com a outra dentro de um limite de alcance, tem-se uma *scatternet,* exemplificada na Figura 7.

Figura 7 – Esquema de uma *Scatternet*



FONTE: Alecrim (2011)

Dispositivos que desejam participar de uma *piconet* já existente enviam um sinal denominado *Inquiry*. Ao receberem esse sinal, os demais integrantes respondem com um pacote FHS (*Frequency Hopping Synchronization*), onde informam sua identificação e dados de sincronismo. A partir deste momento, o novo dispositivo pode iniciar a sua comunicação (ALECRIM, 2011).

Caracterizado o comportamento da tecnologia *Bluetooth* na transmissão e recepção de dados a partir da utilização do espectro eletromagnético, seu tráfego pode ser capturado, o que será objeto de estudo do próximo capítulo.

3.2 REDES CELULARES

A segunda tecnologia de interesse para este trabalho são as redes de telefonia celular dos tempos atuais. Com a criação do 3GPP (SVERZUT, 2015), ocorreu o estabelecimento de padrões internacionais que vieram a disponibilizar permanentemente a interoperabilidade entre as tecnologias existentes. A primeira geração a ocorrer a transmissão de voz e dados foi a 2G, com o advento do *Short Message Service* (SMS).

3.2.1 Tecnologia 2G

Segundo Sverzut (2015), o padrão *Global System for Mobile Communications* (GSM) correspondeu à tecnologia de segunda geração da telefonia celular. É um sistema formado por interfaces abertas e padronizadas, onde operadoras puderam combinar equipamentos de diversos fabricantes.

A arquitetura da rede GSM, exemplificada na Figura 8, é formada por quatro componentes:

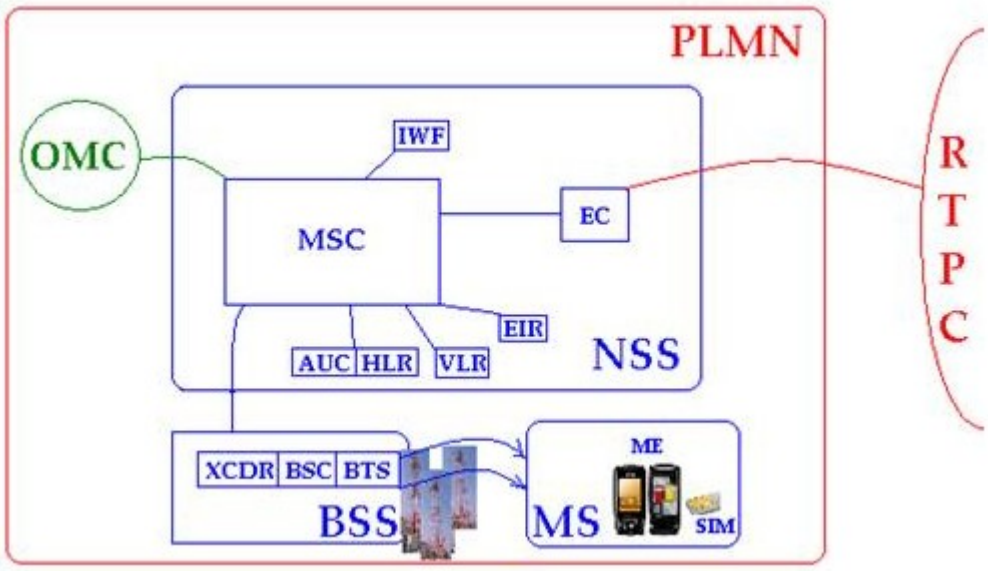
a) Estação Móvel (*Mobile Station* - MS): responsável por conectar o usuário à rede GSM;

b) Sistema de Estação Base (*Base Station System* - BSS): conecta em radiofrequência a estação móvel ao sistema de comutação;

c) Sistema de Comutação de Rede (*Network Switching System* - NSS): interconecta a rede GSM com a rede pública; e

d) Sistema de operação e manutenção (*Operations and Maintenance System - OMS*): opera e faz a manutenção dos grupos componentes.

Figura 8 – Arquitetura da Rede GSM



FONTE: Santos (2008)

Segundo Sverzut (2015), a estação móvel é o aparelho celular propriamente dito, de porte do usuário comum. Ele é constituído pelo equipamento móvel (*Mobile Equipment* – ME) e pelo módulo de identidade do assinante (*Subscriber Identity Mobile* – SIM), em algumas tecnologias o popular “chip”. Cada ME possui um número único que o identifica, de forma equivalente ao endereço MAC nas placas de

rede. Este número é a identidade internacional de equipamento móvel (*International Mobile Equipment Identity* – IMEI).

Cabe destacar ainda, dentro do NSS, a existência de um registro de localização local (*Home Location Register* – HLR). Este realiza o controle da base de dados dos assinantes locais. Neste registro, há o tráfego da identidade internacional de assinante móvel (*International Mobile Subscriber Identity* – IMSI), a qual será objeto de exploração no próximo capítulo. Com esta informação, pode-se realizar diversas ações de informação para alimentar dados de inteligência, por exemplo (SVERZUT, 2015).

3.2.2 Tecnologia 3G

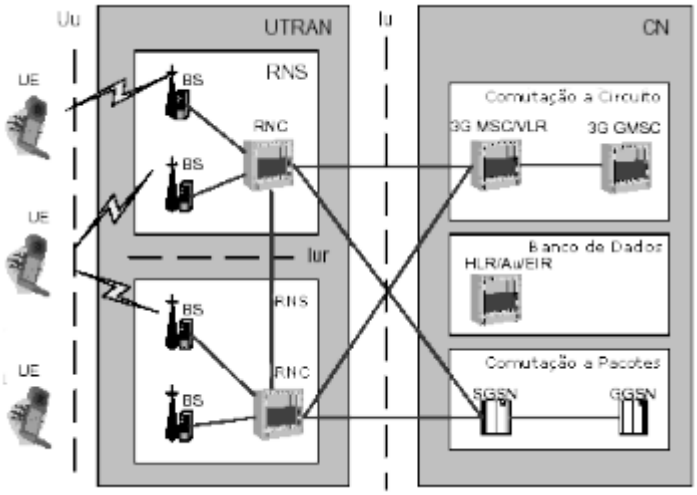
Conforme Sverzut (2015), o motor para a evolução tecnológica dos sistemas celulares é a necessidade de se agregar novos serviços à telefonia móvel. Com a tecnologia 2G, já era possível enviar mensagens de texto, mensagens de e-mail e receber atualizações sobre tempo, clima, esportes, dentre outros.

Entretanto, com a exponencial evolução da Internet, a demanda pelo acesso chegou também aos dispositivos celulares. Mesmo com um computador portátil, fazia-se necessário um ponto de acesso à rede. O 3G supriu a demanda desejada, permitindo que em qualquer lugar com cobertura de telefonia móvel pudesse ser realizado o acesso à Internet (SVERZUT, 2015).

Denominado *Universal Mobile Telecommunications System* (UMTS), é fruto da união entre os padrões UTRA (*Universal Terrestrial Radio Access*) e o WCDMA (*Wide-Band Code-Division Multiple Access*). Mantendo a compatibilidade com a geração anterior, realizou um aprimoramento na capacidade de transmissão de dados e nos serviços oferecidos, permitindo a melhoria de aplicações que utilizam serviços de rede e Internet, como envio de imagens e videoconferência (SVERZUT, 2015).

A arquitetura do UMTS, demonstrada na Figura 9, é constituída pela união de três sub-arquiteturas, as quais são o equipamento do usuário, a rede universal de acesso de RF terrestre (*Universal Terrestrial Radio Access Network*) e a rede de suporte. Esta última é implementada da mesma forma que o 2G, permitindo a migração e a compatibilidade para as tecnologias anteriores (SVERZUT, 2015).

Figura 9 – Arquitetura celular UMTS (3G)



FONTE: Teleco (2004)

3.2.3 Tecnologia 4G

Segundo afirma Sverzut (2015), a proposta de aumentar ainda mais a taxa de transmissão de dados veio a ser atendida com o padrão de quarta geração denominado *Long Term Evolution* (LTE). Sua arquitetura, também chamada de EPS (*Evolved Packet System*), é dividida em duas redes (SVERZUT, 2015):

a) E-UTRAN (*Evolved Universal Terrestrial Radio Access Network)*: rede de acesso sem fio; e

b) EPC (Evolved Packet Core): rede de suporte ou núcleo de rede, sendo baseado na arquitetura IP com suporte a interoperabilidade entre as redes GSM e WCDMA/HSPA.

O 4G tornou-se necessário a partir da evolução dos serviços de rede, que passam a necessitar de um tempo de latência e de taxas de download cada vez menores (SVERZUT, 2015). O quadro 2 exemplifica esta necessidade.

Quadro 2 – Velocidades de *Download* necessárias para serviços

Activity	Required Download Speed
Skype/WhatsApp phone call	0.1Mbit/s
Skype video call	0.5Mbit/s
Skype video call (HD)	1.5Mbit/s
Listening to online radio	0.2Mbit/s
Watching YouTube videos (basic quality)	0.5Mbit/s
Watching YouTube videos (720p HD quality)	2.5Mbit/s
Watching YouTube videos (1080p HD quality)	4Mbit/s
Watching iPlayer/Netflix (standard definition)	1.5Mbit/s
Watching iPlayer/Netflix (high definition)	5Mbit/s
Watching iPlayer/Netflix (4K UHD)	25Mbit/s

FONTE: Lo (2019)

Com o advento do 4G, houve um incremento significativo nas velocidades, conforme o quadro 3.

Quadro 3 – Comparação das velocidades de *Download* para as diferentes tecnologias

Activity	4G Download Time	3G Download Time	2G Download Time
Accessing typical web page	0.5 seconds	4 seconds	3 minutes
Sending an e-mail without attachments	<0.1 seconds	<0.1 seconds	1 second
Downloading high-quality photograph	0.5 seconds	4 seconds	3 minutes
Downloading an music track (MP3)	3 seconds	10 seconds	7 minutes
Downloading an application	8 seconds	1 minute	40 minutes

FONTE: Lo (2019)

Para este quadro, foram consideradas as velocidades médias de *download* de 30 Mbit/s (4G LTE Cat6), 4Mbit/s (3G HSPA+) e 0,1 Mbit/s (2G EDGE). Os tamanhos de arquivo considerados foram: 2 MB para a página *Web,* 10 KB para um e-mail, 2 MB para uma fotografia de alta qualidade, 5 MB para uma faixa de música e 30 MB para uma aplicação (LO, 2019).

