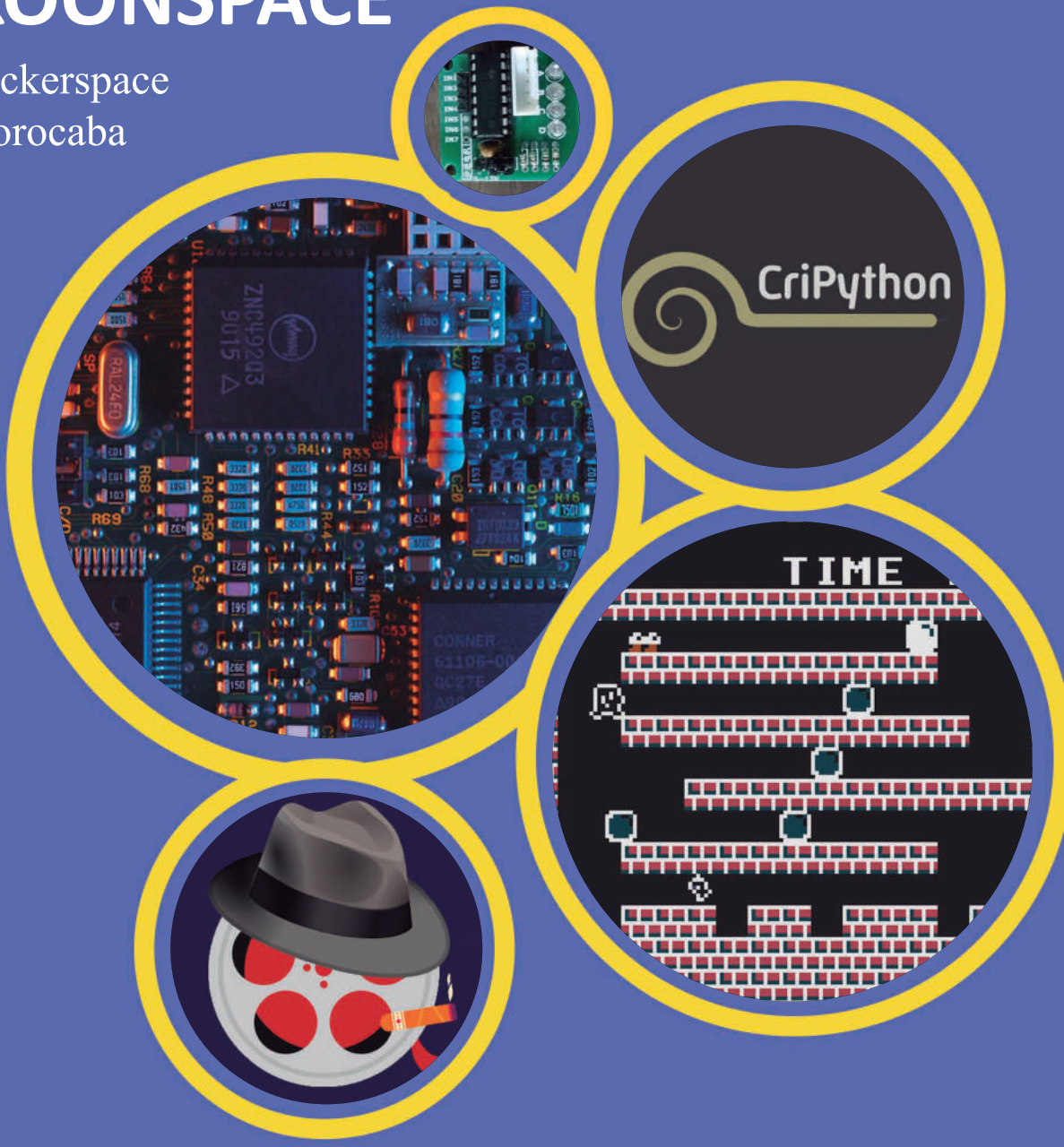
VOL. 2
2020

Revista HACKOONSPACE

Projeto Hackerspace
UFSCar Sorocaba



Apresentação dos artigos e projetos realizados na edição 2020 do HackoonSpace

Universidade Federal de São Carlos
Campus Sorocaba

Revista Hackerspace

Vol. 2

Projeto Hackerspace
2020

Apresentação

O Projeto Hackerspace é um projeto de extensão promovido pela UFSCar Campus Sorocaba e realizado na forma de encontros que discutem a contracultura hacker, apresentando personagens, acontecimentos, aspectos socioculturais, artefatos e atividades com a finalidade de que os participantes sejam expostos à contracultura em questão e desenvolvam um projeto ou material acerca da mesma. A principal intenção do projeto é desmistificar o conceito e figura do hacker, enquanto proporcionando um espaço de exposição, produção e compartilhamento de conteúdo que se configura como dentro da contracultura de forma que participantes de diferentes níveis de conhecimento técnico possam participar.
Esta revista tem como objetivo divulgar os trabalhos desenvolvidos pelos alunos participantes do Projeto Hackerspace no ano de 2020. Esperamos que esta revista possibilite a difusão dos conhecimentos adquiridos na atividade para a comunidade em geral. Gostaríamos de agradecer todos os alunos que contribuíram com artigos e projetos para esta edição da revista.

