Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

ALESSANDRO FRANCISCO DA SILVA

SOFTWARE PARA PENTEST EM REDES WIFI

ASSIS/SP

2020

Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

ALESSANDRO FRANCISCO DA SILVA

SOFTWARE PARA PENTEST EM REDES WIFI

Projeto de pesquisa apresentado ao Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando: Alessandro Francisco da Silva

Orientador: Prof. Me. Fábio Eder Cardoso

ASSIS/SP

2020

FICHA CATALOGRÁFICA

S586s SILVA, Alessandro Francisco da
Software para pentest em redes wifi / Alessandro Francisco da Silva. – Assis, 2020.
39p.
Trabalho de conclusão do curso (Ciência da Computação). – Fundação Educacional do Município de Assis-FEMA

Orientador: Me. Fábio Éder Cardoso

1.Computer network 2.Security 3.Python

CDD005.8

RESUMO

Com o avanço da tecnologia e o grande número de pessoas utilizando redes sem fio para tarefas do dia a dia, surge a necessidade de uma maior segurança e privacidade nos dados que são trafegados em redes públicas e privadas. Atualmente é possível realizar consultas bancárias através de um celular, pagamentos online, além de conversas em redes sociais. Geralmente as pessoas colocam uma senha muito fraca em sua rede sem fio, facilitando assim a exposição de seus dados pessoais e outras informações para hackers, sendo assim alvo de um ataque cibernético. Neste contexto, a proposta desse trabalho é, demonstrar a fragilidade da segurança das senhas utilizadas por usuários comuns em suas casas, através de testes de vulnerabilidades em redes sem fio, sendo que a aplicação a ser desenvolvida irá quebrar a senha da rede sem fio e derrubar a mesma, demonstrando as estatísticas em forma de gráficos e tabelas para uma melhor visualização do tempo estimado, quantidade de tentativas, e outras informações.

Palavras-chave: Rede de computador, Segurança, Criptografia, Python, Hacker.

ABSTRACT

With the advances of the technology and the large number of people in the world that are using wireless networks for day to day tasks, and with the necessity of a greater security and privacy in data that are used on public and private networks. Currently it is possible to carry out bank consultations via a mobile phone, online payments, in addition to social media conversations. Usually people put a weak password on their wireless network, facilitating the exposition of their personal data and other personal information to hackers, thus being the target of a cyberattack. Knowing this, the proposal of this academic work is to carry out wireless vulnerability testing and shows how weak are the password security that are used for common users in their houses, thus the application will broke the password of the wireless network and bring the network down, showing in graphics the estimated time, number of attempts and other information.

Keywords: Computer Network, Security, Cryptography, Python, Hacker.

LISTA DE ILUSTRAÇÕES

SUMÁRIO

1. INTRODUÇÃO

Com o avanço da tecnologia as redes de computadores têm crescido rapidamente, tendo o seu início a partir dos anos de 1970. A união de computadores em rede é utilizada na área de negócios, na qual acaba englobando propaganda, produção, transporte, elaboração, rendimento e no cálculo. Amplamente utilizada nas instituições educacionais, do ensino fundamental à pós-graduação, possibilitando alunos e professores o acesso imediato a dados em bibliotecas digitais em todo o mundo. Organizações governamentais e militares também utilizam a rede de computadores, na qual as redes de computadores estão sendo utilizadas em diversas áreas no mundo todo. No ano de 1980, a Internet apenas existia algumas dezenas de *sites*, atualmente ela evoluiu e acabou se tornando um sistema de comunicação que abrange milhões de pessoas no mundo todo. (COMER, D. E., 2016, p.3).

A partir do grande crescimento da Internet, é imprescindível que as redes de computadores estejam totalmente seguras, podendo haver falhas de segurança desde o código fonte do projeto até a sua estrutura. Portanto, a importância de testar vulnerabilidades é algo que pode ser realizado por meio de *penetration testing* (teste de penetração), é uma bateria de testes metodológicos que pode ser aplicado em redes de computadores, sistemas operacionais, *websites*, redes sem frio e banco de dados, com o propósito de encontrar e expor todas as possíveis falhas de segurança, sendo possível criar mecanismos de defesa para determinado tipo de aplicação. (MORENO, D., 2019, p.43).

Uma das principais dificuldades desse tipo de teste de intrusão é o uso de falhas conhecidas e previamente divulgadas (a sua grande maioria) e que já existem correções, não sendo possível encontrar vulnerabilidades desconhecidas mesmo utilizando o *Pentest*. Atualmente, pode ser utilizada a linguagem de programação Python, que por ser uma linguagem de programação interpretada e assim não necessitando que seu código fonte seja compilado (convertido para arquivo binário), em conjunto com o sistema operacional Kali que é uma distribuição Linux focada na segurança de computadores. De um modo geral, o objetivo desse

trabalho é identificar possíveis falhas de segurança em redes de computadores e em conjunto com os resultados obtidos promover uma solução para tal vulnerabilidade.

1.1 OBJETIVOS

O objetivo geral deste trabalho acadêmico é aplicar teste de vulnerabilidade para encontrar possíveis falhas e assim podendo ser corrigida, possibilitando uma melhor segurança da infraestrutura das redes de computadores. A partir deste projeto, será possível compreender como uma rede de computador pode ficar exposta a falhas de segurança e como se prevenir de possíveis problemas.

Por meio deste trabalho acadêmico, onde serão implementados *scripts* (conjunto de instruções para que uma função seja executada em determinado aplicativo.) para testes de vulnerabilidades em conjunto com ferramentas do Kali Linux, um sistema operacional Linux tendo como foco o teste de penetração e segurança de redes de computadores (KALI, 2019). Sendo assim, tanto a parte prática e teórica estão divididos da seguinte forma:

- Pesquisar sobre o funcionamento de uma rede de computador.
- Pesquisar e analisar as falhas de segurança que são conhecidas atualmente.
- Pesquisar e analisar as ferramentas do Kali Linux.
- Estudar e elaborar os *scripts* que serão utilizados para os testes de vulnerabilidades.
- Testar as vulnerabilidades em redes de computadores em busca de possíveis falhas de segurança.
- Descrever os resultados obtidos.

1.2 JUSTIFICATIVAS

Tendo em vista o grande avanço das redes de computadores ao redor do mundo, surge a importância da implantação de segurança da informação, necessitando cada vez mais a procura de novas falhas de segurança, tendo em vista que conforme a tecnologia avança também surgem novos problemas. Segundo a revista exame, um tipo de *software* malicioso conhecido como "*Ransomware*", criado há mais de uma década, chegou às manchetes nos últimos meses, depois que criminosos cibernéticos atacaram centenas de milhares de computadores no mundo todo. (EXAME, 2017).