3.3 LOW POWER DEVICE 433 MHz

Segundo Wikipedia (2019), LPD433 (*low power device 433 MHz*) é uma

banda de UHF onde dispositivos sem licença são permitidos a operar em algumas regiões. As frequências correspondem à banda ISM da região ITU 1 de 433,050 MHz a 434,790 MHz, com operação limitada aos países da CEPT. As frequências utilizadas estão dentro da banda 70-centímetros, a qual é atualmente reservada para o governo e para atividades de radioamadorismo nos Estados Unidos e na maioria das nações ao redor do mundo.

LPD também é utilizado em dispositivos *key-less* para veículos, garagens, portões automáticos e estações meterorológicas caseiras (Wikipedia, 2019), sendo esta destinação a que será objeto de estudo deste trabalho. A figura 10 exemplifica uma utilização desta tecnologia.

Figura 10 – Controle de abertura de veículo com RF *Wirelles*

FONTE: Reed (2018)

4 CAPTURA DE DADOS

Sendo o objetivo deste trabalho apresentar o Rádio Definido por *Software* (RDS) como uma ferramenta para o escaneamento de diversas tecnologias sem fio, neste capítulo serão abordados táticas, técnicas e procedimentos que demonstram o potencial de utilização do equipamento em questão.

Para tal, foi utilizado o *HackRF ONE*, apresentado no capítulo 2 do presente trabalho, com o *firmware* versão 2018.01.1 (API:1.02). Juntamente ao RDS, foi utilizado um computador portátil com o sistema operacional instalado Ubuntu 19.04 kernel 5.0.0-31-generic, virtualizando uma máquina com Kali GNU/Linux Rolling

kernel 5.2.0-kali2-amd64 com o programa VirtualBox versão 6.0.6.

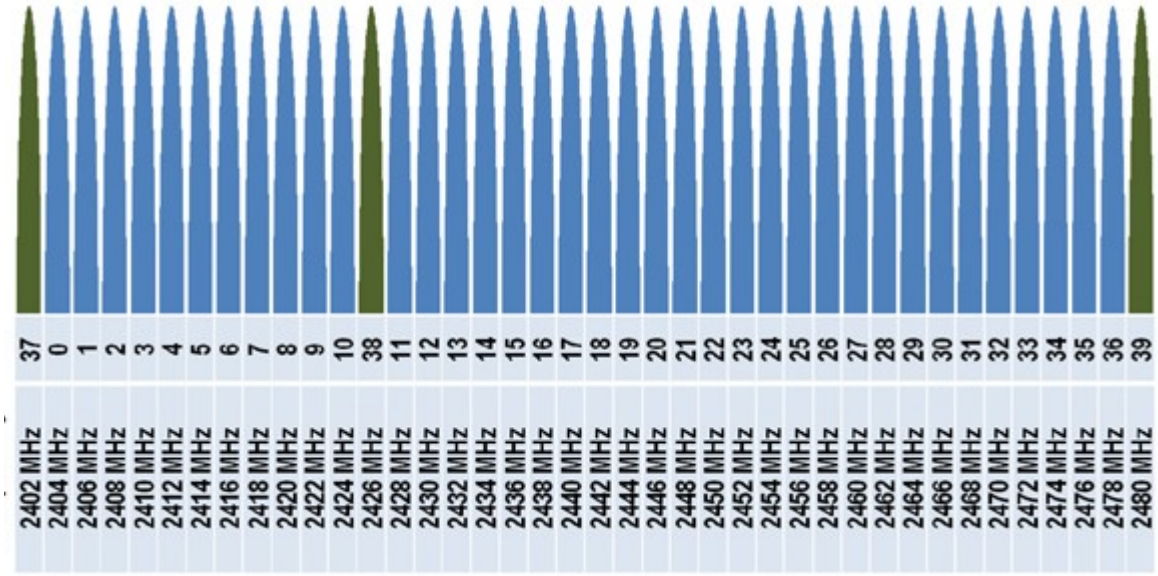
Dentro do escopo das ações aqui elencadas, cabe ressaltar que não somente o modelo *HackRF ONE* pode realizá-las. Qualquer RDS elencado no Quadro 1 é capaz de capturar dados trafegando no espaço dentro da faixa do espectro eletromagnético, desde que respeitadas as características inerentes a cada padrão. Por exemplo, As tecnologias Wi-FI 802.11 utilizam em média 20 MHz de largura de banda para o tráfego de informações. O modelo RTL-SDR possui apenas 3.2 MHz para recepção, logo não pode ser empregado para esta tecnologia.

4.1 TECNOLOGIA *BLUETOOTH*

Os diversos modelos de rádios definidos por *software* (RDS) possuem cada um características específicas que diferenciam as suas formas de emprego (largura de banda, por exemplo). Isto leva também a uma diversificação no desenvolvimento de ferramentas para o escaneamento e captura de diferentes tecnologias.

As ferramentas desenvolvidas atualmente para a captura de sinais *Bluetooth* utilizando RDS possuem um foco voltado para a sua versão mais moderna, o BTLE (*Bluetooth Low Energy*). Nesta tecnologia, os canais são distribuídos de acordo com a Figura 11:

Figura 11 – Distribuição de Canais do BTLE



FONTE: Araujo A. S. *et al* (2012)

Segundo Araujo A. S. *et al* (2012), o BLE trabalha com dois tipos de canais: *advertising channels* e *data channels*, em verde e azul, respectivamente. Os

primeiros tem por função estabelecimento de conexão, descoberta de dispositivos e transmissão *broadcast*. Os demais são utilizados para a comunicação bi-direcional entre os dispositivos, valendo-se do mecanismo de AFH (*Adaptative Frequency Hopping*) a fim de evitar interferência com sinais de Wi-FI/802.11.

Para a captura de tráfego de *Bluetooth* foram realizados dois experimentos.

4.1.1 Captura com a ferramenta ble_dump

Esta ferramenta, disponível no *GitHub* no repositório do usuário “drtyhlpr” (2016), realiza o *dump* de pacotes de BLE utilizando o RDS, sem necessitar de compilação. Os pacotes capturados podem ser salvos para um arquivo de captura (.pcap) ou mostrados diretamente no *software Wireshark* (*DRTYHLPR,* 2016).

Neste experimento foi utilizado o método de amostragem de pacotes simultâneo, sem salvar o *dump*. Para isto foram seguidos os seguintes passos:

a) Criação de um *named pipe*, através do comando:

```
mkfifo /tmp/fifo1
```

b) Início da captura com a ferramenta ble_dump nos *advertising channels*:

```
./ble_dump -o /tmp/fifo1
```

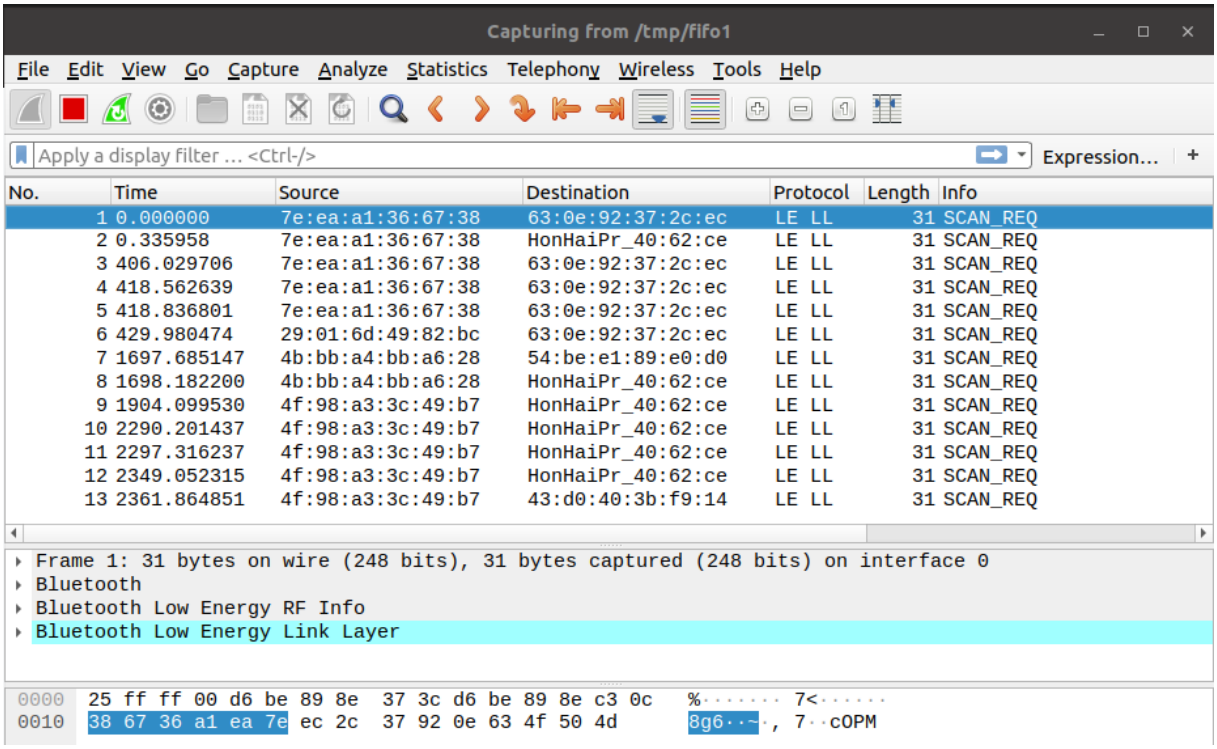
O parâmetro “-o” envia a saída da captura para o *named pipe*.

c) Recebimento dos pacotes no *software Wireshark*:

```
wireshark -S -k -i /tmp/fifo1
```

Após realizados os procedimentos acima, foi realizada a captura de dados, demonstrada na Figura 12:

Figura 12 – Captura de dados com a ferramenta ble_dump



FONTE: o autor (2019)

4.1.2 Captura com a ferramenta btle_rx

Com uma proposta mais robusta, Jiao Xianjun (2015) propõe esta ferramenta como um *sniffer* completo de BLE. O programa em questão se propõe a não somente acompanhar os *advertising channels,* mas também a rastrear o salto de canais durante a transmissão de dados entre dispositivos. Esta funcionalidade se inicia ao capturar um pacote contendo em seus campos o dado ADV_CONNECT_REQ.