Organização e edição:
Bruno Frítoli Carraza
Gustavo de Jesus Rodrigues
Jade Manzur de Almeida
Marcus Vinicius Natrielli Garcia

Supervisão:
Gustavo M. D. Vieira

Conteúdo

Parte I

Artigos

Capítulo 1

Deauther Wi-Fi - Prova de Conceito

Guilherme Brante Cavequia
guilherme.cavequia@gmail.com
Guilherme Milani Oliveira
gui.milani@hotmail.com
Lucca Marques Almeida Paes Teles
lucca.marques08@gmail.com

Resumo

Este documento disserta sobre o processo de concepção de um Deauther Wi-Fi e sua importância para o universo da computação, ou mais especificamente, suas influências no campo de ação da segurança de redes. Entender a fragilidade de uma rede, seja ela pessoal ou corporativa e sendo o usuário comum ou profissional da área de segurança da informação, faz parte do entendimento das limitações da privacidade humana em meio a uma evolução formidável nos sistemas de informação contemporâneos.

Palavras-chave: Computação; Redes; Segurança; Deauther; NodeMCU; ESP8266; Wi-Fi.

1.1 Introdução

O projeto consiste em utilizar técnicas e tecnologias já desenvolvidas para construir um dispositivo que, basicamente, desconecta todos os usuários de uma rede Wi-Fi alvo enviando pacotes especiais de rede que cancelam a conexão de seus aparelhos repetidas vezes. Para agilizar o processo, iremos nos referir a esse dispositivo pelo seu nome técnico, um *deauther* ("desautenticador" na tradução literal do inglês).

A ideia surgiu inicialmente após alguns integrantes se depararem com o repositório do código utilizado e manipular redes dessa maneira deixou o grupo intrigado com o assunto.

É buscado com esse projeto uma prova de conceito, experimentando os códigos e componentes com a finalidade de gerar um deauther funcional ao final do processo.

1.2 Requisitos

Placa NODEMCU ESP8266

A ESP8266 é um microchip de baixo custo produzido pela Espressif com um módulo que permite suporte a TCP/IP (conjunto de protocolos padrões das redes Wi-Fi atuais), o NodeMCU é um firmware baseado em Lua desenvolvido para rodar nesse mesmo chip. O hardware e o firmware são ambos open-source e essa característica alinhada ao fato da placa ter um módulo Wi-Fi, torna ela uma escolha muito procurada para quem está começando a investigar o tipo de exploit em redes que será explicitado adiante.

1.2. ESP8266 DEAUTHER 2.1.0

O software "ESP8266 Deauther 2.1.0" possibilita a fácil execução de uma variedade de procedimentos para experimentar através de testes as redes sem fio usando um SoC (System On a Chip) Wi-Fi ESP8266.

O controle de alguns dos frames de gerenciamento de redes wireless, responsáveis por con-

Figura 1.1: Placa NodeMCU ESP8266 [13]

trolar a entrada e saída das estações (stations) ou STA, também pode ser feito dentro da aplicação. As estações são basicamente os dispositivos conectados a um AP (Access Point), sendo este último qualquer roteador convencional que estabelece a comunicação da rede Wi-Fi com outras redes (internet, redes cabeadas, etc.), podendo agregar a função de roteador.

Os frames MAC manipulados pelo software são baseados nos protocolos WLAN do IEEE 802.11. Um frame MAC é construído com campos comuns presentes em todos os tipos de frames e outros campos um pouco mais específicos, que denotam o tipo e o subtipo característico do frame. O Campo de Controle de Frame (Frame Control Field) é o domínio que registra o tipo e o subtipo do frame MAC, além de registrar outras informações relevantes como a versão do protocolo.

Dessa maneira, os frames MAC podem variar de acordo com quatro tipos: Gestão, Controle, Dados e Extensão (Management, Control, Data e Extension). Os frames de gestão incorporam requisições de associação e autenticação na rede e vão ser afetados nos ataques do dispositivo.

Dentre os frames de gerenciamento (Management Frames) que a interface permite manipular, pode-se destacar os subtipos: beacon frames e probe frames. O primeiro corresponde aos avisos que as estações enviam periodicamente para exteriorizar que a rede está operando, dessa maneira, redes Wi-Fi podem ser identificadas por outros APs nas redondezas. Já os probe frames são fundamentalmente utilizados pelas estações que buscam saber se um AP ainda está operacional.

O programa é licenciado sob a Licença MIT. Dessa maneira, os desenvolvedores possuem discernimento para com as fragilidades das quais o dispositivo explora, as quais residem em roteadores modernos que ainda são baseados no IEEE 802.11, protocolo de rede padrão usado costumeiramente após WEP.

Além disso, os programadores responsáveis conscientizam sobre a não utilização do termo *jammer* para difundir os princípios do projeto, que visam substancialmente, aprendizado e provas de conceito, além é claro, da elucidação dos problemas de segurança que um simples dispositivo pode trazer.

1.3 Experiências e Resultados

Iremos abordar agora todo o procedimento desde a aquisição da placa até seu funcionamento esperado.

Passo a passo

A placa é encontrada em um preço médio de 25 reais. Adquirimos uma e logo após tentamos gravar na unidade removível do dispositivo o arquivo binário do deauther fornecido no repositório, especificamente o release v2.1.0. É importante salientar que antes de realizar este passo, é necessário se certificar de ter um cabo usb que, além de alimentar a placa com 5V, também transmita informação e o driver certo da placa tenha sido instalado na sua máquina. Para gravar com sucesso utilizamos o programa *ESP8266Flasher.exe*.