1.3 MOTIVAÇÃO

Devido ao grande aumento do número de pessoas acessando a Internet e realizando suas tarefas utilizando redes de computadores, e tendo em vista que o Brasil pode chegar a um déficit de 75 mil profissionais qualificados em tecnologia, surgindo assim a necessidade de mais investimento e pesquisa na área de segurança de redes de computadores, devido à falta de profissionais com conhecimentos na área e a grande demanda de novos usuários. (EXAME, 2019)

Seguindo esse pensamento, a proteção de informações é algo muito importante para a sociedade moderna, onde dados pessoais são disponibilizados na rede mundial de computadores diariamente, recentemente aproximadamente 2,4 milhões de pacientes do SUS (Sistema Único de Saúde) tiveram seus dados pessoais compartilhados na Internet. (UOL, 2019)

1.4 METODOLOGIA DE PESQUISA

O método que foi utilizado para a realização dessa pesquisa é o qualitativo, sendo uma metodologia de caráter exploratório e com a finalidade de analisar qual o nível de segurança adotado em uma rede sem fio em uma determinada empresa ou família.

Para obter os dados necessários para o estudo foram feitas pesquisas bibliográficas para obter o conhecimento necessário sobre redes de computadores e seus derivados. As obras e estudos analisados foram (VACCA, J., R., Computer and Information Security,2010) e (Borges, Luiz Eduardo. Python para Desenvolvedores: Aborda Python 3.3. Editora Novatec, 2014).

Com caráter descritivo, sendo desenvolvido um software para poder aplicar os testes de penetração e assim analisando quão seguro uma rede sem fio possa estar, e assim levando em consideração o tempo utilizado para encontrar tal falha de segurança.

A linguagem de programação utilizada para o desenvolvimento do sistema foi o *Python*, pelo qual possui bibliotecas para a realização de testes de segurança em redes sem fio e sendo utilizado um processo de automatização do mesmo como forma de facilitar os testes de penetrações.

O primeiro passo é a criação de uma API (Interface de Programação de Aplicativos) em Python que será responsável pelos *scripts* (conjunto de instruções para que uma função seja executada em determinado aplicativo.), responsável pelos testes de vulnerabilidades de redes sem fio. O segundo passo, após o desenvolvimento da API, é desenvolver a interface gráfica

da aplicação utilizando o *React*, uma biblioteca *Javascript* para o desenvolvimento de interfaces de usuário.

A pesquisa de estudo de caso foi feita ao longo de 8 meses, sendo realizados testes de penetrações em redes sem fio de diversos marcas de roteadores e com vários tipos de chave de segurança que são utilizados para proteger uma rede sem fio.

1.5 RECURSOS NECESSÁRIOS

Para o desenvolvimento dessa pesquisa, será necessário a aquisição de uma placa de rede sem fio com suporte a monitoramento de pacotes, além de conhecimentos adquiridos em livros sobre a linguagem de programação Python e redes de computadores.

Além disso, é imprescindível o uso de um celular com suporte a *hotspot móvel*, tendo a capacidade de criar redes sem fio com a criptografia WEP, WPA e WPA2 para poder realizar o teste de penetração.

Entretanto, será necessário uso tanto de hardware e software, citados a seguir:

- ***HARDWARE***
  - Processador AMD Ryzen 5 2600g 3.6GHz.
  - SSD M2 de 480GB.
  - Memória Ram de 3000Mhz 16GB.
  - Placa de vídeo AMD RX580 8GB.
  - Celular Android com suporte a *hotspot* .

- ***SOFTWARE***
  - **Python** - Uma linguagem de programação interpretada, utilizada para o desenvolvimento de *scripts* para a realização dos testes de vulnerabilidades.
  - **React –** Uma biblioteca JavaScript para criar interfaces de usuário.

- **PyCharm** - O PyCharm é um ambiente de desenvolvimento integrado usado em programação de computadores, especificamente para a linguagem Python.
- **Kali Linux** – Uma distribuição Linux baseada no Debian, voltado principalmente para a auditoria e segurança de computadores em geral.

1.6 ESTRUTURA DO TRABALHO

Este projeto está estruturado da seguinte maneira:

- Capítulo 1 - Introdução
  Neste capítulo é abordado uma introdução sobre a origem da internet e o seu grande avanço com o passar dos anos, além de abordar sobre qual a importância de se ter uma rede sem fio mais segura e protegida. Além disso, é abordado sobre o objetivo deste trabalho, citando a importância de se ter uma rede sem fio com um maior nível de segurança, e também entender como uma rede sem fio funciona, explicando a parte técnica do mesmo. Outro ponto importante abordado nesse capítulo é os argumentos que fizeram a escolha do tema para a realização dessa pesquisa, e sobre a importância da segurança em redes de computadores, além de uma explicação sobre as tecnologias *React* e *Python*.

- Capítulo 2 - Rede sem fio
  Neste capítulo é abordado sobre o funcionamento de uma rede sem fio e sobre os tipos de tecnologias que são usadas para a segurança das mesmas, explicando detalhadamente sobre a criptografia WEP, WPA e WPA2.

- Capítulo 3 - Tecnologias

  Neste capítulo é abordado as tecnologias utilizadas para a pesquisa e desenvolvimento do software, explicando detalhadamente cada linguagem de programação.

- Capítulo 4 – Proposta de trabalho
  Neste capítulo é abordado o passo a passo do desenvolvimento do trabalho.

- Capítulo 5 - Conclusão
  Neste capítulo é abordado conclusão parcial do trabalho.

- Referências

  Neste capítulo é abordado as referências bibliográficas utilizadas para o desenvolvimento desse projeto.

2. DEFINIÇÃO DE REDE SEM FIO

Uma rede sem fio é uma tecnologia de rede simples como o infravermelho, Bluetooth, 4G e o Wi-Fi. Entretanto, o foco é sobre o padrão IEEE 802.11 (mais conhecido como o Wi-Fi) (RUFINO, N, M. O., 2015, p.21).

Podendo ter variações no padrão 802.11, como a 802.11a, 802.11b, 802.11g, 802.11n e a 802.11ac. (WEBPOVOA, Póvoa, Thiago C.)

802.11a

Foi um padrão disponibilizado no final do ano de 1999, sendo que sua principal característica é a possibilidade de operar com taxas de transmissão de dados com os seguintes valores: 6

Mb/s, 9 Mb/s, 12 Mb/s, 18 Mb/s, 24 Mb/s, 36 Mb/s, 48 Mb/s e 54 Mb/s, com um alcance de transmissão de aproximadamente 50 metros. (WEBPOVOA, Póvoa, Thiago C.)

802.11b

Em 1999 foi lançada uma atualização do padrão 802.11 que recebeu o nome de 802.11b, contando com uma velocidade de transmissão de dados de 1 Mb/s, 2 Mb/s, 5,5 Mb/s e 11 Mb/s. O seu intervalo de frequências é o mesmo utilizado pelo 802.11 (entre 2,4 GHz e 2,4835 GHz). O padrão 802.11b foi o primeiro a ser adotado em larga escala, sendo, portanto, um dos responsáveis pela popularização das redes Wi-Fi. (WEBPOVOA, Póvoa, Thiago C.)

802.11g

Sendo o sucessor do 802.11b, e disponibilizado em 2003, podendo trabalhar com taxas de transmissão de até 54 Mb/s (megabits por segundo), operando com frequências na faixa de 2,4 GHz (canais de 20 MHz) e possui praticamente o mesmo poder de cobertura do seu antecessor, o padrão 802.11b. (WEBPOVOA, Póvoa, Thiago C.)