Desenvolvido especificamente para os modelos *HackRF One* e *BladeRF*, para sua execução o programa deve ser compilado. Apesar de apresentar uma funcionalidade mais avançada, ao acompanhar o *frequency hopping*, a captura não pode ser salva em um arquivo de saída ou enviada para um programa de monitoramento, como o *Wireshark.*

Sua configuração é bastante flexível, pois todos os parâmetros presentes em um pacote BLE são configuráveis (canal, a*ccess address,* CRC, dentre outros), permitindo inclusive alterar a frequência utilizada para compatibilidade com diferentes tecnologias (XIANJUN, 2015).

Realizando por padrão a captura no canal 37, sua execução é simples,

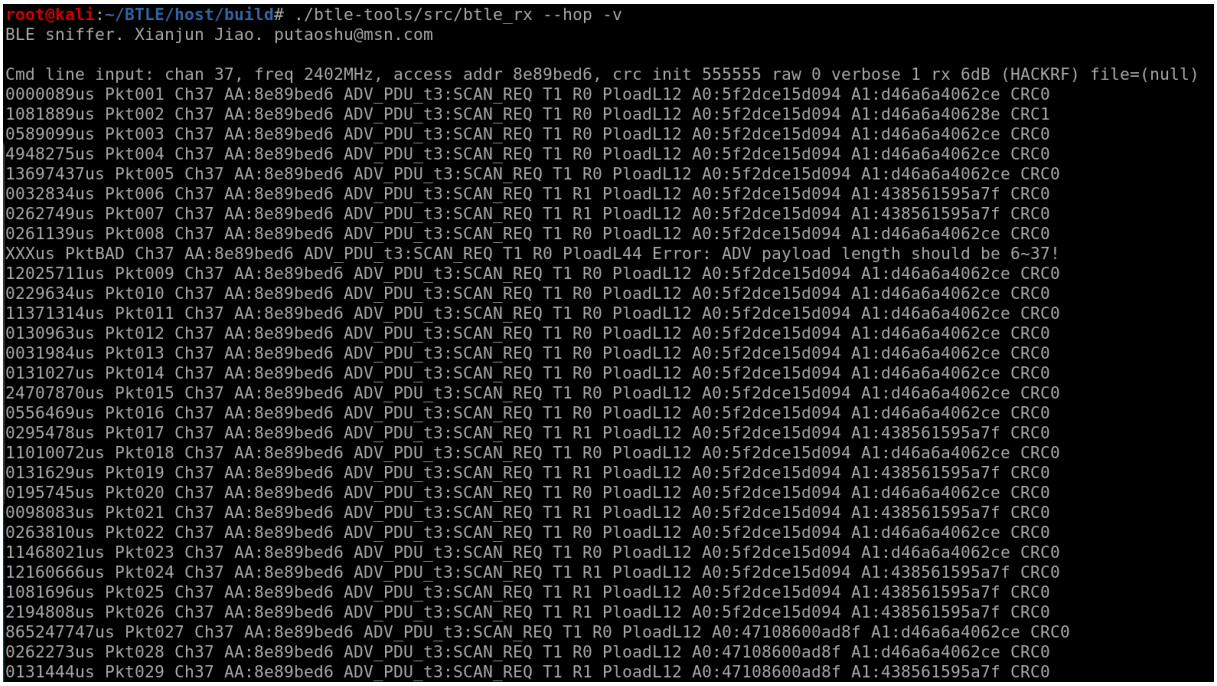
bastando o comando:

```
./btle_rx --hop -v
```

O parâmetro “--hop” visa iniciar o rastreamento de salto de canais, e “-v” mostra informações de erro mais detalhadas.

Realizados estes passos, obtém-se a captura do tráfego, conforme Figura 13:

Figura 13 – Captura de dados com a ferramenta btle_rx



FONTE: o autor (2019)

4.2 TELEFONIA CELULAR

Este experimento se concentrou no tráfego de tecnologia GSM e na captura de códigos IMSI (*International Mobile Subscriber Identity)*.

Estes números são unicamente alocados para cada usuário de uma rede celular, sendo compostos por três identificadores (ROCHA, 2018):

a) *Mobile Country Code* (MCC): identifica o país do usuário;

b) *Mobile Network Code* (MNC): identifica a rede provedora de serviços do usuário; e

c) *Mobile Subscriber Identification Number* (MSIN): identifica o usuário dentro da rede a qual pertence.

Para a a realização da captura, foram feitos procedimentos com a utilização

de diferentes ferramentas.

4.2.1 Descoberta de Estações Rádio-Base (ERB)

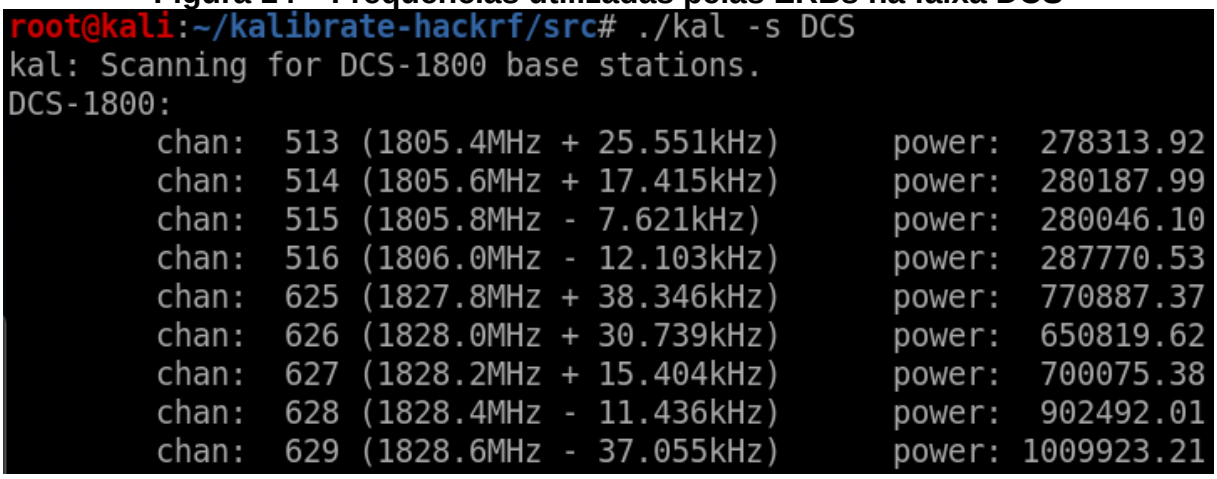
Para esta ação foi utilizada a ferramenta *kalibrate-hackrf,* disponível no *GitHub* de Kang (2014). O autor afirma que o *software* escaneia por ERBs GSM em uma dada faixa de frequência. Como parâmetros, podem ser passadas a frequência diretamente ou o indicador de banda (GSM850, GSM900, EGSM, DCS e PCS, distribuídas entre as operadoras de telefonia celular de acordo com a Anatel, no caso do Brasil) (KANG, 2014).

Foi escolhida a faixa DCS, sendo executado o seguinte comando:

```
./kal -s DCS
```

As frequências utilizadas pelas ERBs dentro do alcance são as mostradas na Figura 14:

Figura 14 – Frequências utilizadas pelas ERBs na faixa DCS



FONTE: o autor (2019)

Por apresentar o sinal mais forte, oferecendo maior estabilidade, foi escolhido o canal 629, à frequência de 1828,6 MHz.

4.2.2 Escaneamento e captura de dados no canal de interesse

O interesse no escaneamento de redes de telefonia móvel existe desde o seu advento. Neste ínterim, Piotr Krysik desenvolveu o projeto *gr-gsm*, onde através do seu GitHub disponibiliza ferramentas para a exploração desta faixa do espectro eletromagnético. Seu mote é providenciar um conjunto de aparatos para receber informações transmitidas por equipamentos e dispositivos que utilizam a tecnologia

GSM (KRYSIK, 2010).