Após isso, apenas ao reiniciar o fornecimento de energia, a placa começa a funcionar com o deauther e o primeiro passo para trabalhar com ele é verificar as redes Wi-Fi disponíveis a partir de outro dispositivo, na listagem irá aparecer o SSID (*service set identifier* é o nome de uma determinada rede como é exibida para os seus usuários) *pwned* por padrão, com a senha *deauther*. Devemos selecioná-lo com o dispositivo que vai comandar o ataque.

Ao acessar o endereço do gateway padrão do NodeMCU no navegador, é exibida a interface gráfica para o usuário do deauther, que é customizável também pois o programa é open-source.

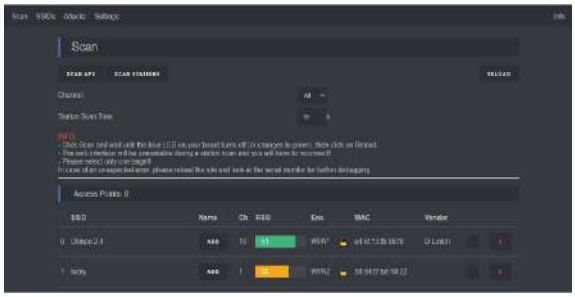


Figura 1.2: Tela onde é realizado o scan de redes próximas [autoria própria]

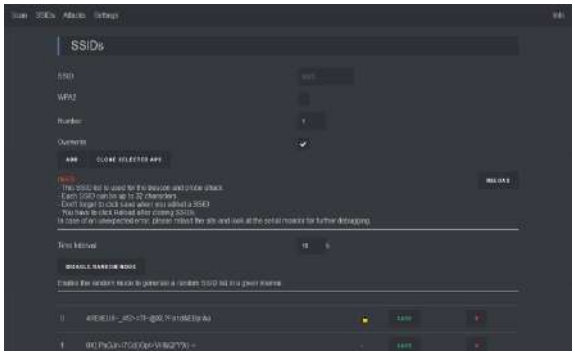


Figura 1.3: Tela onde os SSIDs ficam salvos para ataques Beacon e Probe [autoria própria]

Figura 1.4: Tela de utilização dos ataques [autoria própria]

A interface é bem simples, temos uma tela para escanear as redes próximas, salvá-las e organizá-las, outra para inserir os SSIDs utilizados no ataque beacon e finalmente a tela para realizar os ataques propriamente ditos.

Tipos de ataques

Os ataques que integram o dispositivo são três:

Deauther attack

Este ataque é o foco do projeto e é possível de ser realizado em redes que tem seus roteadores na frequência 2.4Hz e baseados no protocolo IEEE 802.11, o qual permite que os pontos de acesso (roteadores) enviem frames de desautenticação para as estações (dispositivos conectados à rede sem fio), avisando os mesmos que foram desconectados.

O dispositivo que realiza o ataque, envia para o ponto de acesso um frame de desautenticação com o endereço MAC das estações, enganando e forçando ele a enviar, também, um frame de desautenticação para as estações, que por esse motivo acabam sendo desconectadas. O endereço MAC das estações é obtido interceptando o tráfego da rede local e analisando seus pacotes.

Dentro do período de tempo especificado na interface do programa, o dispositivo envia esses frames constantemente, não permitindo que os usuários se conectem novamente à rede.

É interessante mencionar que esse é um tipo de ataque DoS (Denial of Service).

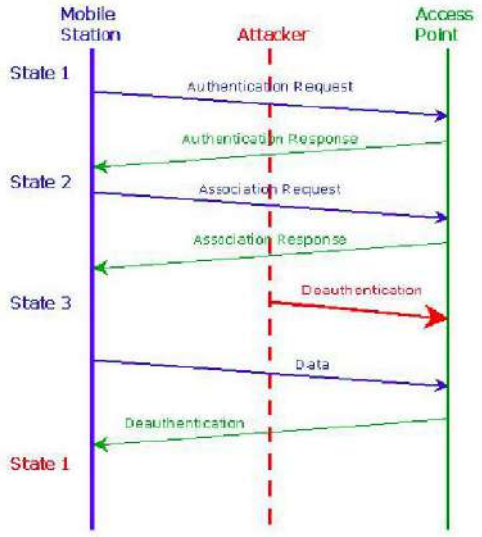


Figura 1.5: Processo de desautenticação ilustrado [12]

Dentro da implementação desse ataque, podemos identificar algumas características importantes de se comentar:

Na função responsável pela maior parte do ataque existe uma diferenciação nos métodos chamados caso o pacote seja enviado para as estações ou para os pontos de acesso, pois o tamanho dos mesmos pode variar. Também nela são enviados os frames de desautenticação e, logo após, os frames de desassociação.

Ambos são enviados tanto para as estações quanto para os APs. Em todos esses casos, a função responsável por enviar de fato os pacotes é a wifi_send_pkt_freedom, que foi descontinuada em versões futuras do SDK da Espressif para a placa ESP8266 devido às potenciais brechas permitidas. Por esse motivo, usar o SDK na versão certa é essencial para realizar o projeto.