802.11n

O 802.11n teve o seu desenvolvimento com início em 2004 e sendo finalizado em 2009, é o sucessor do 802.11g. A principal característica dele é *Multiple-Input Multiple-Output* (MIMO), capaz de aumentar consideravelmente as taxas de transferência de dados por meio da combinação de várias antenas, sendo possível o uso de dois ou mais emissores e receptores para o funcionamento de uma rede. Em relação à sua frequência, o padrão 802.11n pode trabalhar com as faixas de 2,4 GHz e 5 GHz, o que o torna compatível com os padrões anteriores, inclusive com o 802.11a (pelo menos, teoricamente). Cada canal dentro dessas faixas possui, por padrão, largura de 40 MHz. (WEBPOVOA, Póvoa, Thiago C.)

802.11ac

O padrão mais atual é o 802.11ac, tendo o seu desenvolvimento entre os anos de 2011 e 2013, com uma velocidade de até 433 Mb/s (megabits por segundo) no modo mais simples tendo a capacidade de fazer a rede superar a cada dos 6 Gb/s (gigabits por segundo) utilizando múltiplas antenas. (WEBPOVOA, Póvoa, Thiago C.)

2.1 CONCEITO DE PROTOCOLO

Os protocolos são uma forma de padronizar os dispositivos que utilizam as redes Wi-Fi, permitindo assim que diversos tipos de dispositivos se comuniquem entrei si utilizando a mesma tecnologia. As normas e especificações que as fabricantes devem utilizar é gerenciada pelo IEEE (Instituto de Engenheiro Elétricos e Eletrônicos), sendo eles os responsáveis por definir os padrões. Esse padrão é utilizado por qualquer aparelho que utilize o Wi-Fi, independentemente do tipo do dispositivo utilizado.

Com o avanço da tecnologia, a IEEE desenvolve novos padrões que se diferenciam pela sua velocidade, quantidade de antenas que são suportadas e além de outros recursos. (TECHTUDO, 2016).

2.2 DIFERENÇAS ENTRE 802.11N e 802.11AC

Uma das principais diferenças do padrão N para o padrão AC é a sua velocidade, os dispositivos com tecnologia N podem obter uma velocidade de até 450 Mbps, sendo que os com tecnologia AC podem obter uma velocidade de até 1300 Mbps. Essa velocidade é a de transferência entre os dispositivos na rede, sendo diferente da velocidade de Internet contratada pelo provedor. Os aparelhos eletrônicos que utilizam a tecnologia N podem trabalhar com até quatro antenas simultâneas, ao contrário dos que utilizam a tecnologia AC que suportam até oito antenas em paralelo. Outra diferença é que o padrão N opera em 2.4 GHz e o padrão AC opera em 5GHz. Na prática, apesar de oferecer alcance menor, operar em 5 GHz que dizer trabalhar com menos interferências. (TECHTUDO, 2016).

2.3 CANAIS DE REDES SEM FIO

Uma radiofrequência é dividida em intervalos que são normalmente reservados para algum tipo de serviço, que são definidos por convenções internacionais. Sendo que um intervalo é dividido em frequências menores para assim poder permitir uma transmissão em paralelo de

sinais diferentes em cada uma delas. Essas frequências menores são denominadas canais, pelo qual são utilizadas há bastante tempo na sociedade, um exemplo são os canais de rádio AM/FM e os canais de televisão. (RUFINO, N, M. O., 2015, p.22).

A banda 2.4GHz possui 75 canais de comunicação e os dados são enviados por meio desses canais numa forma que aparenta ser aleatória, mas de fato não é, alterando a frequência de transmissão em forma de saltos, uma vez sincronizados estabelecendo assim um canal lógico entre o transmissor e o receptor. (RUFINO, N, M. O., 2015, p.23).

2.4 CONCEITO DE CRIPTOGRAFIA

Quando um ponto de acesso é configurado, existem três opções que podem ser usadas para a segurança da rede, sendo elas a Autenticação aberta, compartilhada e a Rede-EAP. A autenticação aberta qualquer usuário pode se conectar e obter o acesso à rede, a autenticação compartilhada podendo usar chaves WEP e WPA, e por último a Rede-EAP que dão suporte à autenticação com servidores Radius (*Remote Authentication Dial in User Service*, é uma rede de protocolo que fornece Autorização, Autenticação e Contabilidade). (JOBSTRAIBIZER, F., 2010, p.36).

2.5 PROTOCOLO WEP

A WEP foi a primeira criptografia disponível por redes sem fio, podendo ter dois tamanhos: 64-bit e 128-bit. Ela foi implementada baseando-se em chaves compartilhadas e no algoritmo RC4 (algoritmo simétrico de criptografia de fluxo mais usado no software e era utilizado nos protocolos mais conhecidos, como *Secure Socket Layers* e WEP), primeiro ele produz uma soma de verificação do valor, e então ele criptografa utilizando o algoritmo RC4). (VACCA, J., R., Computer and Information Security,2010, p.172).

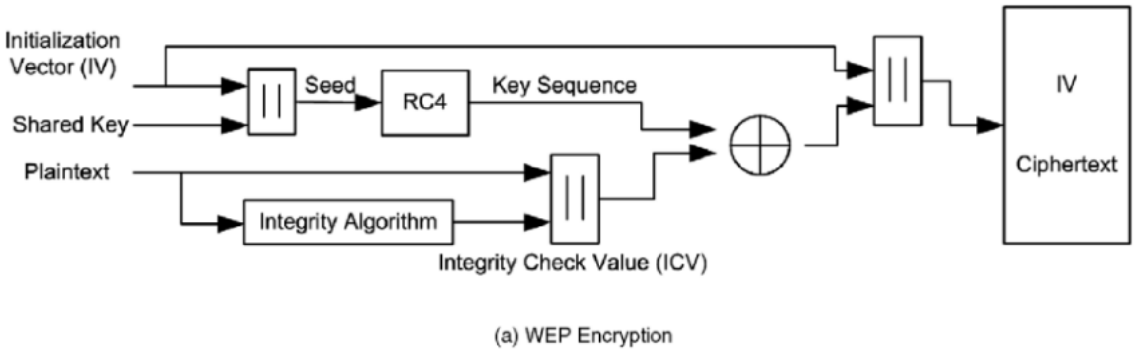

Figura 1: Processo de criptografia com o RC4.
Fonte: Computer and Information Security, maio de 2009.