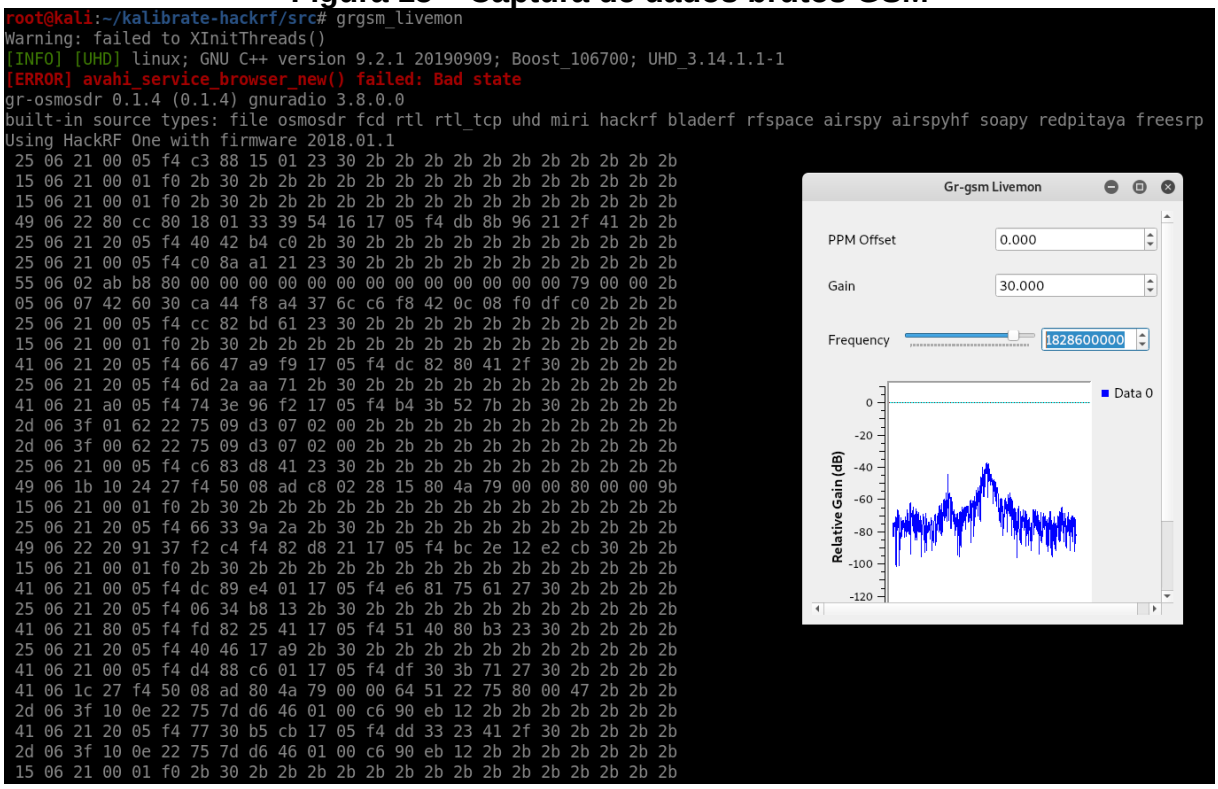
Do conjunto de ferramentas oferecidas pelo pacote de *software gr-gsm*, será utilizado o *grgsm_livemon*, que realiza o monitoramento em um determinado canal, coletando dados brutos. Estes dados coletados devem ser repassados a outra aplicação para a devida análise.

Para o funcionamento da aplicação é necessário após compilado e instalado, executar o comando:

```
grgsm_livemon
```

Aberta a aplicação, deve ser configurada a frequência do canal encontrado no tópico 4.2.1. Realizados estes passos, a saída bruta de dados será mostrada na tela, conforme Figura 15:

Figura 15 – Captura de dados brutos GSM



FONTE: o autor (2019)

4.2.3 Envio dos dados para a aplicação *Wireshark*

Os dados de GSM tramitam entre seus dispositivos também na forma de pacotes. Para uma melhor análise das informações capturadas, é de suma importância facilitar a sua visualização e utilizar as ferramentas adequadas. Nesta situação, os dados são enviados para a aplicação através do comando:

```
wireshark -k -Y '!icmp && gsmtap' -i lo
```

Os parâmetros passados são os seguintes:

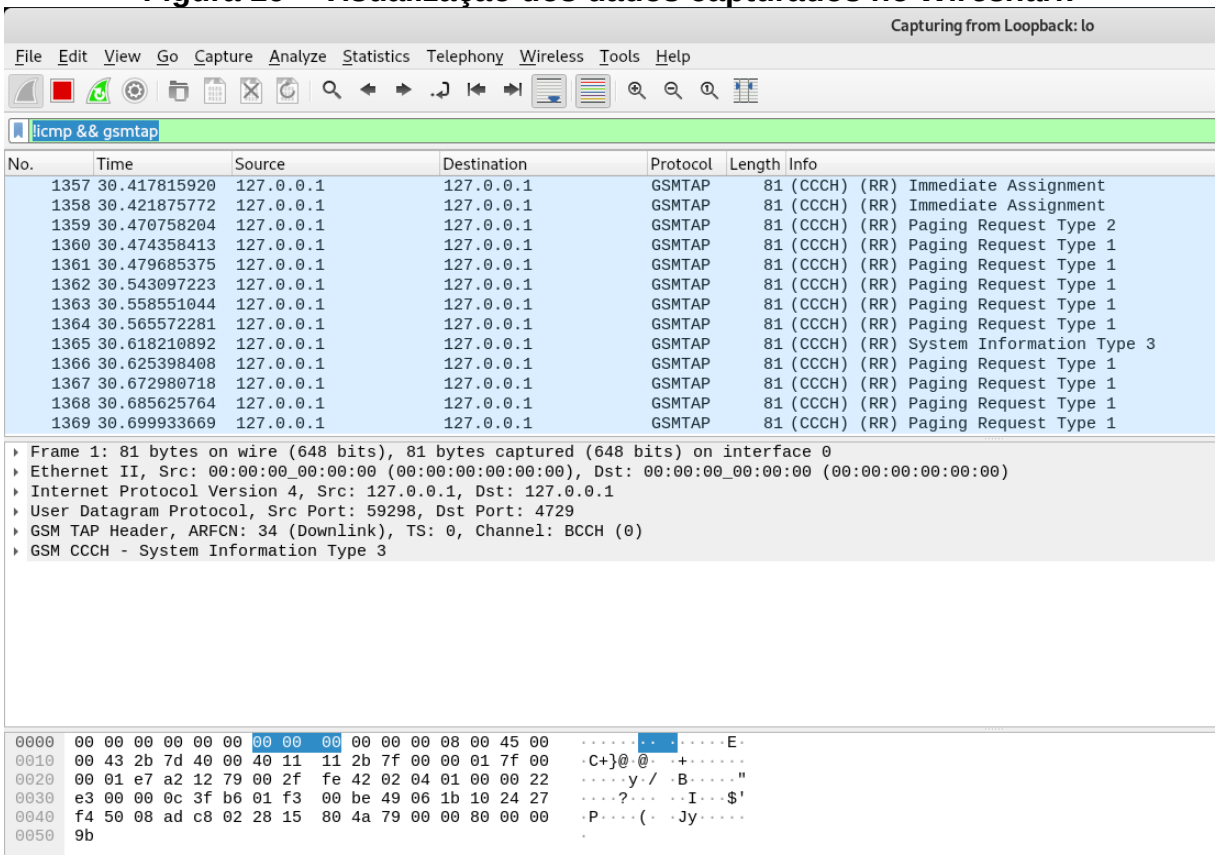
a) “-k”: inicia a captura imediatamente;

b) “-Y”: passa um filtro de captura. No comando em questão, todos os pacotes ICMP não serão capturados, ao contrário dos que contenham o rótulo ‘gsmtap’; e

c)”-i”: determina uma interface para a captura.

Após a execução do comando, o *Wireshark* converte os dados em uma captura de fácil análise, conforme demonstrado na Figura 16.

Figura 16 – Visualização dos dados capturados no *Wireshark*



FONTE: o autor (2019)

4.2.4 Captura de IMSI

As informações contidas nos números IMSI também trafegam por estes dados que estão sendo capturados pelo procedimento do tópico 4.2.2. Desta forma, é necessária uma ferramenta que realize a filtragem dos dados e extraia os informes de interesse para este contexto.

No *GitHub do usuário “Oros42”* é disponibilizado o script em *Python*

simple_IMSI-catcher.py, que realiza a extração do código de interesse, sem a necessidade de instalar ou compilar. Para isto executa-se o comando:

```
python simple_IMSI-catcher.py –sniff
```

Após sua execução, é retornada na tela a saída com as informações desejadas, de acordo com a Figura 17:

Figura 17 – Códigos IMSI capturados

```
root@kali:~/IMSI-catcher# python simple_IMSI-catcher.py --sniff
Nb IMSI ; TMSI-1     ; TMSI-2     ; IMSI            ; country      ; brand      ; operator              ; MCC  ; MNC   ; LAC    ; CellId
1       ;            ;            ; 716 10 1637085964 ; Peru         ; Claro      ; América Móvil Perú     ; 724  ; 05    ; 2221   ; 4132
2       ; 0xe88ac401 ; 0xf3310e41 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xcb83c961 ; 0xf3310e41 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
3       ;            ;            ; 724 05 4208523051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xfe825541 ; 0xe88ac401 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132

4       ; 0xec888241 ; 0xd580d221 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
5       ;            ;            ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc0287a01 ;            ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc4297a01 ; 0xec888241 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xe0813f21 ; 0xd580d221 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xfb800b71 ;            ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc8337401 ; 0xc4297a01 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xfd81eb01 ; 0xfb800b71 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xd3832321 ; 0xc8337401 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0x692ffc3c ; 0xfd81eb01 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc02b4661 ;            ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc5323f61 ; 0xc02b4661 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xdd315201 ; 0xc5323f61 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc98bdd61 ; 0x692ffc3c ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0x2b2b97c7 ; 0xcb83c961 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xf730ec41 ;            ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xd3832741 ; 0xd989d061 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xa13b8987 ; 0xd3832741 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xca8adc21 ; 0xa13b8987 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xa13b8987 ; 0xca8adc21 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xa645f5ad ; 0xd989d061 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xe7329341 ; 0xd3832321 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xac2fab04 ; 0xf730ec41 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0x6f2b6ed9 ; 0x2b2b97c7 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0x2b2b97c7 ; 0x6f2b6ed9 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xdf295021 ; 0x2b2b97c7 ; 724 05 2211961896 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xc5808941 ; 0xa13b8987 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xdc89e401 ; 0xac2fab04 ; 724 05 2211673051 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
        ; 0xe4295d21 ; 0xe7329341 ; 724 05 2308061270 ; Brazil       ; Claro      ; Claro                  ; 724  ; 05    ; 2221   ; 4132
```

FONTE: o autor (2019)

4.3 *LOW POWER DEVICE 433 MHz*

A captura de sinais na faixa de 433 MHz é realizada utilizando-se ferramentas próprias para o HackRF One presentes nos repositórios de diversas distribuições *Linux.* Para este caso, foram instalados três pacotes (*hackrf*, *libhackrf0* e *libhackrf-dev),* os quais proporcionam os aparatos necessários para a captura dos sinais.