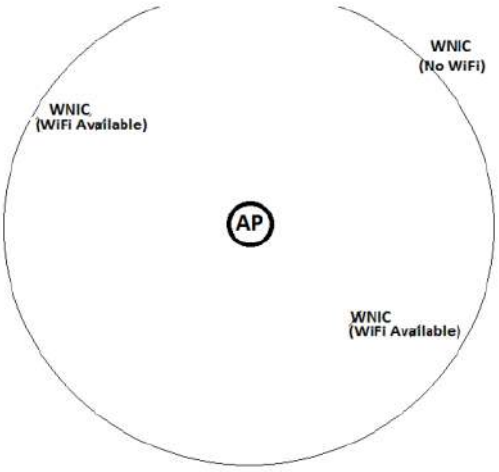


Figura 1.6: - Funcionalidade dos beacon frames a partir do AP [10]

Beacon attack

Como aludido anteriormente, os beacon frames fazem parte do frame de gerenciamento das WLANs baseadas no IEEE 802.11 e basicamente contém todas as informações da rede. Esses frames são transmitidos periodicamente, servindo para anunciar a presença de uma rede sem fio e sincronizar os pontos de acesso. Dessa forma, os APs que transmitem os beacon frames e os mesmos são propagados nos padrões da infraestrutura de rede da IEEE 802.11 para as STAs.

Para os padrões de redes 2.4 GHz, quando mais de quinze SSIDs estão em canais sobrepostos ou existem mais de quarenta e cinco no total, os beacon frames começam a consumir uma quantidade significativa de tempo, influenciando o desempenho do AP mesmo quando a maioria das redes estão ociosas.

Assim, o beacon attack é outro tipo de ataque DoS onde o dispositivo envia diversos frames falsos de redes, distraindo o usuário, pois são tantas redes que ele dificilmente encontrará a que está buscando e ficará perdido numa enorme lista de SSIDs.

Normalmente essa abordagem é usada em conjunto com outras técnicas para ludibriar quem quer que esteja tentando se conectar, por exemplo, desconectando o usuário da rede que estava anteriormente com um ataque deauther, enviando múltiplas redes falsas com o beacon para confundi-lo e após isso criar uma rede armadilha com o SSID da rede que ele buscava. Quando a vítima se conectar, todos seus dados e tráfego na internet poderão ser analisados e manipulados pelo responsável do ataque.

Probe attack

Dentre os outros ataques citados, este é o mais simples e interage com os probe frames. Esses frames também são de gerenciamento e podem ser de dois tipos: frame de probe request ou frame de probe response. O primeiro se refere às requisições de informação enviadas de uma estação, já o segundo é enviado de um ponto de acesso e contém suas informações de capacidade, taxas de dados suportadas, etc., depois de receber um frame de probe request.

Sendo assim, o probe attack encaminha diversas solicitações para os APs, gerando tráfego na rede. Sua manifestação pode acontecer quando a estação incorporada à rede sugere conexão com o AP. Contudo, não tem efeitos destrutivos e não gera problemas.

Resultado

Ao realizar o procedimento descrito, foi possível desconectar todas as estações conectadas na rede de teste, além de termos gerado ao mesmo tempo

60 redes falsas com SSID de nossa escolha e efetuar uma probe request com sucesso. É importante notar que as redes falsas geradas tinham mais de 30 metros de alcance, muito mais do que redes Wi-Fi comuns.

1.4 Contramedidas

Para se defender completamente do ataque Deauther, apenas tendo um roteador que suporte os padrões IEEE 802.11w já é necessário. Porém nem todas as redes estão atualizadas com essa emenda do protocolo 802.11.

Conhecida também como Protected Management Frames, a medida cabe apenas aos frames de gerenciamento sendo aparentemente boa para alguns ataques. Entretanto, existem algumas limitações de infraestrutura e dentre os frames que essa extensão pode proteger, podemos destacar os frames de dissociação (Disassociate Frames) e os frames de "desautenticação" (Deauthenticate Frames).

Consequentemente, o deauther attack, principal ataque abordado nesse artigo, pode ser prevenido por essa emenda do protocolo IEEE 802.11, mas os outros ataques não tão prejudiciais, que envolvem o beacon frame e os dois tipos de probe frame, ainda podem ser explorados na IEEE 802.11w.

1.5 Considerações finais

A elaboração desse projeto e todos os estudos e concepções que abrangem a segurança da informação (confidencialidade, integridade, etc.), tornou mais elucidativo o entendimento das limitações de uma rede Wi-Fi e, sendo ainda mais preciso, as barreiras que cercam a privacidade digital das pessoas em qualquer âmbito, seja este doméstico ou corporativo e sejam essas comuns ou com conhecimentos aplicados em segurança da informação.

Entender as implicações que um simples dispositivo como o NodeMCU ESP8266 pode ocasionar, difundem as possibilidades de infrações que outras tecnologias mais ou menos complexas também podem causar. Tudo isso, torna perceptível o prejuízo que a constante evolução da tecnologia alinhada à desinformação pode trazer ao cotidiano de usuários comuns.

Contudo, essa argumentação também instaura o aperfeiçoamento dos princípios da segurança da informação. Garantir que qualquer usuário ao redor do mundo esteja seguro ao se conectar a qualquer rede sem fio é definitivamente uma tarefa árdua, já que existem fatores externos que não estão diretamente relacionados com a proteção das informações, como o conhecimento do próprio usuário e que contribuem para a violação das informações.