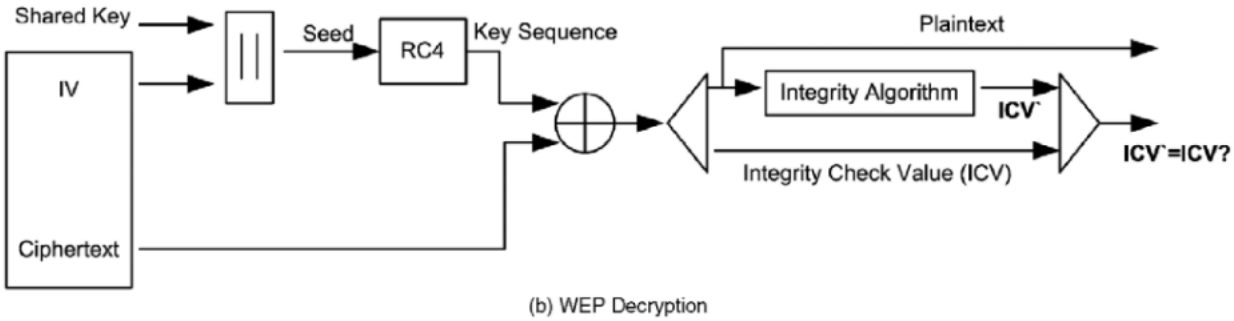

Figura 2: Processo de criptografia com o RC4.
Fonte: Computer and Information Security, maio de 2009.

2.6 PROTOCOLO WPA

Redes Wi-Fi protegidas com o WPA com o padrão da IEEE 802.11i, possui uma segurança mais forte que a WEP. A WPA foi projetada para atender consumidores comuns e empresariais, sendo requerido o uso da autenticação IEEE 802.11x que é responsável por distribuir diferentes chaves para cada usuário. O protocolo WPA funciona de uma maneira similar ao WEP, utilizando um tamanho de 128-bit e um vetor de inicialização com 48-bit,

também oferecendo diversas melhorias se comparado ao WEP, incluindo o TKIP (*Temporal Key Integrity Protocol*) e o MIC (*Message Integrity Code*). Com o uso da TKIP, a WPA vai dinamicamente mudar as chaves periodicamente e como o seu vetor de inicialização é maior, gerando assim uma chave mais segura. (VACCA, J., R., Computer and Information Security,2010, p.172).

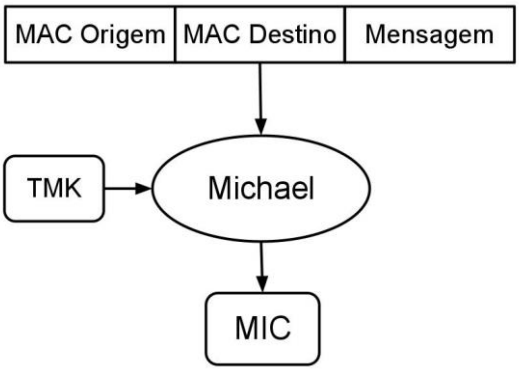

Figura 3: Código de Integridade de Mensagem.
Fonte: Rodrigo R. Paim, UFRJ.

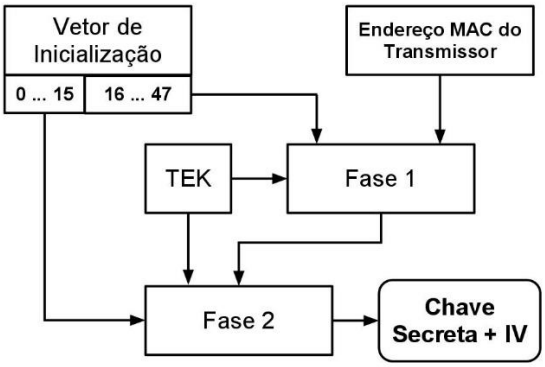

Figura 4: Algoritmo de Mistura de Chaves.
Fonte: Rodrigo R. Paim, UFRJ.

2.7 PROTOCOLO WPA2

Em 2004, com a ratificação do WPA2 sendo conhecido como a segunda geração do WPA e sendo reconhecido como o protocolo mais seguro em redes sem fio. Esse protocolo usa a implementação de 128 bits com o algoritmo AES (*Advanced Encryption Standard*) para o processo de autenticação e criptografia. (ARANA, P., Benefits and Vulnerabilities of Wi-Fi Protected Access 2 (WPA2), 2006, p.1).

A autenticação possui o modo Personal e o modo Enterprise. O modo Personal requere o uso de uma PSK (*Pre-Shared Key*) e não exige que os usuários estejam autenticados separadamente, ao contrário do modo Enterprise onde há a necessidade dos usuários estarem autenticados separadamente por ser baseado na autenticação do protocolo IEEE 802.1X. O WPA2 estabiliza uma conexão segura em quatro fases. Na primeira fase tanto o ponto de acesso e o cliente vão concordar com a política de segurança (método de autenticação) que é suportado tanto pelo ponto de acesso e pelo cliente. Na segunda fase que é aplicado somente no modo Enterprise, a autenticação 802.1X é inicializada entre o ponto de acesso e o cliente, gerando assim uma MK (*common master key*). Na terceira fase depois da autenticação ser realizada, são criadas chaves temporárias que são atualizadas regularmente. Na última fase todas as chaves geradas anteriormente são utilizadas pelo CCMP (Protocolo de Código de Autenticação de Mensagens em Cadeia de Blocos de Cifra no Modo Contado) para fornecer integridade e confiabilidade nos dados (ARANA, P., *Benefits and Vulnerabilities of Wi-Fi Protected Access* 2 (WPA2), 2006, p.2).

O processo de criptografia:

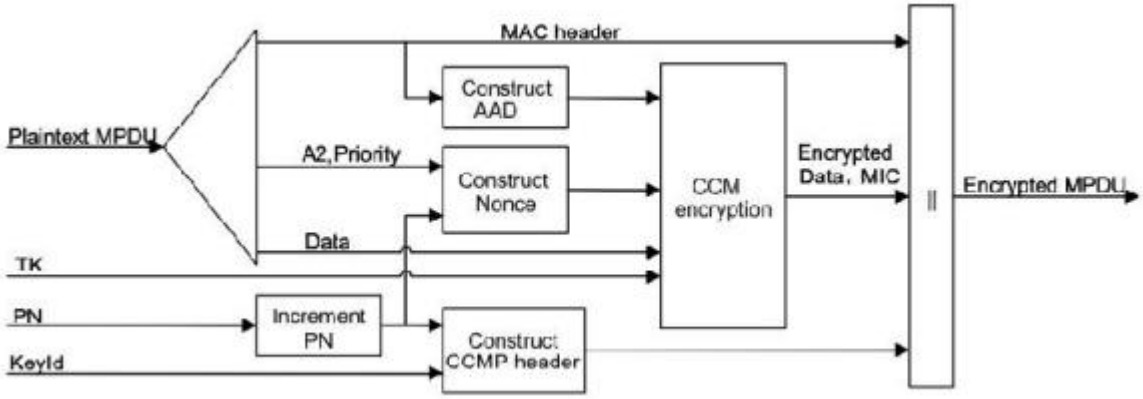

Figura 5: CCMP Encryption Process.

Fonte: Khasawneh, Mahmoud & Kajman, Izadeen & Alkhudaidy, Rashed & Althubyani, Anwar. A Survey on Wi-Fi Protocols: WPA and WPA2.