Primeiramente, verifica-se a disponibilidade do RDS com o comando:

```
hackrf_info
```

O qual apresenta o retorno mostrado na Figura 18:

Figura 18 – Informações sobre o HackRF One

```
root@kali:~# hackrf_info
hackrf_info version: unknown
libhackrf version: unknown (0.5)
Found HackRF
Index: 0
Serial number: 0000000000000000644064dc3735abcd
Board ID Number: 2 (HackRF One)
Firmware Version: 2018.01.1 (API:1.02)
Part ID Number: 0xa000cb3c 0x0064475f
```

FONTE: o autor (2019)

O objeto-teste desta demonstração será a captura e replicação do código de travamento e destravamento de um veículo Fiat Argo ano 2018, modelo demonstrado na Figura 19.

Figura 19 – Modelo de Chave de um Fiat Argo 2018

FONTE: Carros Fiat, (2018)

4.3.1 Descoberta de Frequência Utilizada

Para a captura do sinal, primeiramente deve ser descoberta a frequência utilizada pela chave do veículo. Foi utilizado para isto o *software Gqrx,* disponível nos repositórios de pacotes das principais distribuições *Linux*.

O *software Gqrx* é uma ferramenta *open source* para trabalho com rádios definidos por *software*, estruturada com *GNU Radio* e o kit de ferramentas gráficas *Qt*. O programa oferece as seguintes capacidades, dentre as suas diversas (GQRX, 2013):

a) Descobrir dispositivos RDS conectados à estação de trabalho;

b) Processar dados I/Q de dispositivos RDS;

c) Alterar frequência e ganho;

d) Demodular AM, SSB, CW, FM-N e FM-W (*mono* e *stereo*);

e) Filtro de passagem de banda variável; e

f) Reduzir ruído, *squelch* e AGC.

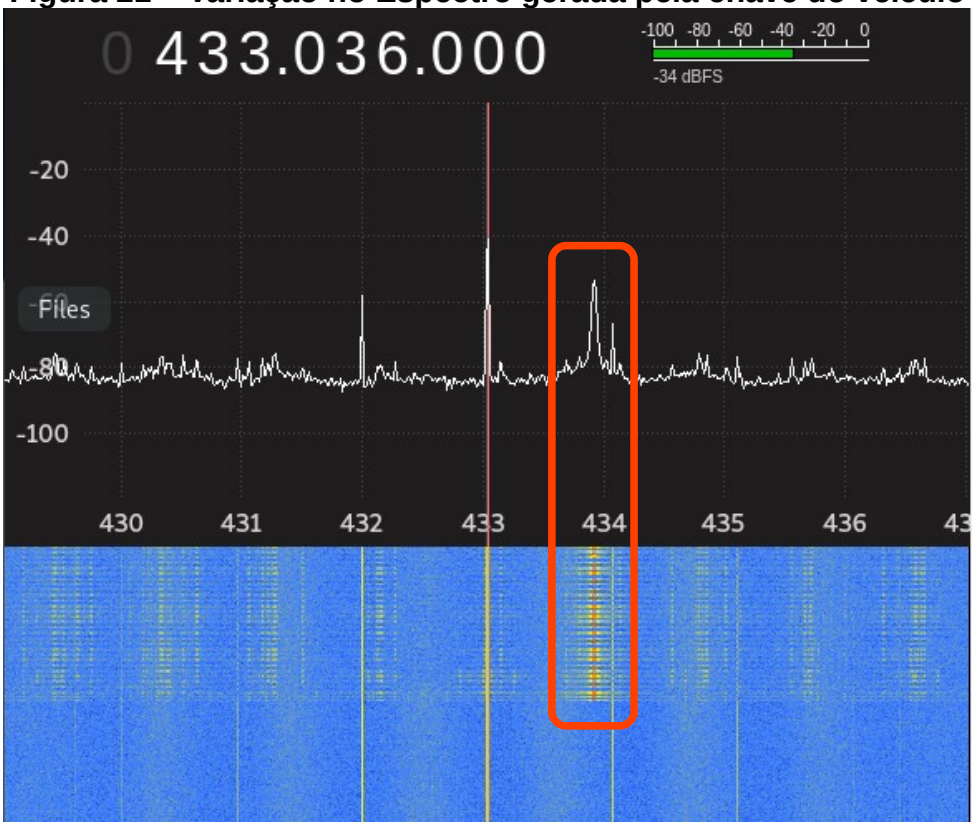
Configurado o programa para a frequência central de 433 MHz, observa-se o seguinte status no espectro eletromagnético, conforme Figura 20:

Figura 20 – Visualização do Espectro na faixa de 433 MHz

FONTE: o autor (2019)

Ao serem pressionados os botões de destravamento e travamento do controle, ocorre a seguinte variação no espectro, conforme Figura 21:

Figura 21 – Variação no Espectro gerada pela chave do veículo



FONTE: o autor (2019)

Constatou-se após análise na variação gerada que a frequência central utilizada para transmissão é de 433,915 MHz. Os sinais de travamento, destravamento, abertura de porta-malas e acendimento de faróis utilizam o mesmo fluxo.

4.3.2 Captura do sinal transmitido

Dentro da suíte de ferramentas adquirida com a instalação dos pacotes do *HackRF One* há uma funcionalidade que realiza a captura e transmissão de sinais. Com o comando `hackrf_transfer` estas ações são realizadas, o que permitirá também a análise do sinal apanhado.

Sabendo-se da frequência que o equipamento utiliza, foi dado o seguinte comando:

```
hackrf_transfer -f 433915000 -s 11000 -w
```

Este comando foi executado para a captura dos sinais de travamento e destravamento do veículo. Os parâmetros são:

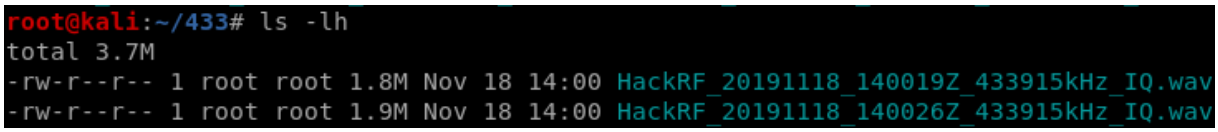
a) “-f”: frequência utilizada, em *hertz*;

b) “-s”: taxa de amostragem, em *hertz*; e

c) “-w”: grava o sinal capturado em um arquivo no formato *wav*.

A captura gerou os arquivos mostrados na Figura 22:

Figura 22 – Arquivos de captura dos sinais de travamento e destravamento do veículo



FONTE: o autor (2019)

5 ANÁLISE DE RESULTADOS

Este capítulo aborda os aspectos colhidos de cada experimento abordado no

capítulo anterior, visando comparar resultados e, por fim, apresentar uma metodologia para escaneamento e captura de dados utilizando o RDS.

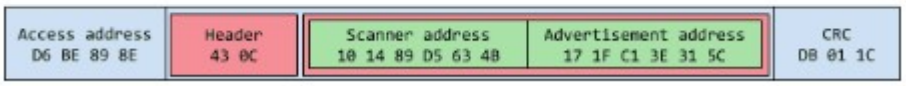
5.1 TECNOLOGIA *BLUETOOTH*

Os testes para o escaneamento e captura de pacotes *Bluetooth* foram realizados utilizando-se os *advertising channels* (37, 38 e 39). Esta ação foi feita visando a descoberta de dispositivos e periféricos que utilizam BLE dentro do raio de alcance do sinal dos emissores.

Com o software *ble_dump*, o teste retornou catorze pacotes com a informação SCAN_REQ. Foram utilizados um computador pessoal com interface *Bluetooth* nativa e um aparelho celular modelo Moto G6 Plus. Dentro deste escopo, foram detectados quatro dispositivos realizando o escaneamento de três outros dispositivos.

Pacotes SCAN_REQ possuem a estrutura mostrada na Figura 23:

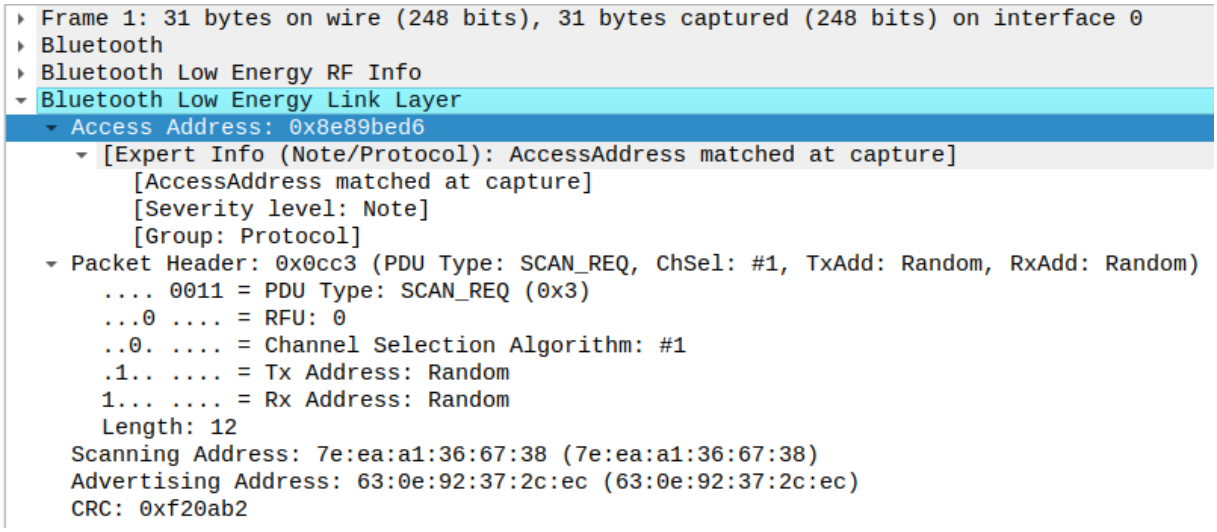
Figura 23 – Modelo de pacote *Bluetooth*



FONTE: Augustyn (2018)

Na captura realizada, as mesmas informações podem ser encontradas concatenadas. O *software Wireshark*, ao processar os dados recebidos, incorpora-os à camada mais externa do pacote, contendo as interiores informações como canal utilizado, endereço de acesso de referência, dentre outras, demonstrado na Figura 24.