De qualquer forma, a tentativa de assegurar os dados do usuário é válida mesmo com essas adversidades citadas, devendo sempre ser uma prioridade. As facilidades que permeiam a manipulação do NodeMCU ESP8266 devem ser exploradas didaticamente e/ou usadas para consolidar provas de conceito com o intuito de enriquecer o conhecimento acerca das fragilidades dos sistemas, como aconteceu nesse projeto.

1.6 Bibliografia

[1] NICOLAS DARCHIS. A starter guide to learn wireless sniffer traces. [Acessado em: 29/01/2021.]. URL: https://community.cisco.com/t5/wireless-mobility-documents/802-11-frames-a-starter-guide-to-learn-wireless-sniffer-traces/ta-p/3110019#toc-hId--1447989924.

[2] AJUDA DIGITAL. Criando um esp8266 deauther. [Acessado em: 17/11/2020.]. URL: http://ajudadigital.com.br/index.php/windows/146-criando-um-esp8266-deauther.

[3] HACKDAY. Nodemcu lua firmware. [Acessado em: 17/11/2020.]. URL: https://hackaday.io/project/3465-playing-with-esp8266/log/11449-nodemcu-lua-firmware.

[4] IEEE.ORG. Status of project ieee 802.11 task group w. [Acessado em: 29/01/2021.]. URL: https://grouper.ieee.org/groups/802/11/Reports/tgw_update.htm.

[5] STEFAN KREMSER. Affordable wifi hacking platform for testing and learning. [Acessado em: 24/09/2020.].

URL: https://github.com/SpacehuhnTech/esp8266_deauther/.

[6] STEFAN KREMSER. Mit licence. [Acessado em: 24/09/2020.]. URL: https://github.com/SpacehuhnTech/esp8266_deauther/blob/v2/LICENSE.

[7] STEFAN KREMSER. Releases. [Acessado em: 24/09/2020.]. URL: https://github.com/SpacehuhnTech/esp8266_deauther/releases.

[8] LUA.ORG. The programming language lua. [Acessado em: 17/11/2020.]. URL: http://www.lua.org/.

[9] NODEMCU. Lua based interactive firmware for esp8266, esp8285 and esp3. [Acessado em: 17/11/2020.]. URL: https://github.com/nodemcu/nodemcu-firmware.

[10] perfil de criador do Wikimedia Pionita8. Funcionalidade dos beacon frames a partir do ap.

[11] ESPRESSIF SYSTEMS. Esp8266 wi-fi mcu. [Acessado em: 17/11/2020.]. URL: https://www.espressif.com/en/products/socs/esp8266.

[12] Boon Hwed Tan. Defending ieee 802.11-based networks against denial of service attacks.

[13] WIKIPEDIA. Nodemcu esp8266. URL: https://pt.wikipedia.org/wiki/ESP8266#/media/Ficheiro:NodeMCU_DEVKIT_1.0.jpg.

Capítulo 2

Hackers em ação: Problemas em Época de Pandemia Contra Ataques Cibernéticos

Leandro Naidhig

Resumo

Este artigo apresenta uma visão sobre os ataques cibernéticos em uma pandemia, descrevendo um pouco sobre o conceito hacker, seus principais tipos, como esses ataques ocorrem, principais alvos dos ataques, prevenção de dados, notícias, investimentos das empresas e ataques a diversas pequisas científicas para encontrar uma vacina por todo o mundo.

Palavras-chave: Covid-19; Ataques Cibernéticos; Hackers.

2.1 Introdução

Com o aparecimento de um vírus denominado Sars-CoV-2 (Vírus da família Coronaviridae), o mundo acabou presenciando uma das maiores pandemias enfrentada pela humanidade, assim, hackers aproveitam o atual momento em que as pessoas precisam ficar em suas casas para descobrir vulnerabilidades em sistemas para roubar informações e dados para realização de golpes virtuais. Os ataques praticados por esses indivíduos são bem diversificados e normalmente possuem focos em sistemas com alto índice de uso por parte da população, como é os casos de chamadas de videoconferência ou até mesmos pesquisas científicas para encontrar uma vacina para o atual vírus. Um caso recente das vulnerabilidades de segurança é da empresa ZOOM, em que antes do isolamento social praticamente ninguém conhecia quase sua aplicação, porém logo com o aumento do home-office, cada vez mais pessoas procuraram ferramentas para reuniões e interações sociais de forma virtual para continuar seus serviços. Logo, essa popularidade e crescimento são alvos perfeitos para encontrar diversas brechas nessas aplicações.

2.2 Conceito de Hacker

O conceito de hacker era originalmente utilizado para referir pessoas com um alto conhecimento em programação, com maior foco em segurança de dados, sem fins comerciais e acesso não autorizado de informações. Porém, com o passar do tempo esse conceito foi sendo modelado para indivíduos com motivações para criminalidade digital, sendo inserido nos mais diversos níveis de segurança para quebra de software, violando toda a privacidade e sigilo [7].