O processo de decriptografia:

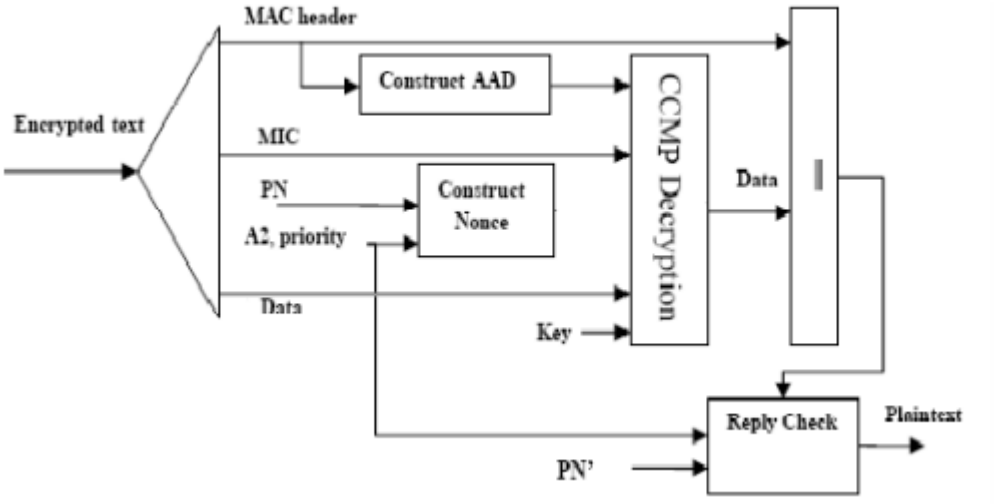

Figura 6: CCMP Decryption Process.

Fonte: Khasawneh, Mahmoud & Kajman, Izadeen & Alkhudaidy, Rashed & Althubyani, Anwar. A Survey on Wi-Fi Protocols: WPA and WPA2.

2.8 PROTOCOLO EAP

O protocolo EAP (*Extensible Authention Protocol*) é um método de conduzir uma conversa de autenticação entre um usuário e um servidor de autenticação, contanto que os dispositivos intermediários tais como os pontos de acessos e os servidores proxy não fazem parte da comunicação. O seu papel principal é retransmitir mensagens EAP no processo de autenticação. (CHAUDHARI, D. N., IEEE 802.11x, and WEP, EAP,WPA / WPA2, p.3).

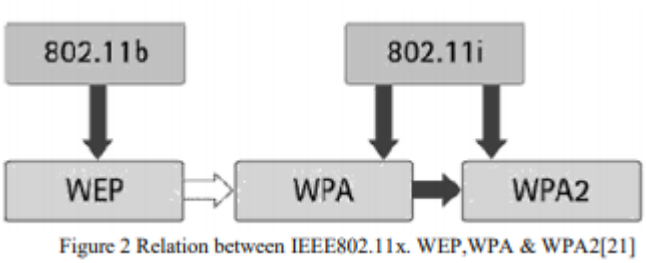

Figure 2 Relation between IEEE802.11x. WEP,WPA & WPA2[21]

Figura 7: Relation between IEEE802.11x. WEP,WPA & WPA2.
Fonte: IEEE 802.11x, and WEP, EAP,WPA / WPA2

3. TECNOLOGIAS

3.1.1 INTRODUÇÃO AO PYTHON

O Python é uma linguagem de alto nível orientada a objeto, com forte tipagem e interpretada. Sendo uma linguagem com uma sintaxe clara, favorecendo a legibilidade do código-fonte, tornando a linguagem mais produtiva e intuitiva. Além de ser utilizado no desenvolvimento de sistemas, o Python também é muito usado para o desenvolvimento de *scripts* permitindo assim automatizar tarefas e adicionar novas funcionalidades em diversos dispositivos. (BORGES, L. E., Python para Desenvolvedores: Aborda Python 3.3, 2014, p.14).

3.1.2 HISTÓRICO

A linguagem foi criada em 1990 no Instituto Nacional de Pesquisa para Matemática e Ciência da Computação da Holanda por Guido Van Rossum, um programador Holandês nascido em 31 de janeiro de 1956, tendo primeiramente o seu foco em usuários como físicos e engenheiros. (BORGES, L. E., Python para Desenvolvedores: Aborda Python 3.3, 2014, p.15).

Atualmente a linguagem é bem aceita na indústria por empresas de alta tecnologia:

- Google
- Yahoo
- Microsoft
- Nokia
- Disney

A versão 3 do Python foi lançada em 2008, quebrando a compatibilidade com versões anteriores com o objetivo de consertar falhas de segurança. (CAELUM, 2018)

3.1.3 SINTAXE

Um programa desenvolvido em Python é constituído por linhas que podem continuar nas linhas seguintes, pelo uso de barra invertida (\) no final da linha ou de parênteses, colchetes ou chaves, em expressões que utilizam os caracteres.

O caractere # marca o início de um comentário, portanto qualquer texto após o # será ignorado até o fim da linha, tendo a exceção dos comentários funcionais.

Exemplo de um comentário funcional:

```
#!/usr/bin/env python
# Uma linha de código que mostra o resultado de 7 vezes 3
print(7 * 3)
```

Figura 8: Exemplo de comentário funcional.

3.1.4 TIPOS DE VARIÁVEIS

As variáveis em Python são criadas através da atribuição e são destruídas pelo coletor de lixo (*garbage collector*), quando não existem mais referências a elas.

Nomes de variáveis em Python devem começar com uma letra sem acentuação ou sublinhado, sendo que maiúsculas e minúsculas sendo consideradas diferentes.

- Números (inteiros, reais, complexos).
- Texto.
- Lista.
- Tupla.
- Dicionário
- Os tipos em Python podem ser:
    - Mutáveis - permitindo assim que o conteúdo da variável seja alterado.
    - Imutáveis - não permitindo que o conteúdo da variável seja alterado.

Em Python, os nomes das variáveis são referências que podem ser alteradas em tempos de execução.

3.2.1 INTRODUÇÃO AO REACT

O React é a biblioteca mais popular do JavaScript e é usada para construir uma interface de usuário. Sendo *open source*, usada para construir *user interfaces* (telas de usuário) nomeadamente para aplicações de página única. O seu principal objetivo é ser rápida, escalável e simples, podendo ainda ser usada em combinação com outras bibliotecas ou

frameworks de JavaScript, como o Angular JS. Referência para quem trabalha com desenvolvimento *frontend*, o JavaScript é a linguagem de programação que mais evoluiu nos últimos tempos, consolidando-se como a mais utilizada pelos desenvolvedores atualmente. (MEDIUM, 2018)

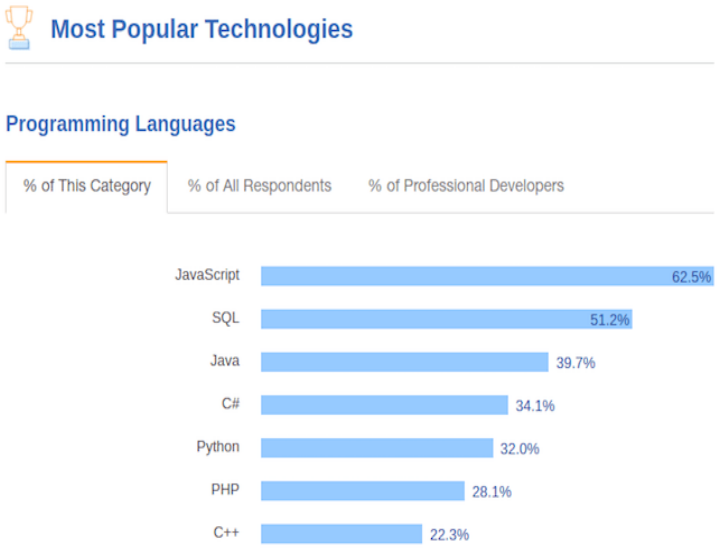

Figura 9: Linguagens mais populares em 2017.
Fonte: StackOverflow Survey, 2017.