Figura 24 – Pacote *Bluetooth* capturado pela ferramenta ble_dump visualizado no *Wireshark*



FONTE: o autor (2019)

O teste realizado com a ferramenta *Btle_rx* apresentou resultados semelhantes. Como mencionado anteriormente, o *software* ainda não oferece um recurso de saída da captura para arquivo. A figura 13 demonstra que as informações mostradas em sua saída são mais resumidas, limitando-se ao canal, *payload* com *flags* omitidas e ordenamento dos pacotes.

O acompanhamento do salto de frequência não foi observado durante o uso desta ferramenta. Foram realizadas conexões entre dispositivos durante o procedimento, entretanto não foi observado um pacote com a informação ADV_CONNECT_REQ.

Os dois *softwares* foram testados durante intervalos de tempo iguais. No que se refere à eficiência, *btle_rx* capturou um número superior de informações em comparação ao *ble_dump*, realizando o salvamento de trinta pacotes.

A função de acompanhamento de *frequency hopping* tem uma importância vital para a atividade cibernética. Com o uso atual constante como ferramenta para conversas telefônicas, envio de arquivos e conexão de *gadgets*, um fluxo de pacotes capturado pode oferecer diversas informações, surgindo assim a possibilidade de mais um vetor de exploração.

Como exemplos, uma conversa telefônica pode ser captada capturando-se o fluxo de pacotes entre o aparelho celular e o fone de ouvido *Bluetooth* utilizado. Outra possibilidade é a captura do fluxo de informações entre um *smartphone* e um *smartwatch*, o qual contém notificações de aplicativos de mensagens instantâneas, notificações de aplicativos bancários, dentre outras situações onde o RDS pode ser utilizado como uma nova forma de ataque.

Sendo o *HackRF One* um dispositivo que atua em *half-duplex*, em todos os testes o mesmo atuou apenas de forma passiva, sem realizar qualquer transmissão. Desta forma, seu uso é indetectável para os dispositivos explorados, garantindo a anonimização das ações.

5.2 TELEFONIA CELULAR

A pesquisa desenvolvida neste trabalho limitou-se a explorar o espectro da tecnologia 2G (GSM) como prova de conceito da potencialidade do Rádio Definido por Software, devido a implementação de criptografia e de sistemas de autenticação nas terceira e quarta gerações de telefonia móvel, o que demandaria tempo superior ao disponível para a complitude das atividades.

As ferramentas utilizadas valem-se da implementação do GNU Radio. Sua definição é a seguinte:

> “GNU Radio é um conjunto de ferramentas de código aberto que provê um ambiente de desenvolvimento e blocos de processamento para implementar rádios definidos por software. […] As aplicações em GNU Radio são desenvolvidas utilizando a linguagem de programação Python, onde são construídas ligações entre os blocos de processamento.” (GARCIA REIS *et al.*, 2012, p. 1158)

E sobre a sua utilização:

> “Ele pode ser usado com hardware externo de RF - de baixo custo e prontamente disponível - para criar rádios definidos por software [RDS] ou sem hardware em um ambiente de simulação. É amplamente utilizado em ambientes de pesquisa, da indústria, acadêmico, governamental e de entusiastas para apoiar pesquisas de comunicações sem fio e sistemas de rádio do mundo real” (GNURADIO, 2019 – tradução nossa)

O projeto *gr-gsm* se vale desta tecnologia para realizar suas diversas ações de escaneamento e captura. No experimento em questão, as informações adquiridas foram processadas por dois *softwares: Wireshark* e Simple_IMSI-catcher.

O primeiro programa oferece a compatibilidade com arquivos de captura de tráfego de diversas tecnologias. Alinhado à sua interface amigável e sua flexibilidade de uso, seu uso é consagrado como ferramenta de análise.

Ao capturar o tráfego, a aplicação *grgsm_livemon* envia para o *Wireshark* dados no formato GSMTAP. Segundo WIRESHARK (2019), o GSMTAP é um pseudo-cabeçalho que é utilizado para encapsular quadros de uma interface GSM em pacotes UDP/IP. Estes são encaminhados por padrão para a porta UDP 4729 da interface de rede local.

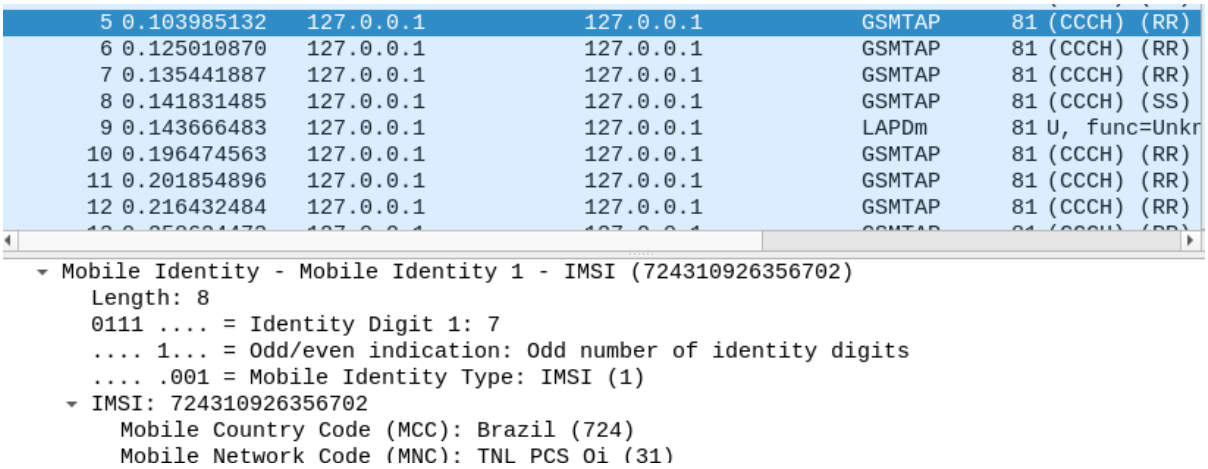
Realizada a captura, foram analisados os pacotes no *Wireshark.* As quatro primeiras camadas dos pacotes são adaptações realizadas pelo *software,* onde foram inseridos os encapsulamentos das camadas TCP/IP. Após, encontra-se o *payload,* contendo duas camadas:

a) GSM *Tap Header*: contém informações de camada física, como nível do sinal, número de *frame* GSM e número de antena; e

b) GSM CCCH: carrega as informações funcionais, como o IMSI propriamente dito e modo de paginação.

No pacote capturado abaixo, exemplificado na Figura 25, foi encontrado um IMSI com suas informações inerentes (MCC e MNC).

Figura 25 – Pacote GSMTAP capturado pela ferramenta grgsm_livemon visualizado no *Wireshark*



FONTE: o autor (2019)

Constatou-se, analisando a captura, que não são todos os pacotes que trafegam IMSI. A grande maioria das informações adquiridas contém dados de controle gerados pelas Estações Rádio-Base. As frequências encontradas no tópico 4.2.1 pertencem a diferentes operadoras. Para explorar a faixa de uma operadora específica, o *link* https://www.teleco.com.br/areasc.asp oferece para consulta a distribuição das frequências utilizadas por cada uma.

A mesma informação adquirida com o *software Wireshark* em relação aos IMSI pode ser analisada com o Simple_IMSI-catcher. Sendo seu objetivo-fim a colheita de IMSI, o seu código realiza a filtragem da informação recebida pelo *grgsm_livemon*.

Todas as ferramentas utlizadas atuam de forma passiva, onde o HackRF One atua apenas na recepção de informações. A captura de IMSI foi utilizada como prova de conceito pois, dado o potencial do RDS, seu potencial se expande a escuta

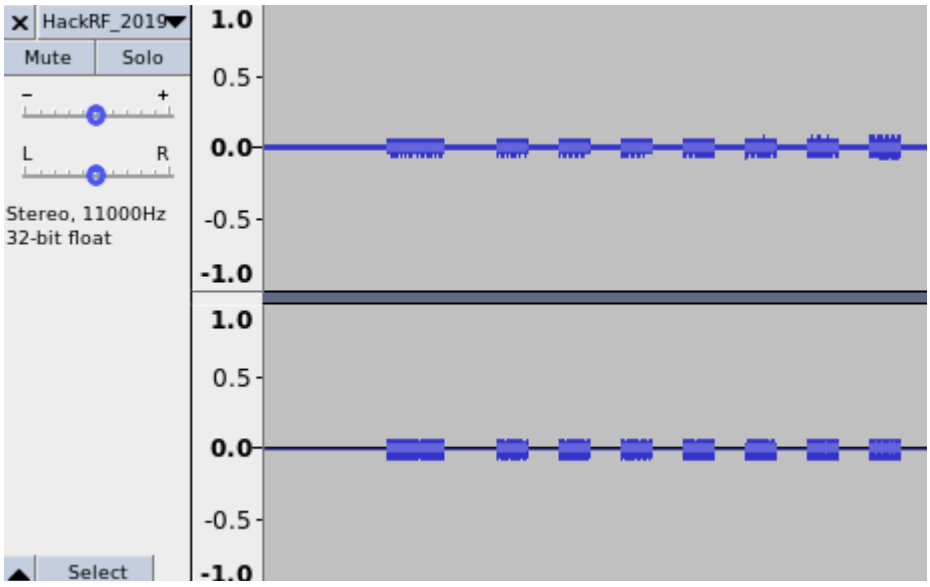
de ligações, captura de SMS, dentre outros.

Como um simples exemplo de potencial de utilização apenas com a captura de IMSI, pode ser desenvolvida a seguinte hipótese de emprego: havendo a distribuição de RDS na faixa de fronteira do Brasil (um RTL-SDR é suficiente para a captura de IMSI, reduzindo consideravelmente o custo de implantação) realizando o escaneamento da rede e a captura, um IMSI adquirido via canal de inteligência de um agente perturbador da ordem pública (APOP) pode ser detectado assim que tente cruzar a fronteira, denunciando sua posição. Implantado ainda de forma capilar em um perímetro urbano, a detecção de IMSI pode levantar a posição e a trajetória de um determinado alvo, levantando-se assim mais um vetor de exploração para ações cibernéticas.