2.3 Tipo de Hackers

Existem diversos tipos de hackers, muitos deles poucos conhecidos nas comunidades existentes. Eles geralmente são pertencentes ás principais seis categorias abaixo:

- *A. Hackers de Chapéu Branco (White Hat Hackers)*

São considerados hackers éticos, aqueles com autorização ou certificação para trabalharem em testes de penetração e observando brechas de segurança em sistemas. Trabalham sob regras e são especialistas na área de segurança cibernética. [1] [6]

- *B. Hackers de Chapéu Preto (Black Hat Hackers)*

  São as pessoas mais perigosas, sendo denominadas crackers, em que acessam dados de forma não autorizada de sistemas e destroem ou roubam essas informações para benefício próprio. São considerados criminosos e normalmente possuem alto conhecimento em programação. [1] [6]

- *C. Hackers de Chapéu Cinza (Gray Hat Hackers)*

  São hackers que ficam 'em cima do muro', muitas vezes não sabemos suas reais intenções, normalmente visando seus próprios desejos como roubar propriedades intelectuais ou apenas tratam o hacking como um hobby. [1] [6]

- *D. Hackers de Chapéu Azul (Blue Hat Hackers)*

  Os chapéus azuis são hackers que operam fora das empresas de segurança e têm a tarefa de atacar softwares ou sistemas antes do seu lançamento. A Microsoft usa os chapéus azuis extensivamente para proteger os seus produtos e até mesmo realiza uma competição com prêmios em dinheiro. [1] [3] [6]

- *E. Hackers de Chapéu Verde (Green Hat Hackers)*

  São iniciantes na área hacking, possuem forte ambição para aprender e melhorar suas habilidades, porém podem causar problemas pois não tem uma compreensão sobre suas consequências. [1] [6]

- *F. Hackers de Chapéu Vermelho (Red Hat Hackers)*

  São hackers que possuem suas próprias regras, querendo o mesmo objetivo que os *White Hats*, parar os *Black Hats* e também querer fazer algum "mal"aos mesmos utilizando métodos agressivos parar destruí-los completamente. [1] [6]

Observação: As cores de chapéus menos conhecidas possuem significados diferentes e/ou discrepantes dentro da comunidade hacker. Logo, certas definições dadas anteriormente podem mudar, dependendo do contexto.

2.4 Como os Ataques Cibernéticos Ocorrem

Os criminosos cibernéticos possuem diversas formas de enganar as pessoas para conseguir acesso a informações, sendo a principal delas a Engenharia Social, que consiste em basicamente na arte de persuasão no uso da autoconfiança, comunicação e aptidão profissional. Porém a dependência tecnológica continua sendo perigosa, muitos fraudadores aproveitam a pandemia mundial para realizar o envio contínuo de e-mails, mensagens de SMS, links de contas bancárias supostamente atrasadas, adiantamento de pagamentos, bloqueio de contas, entre muitas outras abordagens [5].

2.5 Prevenção de dados e informações

Todas as pessoas do mundo devem saber que estão sempre vulneráveis a ataques de cibercriminosos e que devem tomar todas as medidas possíveis para evitar qualquer consequência. Principalmente em meio essa pandemia, foram analisados alguns tipos de golpes e como se prevenir corretamente, sendo eles:

- *A. Sites Falsos sobre Informações da Covid-19*

  Ataque: Com a procura constante por informação sobre a atual pandemia, pesquisadores observaram vários sites falsos sobre o coronavírus para atrair diversas pessoas, porém essa fachada apenas esconde possíveis malwares e spywares para injetar nos sistemas das vítimas, sendo recuperado dados de cookies, cripto-moedas, etc.

  Prevenção: Nenhum usuário deve instalar qualquer aplicativo em locais sem nenhum nível de segurança, apenas lojas com certificações, olhando em conjunto o nível de acesso que os aplicativos tem sobre o celular, além de verificar em navegadores web se a conexão é segura.

- *B. E-mails de Pishing sobre o Coronavírus*

  Ataque: Os criminosos se passam por agentes, desde a Organização Mundial de Saúde (OMS) ou do próprio Estado, fazendo a solicitação de doações, acesso a links ou download de arquivos para inserir malwares no computador da vítima [8].

  Prevenção: Para se proteger desse tipo de ataque, deve ficar atento a fonte do envio dessas mensagens, não baixar nenhum arquivo confiável e buscar dados nas páginas oficiais dessas organizações [8].

- *C. Falsas Lojas e Serviços Online de Compra e Venda*

  Ataque: Cada vez mais novas lojas surgem no ambiente online por causa da pandemia e do desenvolvimento tecnológico, com isso golpistas aproveitam a venda de produtos necessários para prevenção contra o Covid-19 como álcool gel, mascaras faciais, luvas, entre outros materiais para comercializar de forma irregular e com baixa qualidade, podendo até mesmo nem chegar ao consumidor final. Para exibição do anuncio desses produtos, esses indivíduos fazem descontos enormes ou com estoque limitado, enganando facilmente qualquer pessoa com essas ofertas sedutoras [8].

  Prevenção: As pessoas devem comprar produtos ou mercadorias de lojas confiáveis e com determinadas certificações. Procure sempre por informações relacionadas a loja como reclamações, além de detalhes de contato, endereço e localização para contatar pessoas da equipe [8].

2.6 Notícias de Ataques em Época de Pandemia

Com o aumento crescente de usuários no uso de computadores e dispositivos moveis, diversos criminosos se aproveitam do momento para realização de golpes utilizando o coronavírus como ”isca”, assim, o aumento de tentativas gerais de ataques cresceu 15% de janeiro a fevereiro de 2020 [4]. Um dos casos mais recentes desses problemas teve como alvo a empresa Zoom, basicamente o nível de segurança do software possuía diversas brechas em sua aplicação, criando um ambiente perfeito para vazamento de dados de login e senha de usuários, que foram distribuídas na Dark Web em um preço de R$ 0,10 cada uma. Em torno de 500 mil contas teve seus dados vazados, com isso foi gerado enorme repercussão em blogs, sites e fóruns sobre a confiança da empresa com seus clientes, porém com o tempo a Zoom investiu muito em segurança para mudar esse cenário [10].