3.2.2 HISTÓRICO

Essa biblioteca surgiu em 2011, no Facebook, e passou a ser utilizada na interface do mural de notícias da rede social. No ano seguinte, passou a integrar também a área de tecnologia do Instagram e de várias outras ferramentas da empresa. Em 2013, o código foi aberto para a comunidade, o que colaborou para sua grande popularização. (MEDIUM, 2018)

3.2.3 SINTAXE

O JSX é uma extensão de sintaxe para o JavaScript, baseado no ES6 (ECMAScript 6), sendo a sua mais nova versão. Ele possui uma sintaxe muito semelhante ao HTML. O código abaixo demonstra claramente esta característica. Apesar de muito parecido, o código a seguir não é HTML e sim um trecho de código JSX. (TREINAWEB, 2020)

Exemplo de um código em JSX.

```
const nav = (
  <nav>
    <ul>
      <li><a href="#">Início</a></li>
      <li><a href="#">Sobre</a></li>
      <li><a href="#">Contato</a></li>
    </ul>
  </nav>
)
```

Figura 10: Exemplo de código em JSX.
Fonte: O que é JSX, TreinaWeb.

Porém, o navegador não consegue entender um código em JSX, sendo necessário que o transpilador converta o código para sintaxe do React, o papel do transpilador é converter um código de uma linguagem de programação para outra.

Exemplo de código nativo do React.

```
var nav = React.createElement(
  "nav",
  { className: "links" },
  React.createElement(
    "ul",
    null,
    React.createElement(
      "li",
      null,
      React.createElement(
        "a",
        { href: "#" },
        "Inicio"
      )
    ),
  )
);
```

Figura 11: Exemplo de um código convertido para a sintaxe do React.
Fonte: O que é JSX, TreinaWeb.

3.2.4 COMPONENTES

Um componente em React é utilizado para dividir a UI (*user interface* / interface de usuário) em partes independentes, sendo possível reutilizar o código sem a necessidade de reescrever o mesmo trecho de código diversas vezes, deixando assim a aplicação mais organizável.

De fato, utilizando o React também é possível desenvolver os componentes como *class components* ou funções. (REACTJS, 2020)

Exemplo de um componente no React.

```
class Welcome extends React.Component {
  render() {
    return <h1>Hello, {this.props.name}</h1>;
  }
}
```

Figura 12: Componentes de Função e Classe.
Fonte: Página oficial do React.

Para renderizar um componente na tela do usuário é necessário o uso de uma função nativa do React, que é chamado de *render*.

Exemplo de renderização de componentes no React.