5.3 *LOW POWER DEVICE 433 MHz*

A partir dos dados capturados no tópico 4.3.2, sua análise foi realizada utilizando-se o *software Audacity*. Este é um programa de código aberto utilizado para a produção, edição e análise de faixas de áudio. Como a ferramenta *hackrf_transfer* salvou o sinal capturado no formato *.wav,* os dados obtidos podem ser analisados por esta aplicação, de acordo com o mostrado na Figura 26.

Figura 26 – Sinal de destravamento de veículo analisado no Audacity



FONTE: o autor (2019)

Conforme analisado no trabalho de Azzolin (2018), chaves de veículos atuais implementam uma sofisticação no sinal, denominada *rolling code:*

> Um método onde um potencial ladrão pode adquirir a informação relativa ao código transmitido é através da gravação do sinal transmitido no momento em que o dono do veículo apertar o botão. Quando o dono do veículo nào estiver mais presente, o ladrão pode reproduzir novamente o sinal copiado, podendo assim obter acesso fraudulento ao veículo. Como forma de prevenir que sinais previamente copiados possam ser aceitos como sinais válidos, é recomendável a utilização de técnicas de codificação que utilizem *rolling codes*, onde o código muda a cada transmissão. Nesse tipo de sistema, o receptor precisa sistematicamente atualizar o código válido esperado para que possa sempre estar sincronizado com o sinal que será transmitido. (MICHAELS, 1995, p. 3, tradução nossa).

Esta técnica é verificada na Figura 27, onde há o primeiro bloco de sinal, o preâmbulo de sincronização, e após o sinal de destravamento propriamente dito. Sendo esta implementação realizada para evitar ataques do tipo replay, Azzolin (2018) aborda em seu trabalho que uma forma de realizar esta ação com sucesso é utilizando a técnica de *jamming*, onde um sinal interferente impede que o veículo receba o sinal enviado enquanto o atacante captura o sinal do dono do veículo.

Entretanto, para este experimento a técnica de *replay* foi utilizada de maneira mais simples. Em um cenário real pode ser mais vantajoso obter contato direto à chave do dono do veículo, seja através de um simples acesso por falta de cuidado ou por técnicas de engenharia social, do que esperar o seu uso e conseguir sucesso com um *jamming.*

Obtido o acesso à chave do veículo, os sinais são copiados conforme tópico 4.3.2. Neste procedimento, o veículo não pode ter contato com a transmissão emitida da chave, senão o código copiado será invalidado. Após isto, deve ser assegurado que o código adquirido seja utilizado no veículo antes do uso da chave pelo dono, caso contrário a informação captada também será inutilizada.

Para a abertura do carro, o sinal de destravamento foi retransmitido através do comando:

```
hackrf_transfer -f 433915000 -t HackRF_20191118_140019Z_433915kHz_IQ.wav
```

O parâmetro “-t” transmite o arquivo passado contendo o sinal. Realizada esta transmissão, o veículo foi destravado e o acesso ao seu interior foi liberado.

Para diversos casos onde se utilizam transmissões RF para controle de acesso, como portões e alarmes de áreas sensíveis, ainda não há a implementação da proteção para captura e *replay attack*. Logo, para estes casos, a vulnerabilidade é latente e de fácil exploração, possuindo assim um potencial muito importante para ações onde o acesso físico seja um fator determinante para o sucesso de uma ação.

5.4 METODOLOGIA PARA A IMPLEMENTAÇÃO E EMPREGO DO RÁDIO DEFINIDO POR SOFTWARE

Heinäaro (2015) realizou um ataque de força bruta ao sistema de alarme de seu veículo utilizando um RDS, as informações constantes do *datasheet* do circuito integrado do sistema de destravamento e um módulo do *metasploit* escrito por ele mesmo.

Ao relatar que com a mesma matriz de procedimentos, Heinäaro (2015) foi capaz de realizar um ataque cibernético a uma rede de rádios táticos, o mesmo propõe uma metodologia onde, sendo aplicada, é possível atacar qualquer sistema de comunicação sem fio com o mesmo conjunto de ferramentas.

Neste trabalho os experimentos foram realizados com *softwares* já desenvolvidos. Logo, com este tópico, visa-se oferecer uma proposta para a evolução do conceito, a fim de ser aprimorada a liberdade de ação no espaço cibernético.

Desta forma, para adicionar o suporte para o escaneamento de um novo sistema, devem ser desenvolvidos (Heinäaro, 2015):

a) O formato de onda da radiofrequência para GNU Radio;

b) *Driver* do GNU Radio para o *hardware* utilizado na tecnologia alvo;

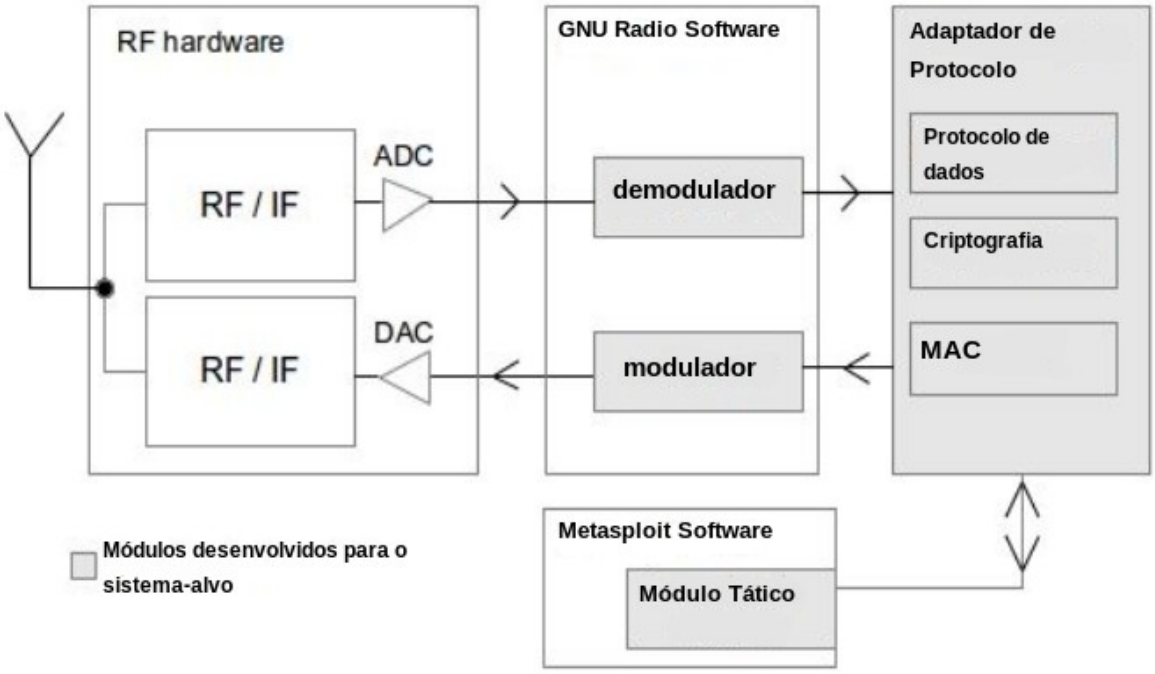
c) Camada de enlace;

d) Camada de encriptação, se for o caso;

e) Camada de protocolo de dados; e

f) Módulo de ataque do *metasploit*, se for o caso.

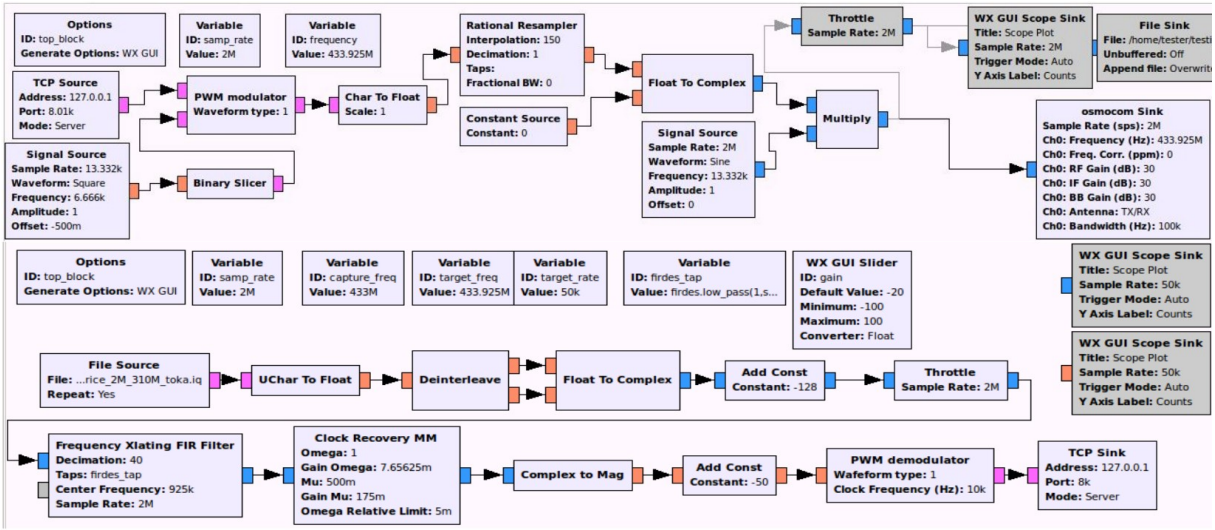
Figura 27 – Configuração para o teste de penetração a uma tecnologia sem fio utilizando RDS



FONTE: Heinäaro, 2015, tradução nossa

Conforme demonstrado na Figura 27, no primeiro bloco temos o RDS, com os seus componentes físicos de ADC e DAC. NO bloco “GNU Radio SW”, o processamento do sinal RF a ser transmitido e recebido é realizado. Esta fase é realizada a nível *software*, sendo desenvolvida através de aplicações como o *GNURadio-Companion*. A Figura 28 oferece um exemplo da configuração realizada por Heinäaro (2015) para realizar o processamento do formato de onda utilizado pelos rádios táticos que utilizou em seu ataque:

Figura 28 – Exemplo de configuração de RDS utilizando o *software* GNURadio-Companion



FONTE: Heinäaro, 2015

Os dados recebidos após o processamento pelos blocos do GNURadio se tratam basicamente de bits 1 e 0. A partir deste ponto, o terceiro bloco da figura 27, o seu tratamento e a criação de ferramentas ocorrem de acordo com a necessidade e o objetivo do atacante.