2.7 Ataques a Pesquisas Científicas Globais

Com a evolução cada vez mais acelerada da Covid-19, muitos países do mundo estão em uma corrida contra o tempo em busca de qualquer método para prevenção e tratamento do vírus. Hackers têm procurado informações e dados relacionados ao vírus e as suas pesquisas em diversas instituições (empresas farmacêuticas, universidades, etc), porém sem nenhuma evidência de roubo de dados, apenas com fortes suposições para busca de inteligência na obtenção de vantagens na criação de uma vacina. Agências de segurança nacional de alguns países já alertaram sobre esses ataques e como se proteger em novos casos [2]. Em outros casos, hackers realizam a criação de sites de rastreamento de casos de Covid-19 para roubar nomes de usuários, senhas, contas bancárias armazenadas no navegador, entre outros ataques, também são responsáveis por espalhar fake news sobre a pandemia para causar caos na população e no governo, aproveitando que maior parte das pessoas estão em casa [9].

2.8 Conclusão

Com o aumento cada vez maior no número de casos de pessoas infectadas pelo mundo pelo novo coronavírus, ficar em casa está sendo umas das melhores medidas preventivas possíveis, porém ao mesmo tempo que os ataques cibernéticos crescem de maneira muito rápida. Aprendendo a lidar com todos esses ataques e como se prevenir, tanto empresas como a própria população podem ficar mais tranquilas no conforto de sua residência, porém sempre observando novos méto-

dos utilizados pelos golpistas para tentar invadir seus sistemas. Logo, o desenvolvimento de novos meios de segurança são essenciais para impedir cada vez mais esses problemas e com isso incentivar outras pessoas a se comportarem de maneira adequada e segura no mundo digital.

2.9 Bibliografia

[1] ALPINESECURITY. Hacker hat colors and inside look at the hacking ecosystem. [Acessado em: 20/07/2020.]. URL: https://alpinesecurity.com/blog/hacker-hat-colors-an-insidelook-at-the-hacking-ecosystem/.

[2] BBC. Coronavirus: Cyber-spies hunt covid-19 research, us and uk warn. [Acessado em: 20/07/2020.]. URL: https://www.bbc.com/news/technology-52551023.

[3] Hackers League Books. Who are blue hat hackers? [Acessado em: 20/07/2020.]. URL: https://medium.com/@hackersleaguebooks/who-are-blue-hat-hackers-aeb443b90c29.

[4] CANALTECH. Ataques hackers crescem á medida que a pandemia da covid-19 se alastra. [Acessado em: 20/07/2020.]. URL: https://canaltech.com.br/hacker/ataques-hackers-crescema-medida-que-pandemia-da-covid-19-se-alastra-162080/.

[5] CSPSECURITY. Preventing fraud: Hackers are taking advantage of covid-19. [Acessado em: 20/07/2020.]. URL: https://www.cspsecurity.com/blog/preventing-fraud-howhackers-are-taking-advantage-of-covid-19/.

[6] GEEKSFORGEEKS. Types of hackers. [Acessado em: 20/07/2020.]. URL: https://www.geeksforgeeks.org/types-of-hackers/.

[7] HELPNETSECURITY. The history of hacking. [Acessado em: 20/07/2020.]. URL: https://www.helpnetsecurity.com/2002/04/08/the-historyof-hacking/.

[8] KASPERSKY. How to stay safe hackers scammers. [Acessado em: 20/07/2020.]. URL: https://www.kaspersky.com.br/resourcecenter/threats/coronavirus-how-to-stay-safe-hackersscammers.

[9] MODERNHEALTHCARE. Hackers taking advantage covid-19 spread malware. [Acessado em: 20/07/2020.]. URL: https://www.modernhealthcare.com/cybersecurity/hackerstaking-advantage-covid-19-spread-malware.

[10] TECMUNDO. Hackers venderam mais de 500 mil contas do zoom na 'dark web'. [Acessado em: 20/07/2020.]. URL: https://www.tecmundo.com.br/seguranca/152061-hackersvenderam-500-mil-contas-zoom-dark-web.htm.

Capítulo 3

Hacking Social: Da Origem à Aplicação

Beatriz Rogers Tripoli Barbosa
Gabriela Bergamo dos Santos
Gabrielle Bulhões Oliveira
Julia Ferreira da Silva
Maria Eduarda Caixeta Lella

Resumo

Artigo com o intuito de apresentar o Hacking Social -também conhecido como Engenharia Social-, bem como suas aplicações, muito presentes na sociedade contemporânea. Além disso, este visa ressaltar medidas de prevenção aos ataques desse tipo, visto que são comumente realizados.

3.1 Introdução

A engenharia social não é um termo com o qual a grande maioria das pessoas está familiarizada e, por isso, quando se fala nesse assunto, é interessante começar desmembrando a palavra. Engenharia significa que existe uma construção - nesse caso, feita sobre o uso inesperado de informações consideradas sigilosas, e social denota a exploração do uso de relações interpessoais como meio para conseguir as informações desejadas.