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}

const element = <Welcome name="Sara" />;
ReactDOM.render(
  element,
  document.getElementById('root')
);
```

Figura 13: Renderização de componentes com o React.
Fonte: Página oficial do React

3.2.5 TIPOS DE VARIÁVEIS

As variáveis em React são criadas através da sintaxe do Javascript, podendo ser definida com o uso do VAR, CONST ou LET. Uma variável do tipo VAR, também conhecida como variáveis

de escopo de funções, um escopo em Javascript é dado por funções e não por blocos, sendo assim a palavra-chave VAR dando a garantia que a variável pode ser acessada de qualquer ponto dentro do código, tendo o seu como global. Assim que a ES6 foi lançada, trouxe consigo diversas novidades e entre elas o LET, que é uma palavra-chave com o objetivo de declarar variáveis com escopo de bloco. Se uma variável for definida como LET dentro de uma função, o seu valor só poderá ser acessado dentro desse escopo de código. O CONST também é um recurso novo do ES6, sendo somente leitura, uma variável declarada como CONST não pode ser alterada. (MEDIUM, 2017)

Tipos de variáveis no Javascript.

- Números (inteiros, reais, complexos).
- Texto.
- Matriz.
- Vetor.
- Indefinido.
- Booleano.
- Constante.

Em Javascript, nome das variáveis são *case-sensitive,* significando que nomes com letras maiúsculas são diferentes de nomes com letras minúsculas.

4. PROPOSTA DO TRABALHO

Este trabalho tem como objetivo desenvolver um software para a realização do *Pentest* (Teste de penetração) em redes sem fio, com o objetivo de mostrar o nível de segurança de redes wireless, portanto, utilizando um método de pesquisa do tipo qualitativo e com um estudo de caso baseado na quantidade de pessoas que possuem seus dados pessoais expostos na internet.

A partir disso, foi utilizado um celular com suporte a compartilhamento de redes sem fio, com o mesmo tendo suporte para as principais tecnologias de criptografia existentes no mercado, aumentando assim a oportunidade de realizar do *Pentest* em diferentes tecnologias utilizando apenas um aparelho celular.

Portanto, para o desenvolvimento desse sistema foi necessário entender como funciona uma rede de computador, tendo como principal objetivo saber como funciona a comunicação entre um dispositivo e uma rede sem fio, depois disso foi necessário desenvolver o *sistema de pentest* utilizando o Python, que é uma linguagem de programação que vem crescendo muito nos últimos anos, possibilitando assim a existência de muitas bibliotecas para a realização de testes de penetração. O principal objetivo foi automatizar tal biblioteca, afim de colocar a mesma em uma API (Interface de programação de aplicações), sendo a mesma disponível para ser utilizada por meio de requisições HTTP. Após isso, teve o desenvolvimento da interface gráfica utilizando o React, que é um framework Javascript, com o objetivo de fazer a comunicação do usuário com a API, assim automatizando o processo de *Pentest*.

Por meio dessa interface gráfica, estará disponível todas as redes sem fio disponíveis em uma determinada área utilizando a placa de rede sem fio que possui suporte a monitoramento e injeção de pacotes, a partir disso estará disponível alguns botões na tela com as opções disponíveis para a realização do *Pentest,* bastando apenas um clique para que a API possa realização tal operação de forma automatizada e assim retornado para a interface gráfica o resultado obtido no mesmo.

Exemplo 1 do protótipo do sistema de penteste.

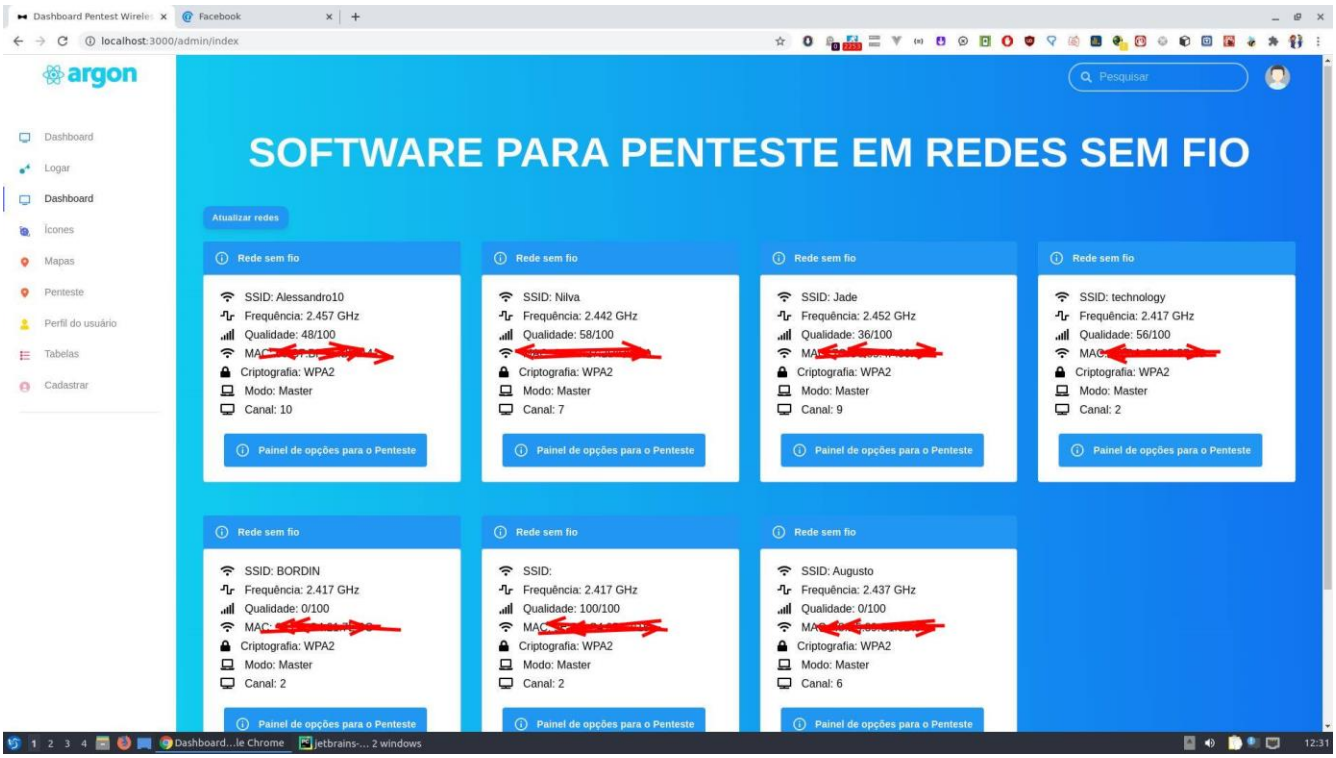

Figura 14: Protótipo do desenvolvimento do sistema.

Fonte: Autor

Exemplo 2 do protótipo do sistema de penteste.

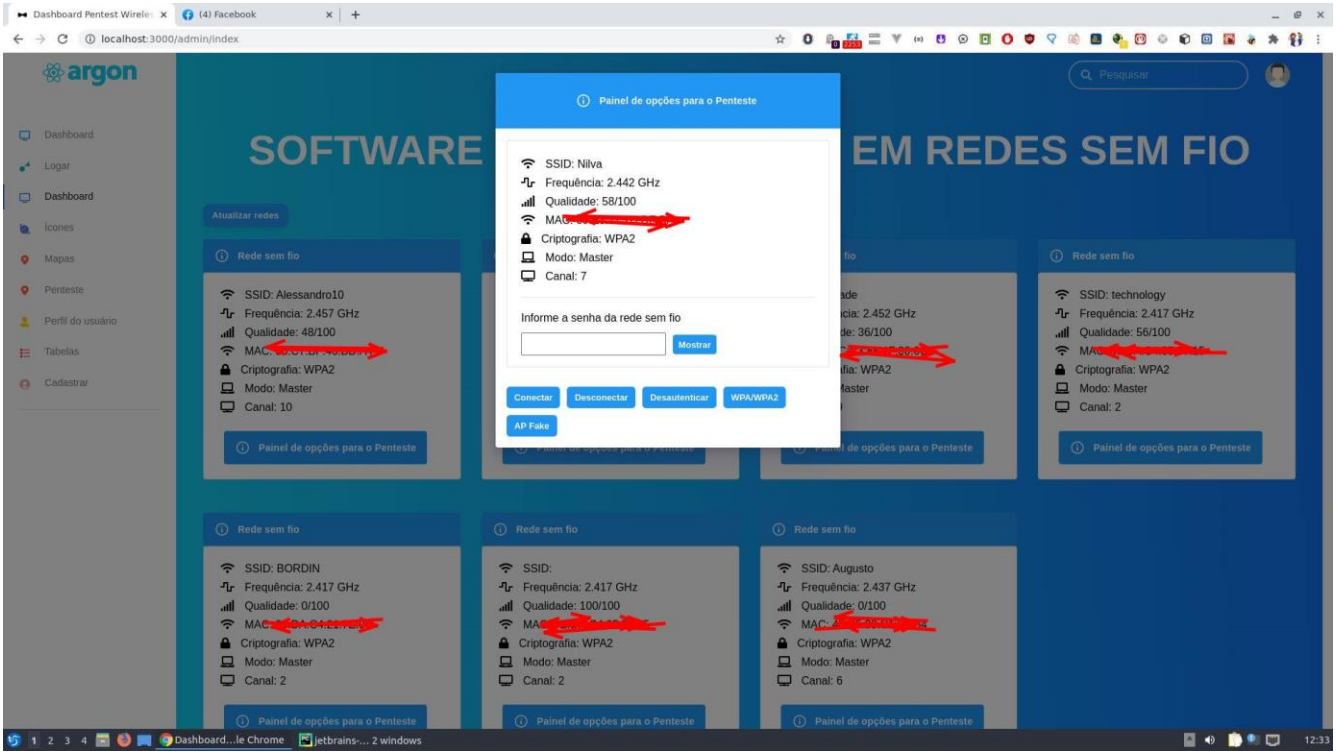

Figura 15: Protótipo do desenvolvimento do sistema.

Fonte: Autor

5. ESTUDO DE CASO

Durante a graduação em computação, eu utilizei o estudo de caso como um dos métodos de meu trabalho de conclusão de curso (*TCC*), que buscou desenvolver um software para a realização de *pentest* redes sem fio. Tendo como o objetivo analisar o nível de segurança adotado de uma rede sem fio através de um sistema desenvolvido principalmente em Python, sendo ele totalmente automatizado para assim poder facilitar a realização do mesmo.

Devido a questão de leis e direitos autorais, essa pesquisa foi aplicada utilizando um aparelho celular com suporte a compartilhamento de internet por meio de *hotspot* móvel, possibilitando assim a criação de redes wireless com diversos tipos de criptografia, permitindo assim que o *pentest* seja realizado a partir apenas de um único local.

Sendo assim, levando em consideração que praticamente quase tudo é feito de modo online, por meio de celulares utilizando uma rede sem fio, gerando assim uma necessidade de se estar protegido evitando assim que dados pessoais do usuário sejam liberados na internet.

A partir disso, será analisado o tempo em relação ao teste de penetração na rede sem fio com base no tipo de senha escolhida pelo usuário, além da exibição de gráficos comparando diferentes tipos de usuários com diferentes tipos de senhas.

Com relação ao processo de validação dos resultados, foi feita uma apresentação mostrando a realização dos testes de penetração em redes sem fio, sendo tudo documentado e armazenado em um gráfico para eventuais consultados e comparações.