Havendo o desdobramento e a evolução na exploração do RDS e no desenvolvimento de artefatos para redes sem fio, os vetores de ataque são multiplicados. Utilizando-se como exemplo um *smartphone*, o mesmo utiliza diversas tecnologias sem fio. De forma simples, o vetor de ataque mais explorado é o acesso via rede 802.11 (Wi-FI). Sendo a forma de acesso mais utilizada, é uma das que mais sofrem atualizações e *patches* de segurança. Entretanto, o mesmo dispositivo celular ainda utiliza dados móveis, *bluetooth*, NFC, GPS dentre outras implementações, dependendo do dispositivo. Com o aprimoramento do emprego do RDS, as possibilidades aumentam exponencialmente.

6 CONCLUSÃO

Com os experimentos e análises realizados neste trabalho, é possível concluir que o Rádio Definido por *Software* (RDS) pode ser empregado como uma estação capaz de capturar e analisar dados de diferentes tecnologias sem fio, apresentando-se como uma solução robusta e de baixo custo.

O primeiro objetivo específico, explorado no capítulo 2, foi alcançado ao ser explorada a flexibilidade do RDS no que tange à sua configuração e à sua extensa compatibilidade com as diversas tecnologias sem fio.

O segundo objetivo específico também foi alcançado, ao serem capturados dados de transmissões Bluetooth, de telefonia móvel e de transmissões RF 433 MHz. As informações obtidas oferecem características dos equipamentos empregados e de rotinas de usuários a partir do colhimento de dados trafegando abertamente no espectro eletromagnético.

Os resultados obtidos foram analisados e comparados, demonstrando a viabilidade do RDS na sua atuação em conjunto com ferramentas de exploração de diferentes tecnologias. Verificou-se que, após processados os dados, estes podem ser convertidos em pacotes através da inserção de pseudo-cabeçalhos, permitindo uma análise de tráfego facilitada através do *software Wireshark*.

Dada a implementação dos experimentos no curto prazo de realização deste curso, foram realizadas provas de conceito que não abordam as potencialidades do RDS dentro de cada tecnologia em sua complitude. Com as devidas ferramentas e, se for o caso, adaptação de antenas pro formato de onda devido, é possível realizar a captura de qualquer dado que trafegue pelo espectro, como imagens de satélites meteorológicos, por exemplo.

E é por sua ampla atuação no espectro que o RDS se apresenta como um dispositivo de ampla utilidade tanto para a Guerra Eletrônica quanto para a Guerra Cibernética, visto que as duas áreas valem-se do espectro eletromagnético para a consecução de suas ações.

Ainda, na execução deste trabalho, foi constatado que o RDS pode ser empregado de forma modular e capilar. Ao passo que com seu emprego, uma captura *Bluetooth* pode ser realizada com o uso pontual e móvel de um único aparelho, a distribuição geográfica ampla no território nacional de diversos RDS permitem o escaneamento de diferentes redes e o monitoramento de transmissões e

de alvos suspeitos, contribuindo a nível governamental para o aperfeiçoamento da defesa do Estado brasileiro.

Para trabalhos futuros, sugerem-se as seguintes pautas:

a) Captura e análise de dados de uma rede tática de rádios de emprego da Força Terrestre, visando a captura de arquivos e dados de GPS;

b) Desenvolvimento de uma ferramenta utilizando o GNU Radio para a exploração e ataque a NFC;

c) Estudo de caso, combinando-se um ataque de *downgrade* realizado pela Guerra Eletrônica e a captura de dados (SMS, ligações, dentre outros) a uma ERB que utilize tecnologia de terceira ou quarta geração; e

d) Exploração de enlaces Wi-FI com o RDS *BladeRF.*

REFERÊNCIAS BIBLIOGRÁFICAS

ALECRIM, Emerson. **Tecnologia Bluetooth**. Disponível em: <https://www.infowester.com/bluetooth.php>. Acesso em 7 nov 2019.

ARAUJO, A. S. *et al*. **Bluetooth Low Energy**. Disponível em: <https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2012_2/bluetooth/index.htm>. Acesso em: 10 nov 2019.

AUGUSTYN, Przemek. **Bluetooth Low Energy Sniffing Guide**. Disponível em: <https://www.polidea.com/blog/bluetooth-low-energy-sniffing-guide/#packet-analysis>. Acesso em: 11 nov 2019.

AZZOLIN, Claisso Pires. **Utilização do equipamento RTL-SDR para coleta de informações**. 2018. 46 folhas. Monografia (Curso de Guerra Cibernética para Oficiais) – Centro de Instrução de Guerra Eletr6onica, Brasília, 2018.

CARROS Fiat. **Chave Canivete – Fiat Argo 2018.** Disponível em: <https://carro-fiat.blogspot.com/2018/04/chave-canivete---fiat-argo-2018.html>. Acesso em: 10 nov 2019.

DILLINGER, M., MADANI, K., ALONISTIOTI, N. **Software Defined Radio: Architectures, Systems and Functions**. Wiley & Sons, 2003

GARCIA REIS, A. *et al*. **Introduction to the Software-defined Radio Approach**. IEEE LATIN AMERICA TRANSACTIONS, Vol. 10, n° 1, janeiro, 2012.

GITHUB. **SDR Bluetooth LE packet dumper**. Página do *GitHub.* Disponível em: <https://github.com/drtyhlpr/ble_dump>. Acesso em: 10 nov 2019.

GNURADIO. **About GNU Radio**. GNU Radio Foundation. Disponível em: <https://www.gnuradio.org/about/>. Acesso em: 12 nov 2019.

GQRX. **Welcome to gqrx**. Disponível em: <http://gqrx.dk/>. Acesso em: 11 nov 2019.

GREAT Scott Gadgets. **HackRF One.** Disponível em <https://greatscottgadgets.com/hackrf/one/>. Acesso em: 5 nov 2019.

HEINÄARO, Kimmo. **Cyber Attacking Tactical Radio Networks**. 2015 International Conference on Military Communications and Information Systems (ICMCIS), IEEE, maio, 2015.

KANG, Wang. **Kalibrate for HackRF.** Página do *GitHub*. Disponível em: <https://github.com/scateu/kalibrate-hackrf>. Acesso em: 11 nov 2019.

KRYSIK, Piotr. **GNURadio blocks and tools for receiving GSM transmissions**. Página do *GitHub*. Disponível em: <https://github.com/ptrkrysik/gr-gsm>. Acesso em: 11 nov 2019.

LO, Ken. **Download Speeds: What Do 2G, 3G, 4G & 5G Actually Mean?**. Disponível em: <https://kenstechtips.com/index.php/download-speeds-2g-3g-and-4g-actual-meaning>. Acesso em 8 nov 2019.

MACHADO-FERNÁNDEZ, J. R. **Software Defined Radio: Basic Principles and Applications***. Revista Facultad de Ingeniería,* 2015.

OROS42. **IMSI-catcher**. Página do *GitHub*. Disponível em: <https://github.com/Oros42/IMSI-catcher>. Acesso em: 11 nov 2019.

REED, Hugo. **Car Remote not Working? 5 Ways To Accurately Diagnose Your Vehicle***.* Disponível em: <https://unitedlocksmith.net/blog/car-remote-not-working-5-ways-to-accurately-diagnose-your-vehicle>. Acesso em 8 nov 2019.

ROCHA, Lucas Lima da. **Métodos de Ataque em Redes Celulares 2G, 3G e 4G**. 2018. 43 folhas. Monografia (Curso Básico de Guerra Eletrônica para Oficiais) – Centro de Instrução de Guerra Eletrônica, Brasília, 2018.

SANTOS, Ricardo Di Lucia. **Redes GSM, GPRS, EDGE e UMTS**. Disponível em: <https://www.gta.ufrj.br/ensino/eel879/Anos-anteriores/2008-2/trabalhos_vf/ricardo/2.html>. Acesso em 7 nov 2019.

SILVA, W. *et al.* **Introdução a Rádios Definidos por *Software* com aplicações em *GNU Radio.*** Vitória: Universidade Federal do Espírito Santo, XXXIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2015), cap. 5, maio, 2015.

SVERZUT, José Umberto. **Redes GSM, GPRS, EDGE e UMTS: evolução a caminho da quarta geração (4G)**. 4. ed. São Paulo: Érica, 2015.

TANEMBAUM, A. S. **Redes de Computadores**, Tradução da 4ª Edição, Rio de Janeiro: Campus, 2003.

TEIXEIRA, Pedro Miguel Marinho. **Localização de Emissões de Rádio por SDR**. 2016. Tese (mestrado) – Faculdade de Engenharia Universidade do Porto, Porto, Portugal. Disponível em: < https://repositorio-aberto.up.pt/bitstream/10216/82860/3/119646.pdf>. Acesso em: 5 nov 2019.

TELECO. **Tecnologias de Celular**. 2017. Disponível em: <http://www.teleco.com.br/tecnocel.asp>. Acesso em: 7 nov 2019.

WIKIPEDIA. **LPD433**. Disponível em: <https://en.wikipedia.org/wiki/LPD433>. Acesso em: 8 nov 2019.

WIRELESS Innovation Forum. **Introduction to SDR, [S.l.], [S.d.]**. Disponível em <http://www.wirelessinnovation.org/Introduction_to_SDR>. Acesso em: 5 nov 2019.

XIANJUN, Jiao. **Bluetooth Low Energy (BLE) packet sniffer and generator for both standard and non standard (raw bit)**. Página do *GitHub*. Disponível em:

<https://github.com/JiaoXianjun/BTLE>. Acesso em: 10 nov 2019.