6. CONCLUSÃO

Tendo em vista que a segurança de informações pessoais é algo que se cada vez deve ter mais cuidado, devido ao aumento da tecnologia atual e da comodidade de se realizar tarefas do dia a dia utilizando um aparelho celular e uma rede sem fio. Atualmente essas tarefas são transações bancarias, conversa em redes sociais e até mesmo pesquisas utilizando um motor de busca, tendo um exemplo sendo o Google. Sendo assim, ter uma boa segurança em redes sem fio é algo de extrema importância, garantindo assim uma segurança para quem está utilizando.

Utilizando a linguagem de programação Python para o desenvolvimento de *scripts*, como o objetivo de realizar testes de penetração, em busca de falhas e mostrar o nível de segurança utilizado em redes sem fio. Um dos fatores que contribuem é devido a utilização de senhas com poucos caracteres e utilizar um protocolo de criptografia desatualizado.

Após a conclusão do desenvolvimento de trabalho, tendo o intuito é deixar uma base de estudos utilizando os conceitos de redes de computadores em conjunto com o Python.

REFERÊNCIAS

ARANA, Paul. **Benefits and Vulnerabilities of Wi-Fi Protected Access 2 (WPA2).** Disponível em

<https://dl.irstu.com/wpcontent/uploads/Download/Education/Book/Network/Network%20Security/WEP-WPA-Article/Benefits%20and%20Vulnerabilities%20of%20Wi-Fi%20Protected%20Access%202%20(WPA2).pdf>. Acesso em 19/02/2020.

Borges, Luiz Eduardo. **Python para Desenvolvedores: Aborda Python 3.3.** Editora Novatec, 2014.

CAELUM. **Quais as diferenças entre Python 2 e Python 3.** Disponível em <https://blog.caelum.com.br/quais-as-diferencas-entre-python-2-e-python-3/>. Acesso em 05/03/2020.

COMER, D. E. Redes de Computadores e Internet. 2016. Rio Grande do Sul, Porto Alegre, Brasil.

EXAME. **Brasil pode chegar a um déficit de 750 mil profissionais qualificados em tecnologia.** São Paulo. Disponível em

<https://exame.abril.com.br/negocios/dino/brasil-pode-chegar-a-um-deficit-de-750-mil-profissionais-qualificados-em-tecnologia/>. Acesso em 05/11/2019.

EXAME. **Como se proteger de ataques virtuais.** São Paulo. Disponível em

<https://exame.abril.com.br/tecnologia/como-se-proteger-de-ataques-virtuais/>. Acesso em 05/11/2019.

HURLEY, Chris; ROGERS, Russ; THORNTON, Frank; BAKER, Brian.

JOBSTRAIBIZER, Flávia. **Desvendando as redes sem fio**. Digerati Books, 2010.

Khasawneh, Mahmoud & Kajman, Izadeen & Alkhudaidy, Rashed & Althubyani, Anwar. (2014). A Survey on Wi-Fi Protocols: WPA and WPA2. 420. 496-511. 10.1007/978-3-642-54525-2_44.

MCCLURE, Stuart; SCAMBRAY, Joel; KURTZ, George. **Hackers Expostos - Segredos e soluções para a Segurança de Redes**. Editora Bookman, 2014.

MEDIUM. **React: o que é e como funciona essa ferramenta**. Disponível em

<https://medium.com/reactbrasil/react-o-que-%C3%A9-e-como-funciona-essa-ferramenta-319922a8371c>. Acesso em 23/03/2020.

MEDIUM. **Variáveis em ES6 — Var, let e const. Como funcionam e qual delas usar**. Disponível em

<https://medium.com/@raphalima8/vari%C3%A1veis-em-es6-var-let-e-const-como-funcionam-e-qual-delas-usar-413938f732f9>. Acesso em 23/03/2020.

MORENO, Daniel. Introdução ao PENTESTE. 2019. São Paulo, São Paulo, Brasil.

PAIM, Rodrigo. **WEP, WAP e EAP**. Disponível em
<https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2011_2/rodrigo_paim/wpa.htm>

Acesso em 25/02/2020.

PARRA, Henrique. **Controle social e prática hacker: tecnopolítica e ciberpolítica em redes digitais.** São Paulo. Disponível em

<https://www.redalyc.org/pdf/703/70324609011.pdf>. Acesso em 16/02/2020.

REACTJS. **Componentes e Props**. Disponível em

<https://pt-br.reactjs.org/docs/components-and-props.html>. Acesso em 22/03/2020.

ROCKETSEAT. **React do zero: componentização, propriedades e estado**. Disponível em

<https://blog.rocketseat.com.br/react-do-zero-componentizacao-propriedades-e-estado/>. Acesso em 22/03/2020.

RUFINO, Nelson Murilo de O. **Segurança em Redes sem Fio**. Editora Novatec, 2015.

RUMALE, Aniruddha S; CHOUDHARY, Dr D. N. **IEEE 802.11x, and WEP, EAP, WPA / WPA2**. Disponível em <https://pdfs.semanticscholar.org/5916/14ae7263e980fcea0b873010fe1a8d3af1b4.pdf>

Acesso em 16/02/2020.

TECHTUDO. **802.11ac e 802.11n: veja diferenças entre padrões da performance Wi-Fi.** Disponível em <https://www.techtudo.com.br/noticias/noticia/2016/09/80211ac-e-80211n-veja-diferencas-entre-padroes-da-performance-wi-fi.html

Acesso em 16/02/2020.

TREINAWEB. **O que é JSX**. Disponível em

<https://www.treinaweb.com.br/blog/o-que-e-jsx/>. Acesso em 02/04/2020.

UOL. **Dados pessoais de 2,4 milhões de usuários do SUS são vazados na internet.** São Paulo. Disponível em <https://www.uol.com.br/tilt/noticias/redacao/2019/04/11/dados-pessoais-de-24-milhoes-de-usuarios-do-sus-sao-vazados-na-internet.htm>. Acesso em 03/11/2019.

VACCA, John R. **Computer and Information Security Handbook.** Editora Elsevier, 2009.

WarDriving and Wireless Penetration Testing. Editora Syngress, 2007.

WEBPOVOA. **Entenda os principais padrões IEEE 802.11**. Disponível em <https://webpovoa.com/entenda-os-principais-padroes-ieee-802-11/>. Acesso em 02/03/2020.