O principal pilar do assunto em questão consiste na manipulação de pessoas sem que elas percebam o que está sendo feito, para que, então, o processo seja bem sucedido. Ou seja, quem está por trás desse tipo de ação, busca utilizar das vulnerabilidades, bem como das confianças que as pessoas apresentam em seus hábitos, para fazer com que a vítima realize a ação desejada.

É importante destacar que a engenharia social é comumente utilizada pelos hackers pelo fato de ser uma das maneiras mais fáceis de induzir as pessoas a cederem as informações necessárias, seja por meio do uso de malwares ou da abertura de links que a direcionam a um site infectado, quando comparada à realização da invasão de um sistema sem a posse das mesmas.

Este artigo procura evidenciar as principais aplicações desse tipo de ataque, tanto benéficas quanto maléficas, buscando ampliar o conhecimento do leitor não só acerca do valor de seus dados pessoais, como também em relação a maneiras para protegê-los.

3.2 Origem

O hacking social está presente no mundo desde os primórdios da humanidade, sendo relatado no livro de Gênesis, contando que o Diabo, na forma de cobra, conseguiu convencer Eva de que Deus estava mantendo poderes para si e deixando ela e Adão sem comerem o alimento proveniente da Árvore da Vida. Com isso, o Diabo, mexeu com o orgulho e ganância de Eva fazendo-a desobedecer a Deus e comer a fruta proibida.

No exemplo citado acima, é possível perceber que o Diabo, claramente, fez o uso das técnicas apresentadas pela engenharia social. Porém tal expressão se popularizou somente em meados de 1990, tendo uma acepção de fácil entendimento: ganhar a confiança e, muitas vezes, enganar alguém buscando conseguir informações importan-

tes e, com isso, moldar formas de burlar o sistema de defesa desejado.

Kevin Mitnick, famoso hacker, que contribuiu com a propagação do assunto, contou em seu livro que um de seus primeiro contatos com a engenharia social aconteceu ainda no ensino médio, na década de 70. Foi nessa época que ele conseguiu acesso a informações confidenciais de uma empresa telefônica, o que tornou possível com que ele realizasse ligações de forma gratuita e reproduzisse diferentes mensagens quando os clientes da companhia fossem utilizar seus telefones. Tudo isso foi feito por meio da exploração dos sistemas e dos funcionários da empresa.

Mitnick acabou ficando 5 anos preso como punição por diversos crimes por ele realizados, incluindo roubo de senhas e acesso a redes de computadores de grandes corporações mundiais. Após ser solto, no ano de 2000, acabou optando por trabalhar como consultor e autor de livros relacionados à segurança de sistemas e, a partir de 2003, decidiu atuar contra hackers, tornando sistemas inatacáveis.

3.3 Tipos

Tailgating

O *tailgating* é um claro exemplo que pode ser utilizado para mostrar que os ataques relacionados à engenharia social não estão restritos ao meio cibernético, tendo em vista que ele ocorre no meio físico. Esse tipo de ataque consiste na entrada em um local de passagem restrita sem que a pessoa cumpra os requisitos para poder entrar. Para isso o atacante possui algumas possibilidades, tais como: entrar logo atrás de alguém que tenha acesso à essa área realizando uma ação rápida, para evitar que a vítima perceba que ele não tem a autorização necessária, como pedindo para que essa pessoa segure a porta, por exemplo, ou se passando por uma pessoa que tenha livre acesso ao local desejado. Uma vez que obtenha sucesso na ação, o atacante inicia a busca pela informação ou objeto desejado.

Phishing

Phishing é o tipo mais comum e utilizado do campo do Hacking Social. Possuindo diversas maneiras de ser implementado, ele é utilizado para a aquisição de diversas informações desejadas por meio de emails, links ou downloads provenientes de pessoas mal intencionadas. De acordo com o Internet Security Threat Report, ISTR, de 2017 [13], a Symantec, monitorando atividades de ataque, constatou que 85% das organizações que utilizam os serviços da empresa, afirmaram terem sofrido ataques de phishing, o que mostra o quão comum é ser vítima desse tipo de ataque.

Figura 3.1: Descrição do processo. [17]

Figura 3.2: Imagem ilustrativa do ataque. [3]

Deceptive Phishing

Essa categoria de *phishing* é considerada a mais comum dentre as demais. Consiste em um ataque generalizado em que tenta-se desenvolver, principalmente, uma réplica de um e-mail da empresa na qual a vítima trabalha. Geralmente os atacantes tentam fazer com que o assunto presente no corpo do e-mail pareça urgente para que, assim, a vítima aja o mais rápido possível, sem realizar grandes análises e, rapidamente, fornecer suas informações pessoais.

No entanto, esse tipo de ataque é possível ser percebido por alguns sinais presentes na mensagem, como erros gramaticais ou o uso de termos considerados muito generalizados.

Spear Phishing

Esse tipo de ataque é direcionado para uma pessoa específica, da qual o atacante obteve informações pelos perfis nas redes sociais e outros locais públicos. Esses dados são, então, utilizados para a escrita de um e-mail que aparenta, além de legítimo, ter sido escrito e enviado por alguém de confiança da vítima, como um amigo, por exemplo.

Com isso, busca-se enganar o usuário e influenciá-lo a responder a mensagem, fornecendo o que é desejado. As consequências do *spear phishing* podem variar desde roubo de identidade até chantagem.

Vishing

O *vishing* tem esse nome, pois é feito com o uso da Voz Sobre Protocolo de Internet, VoIP, que utiliza a internet como meio de transmissão de áudio, mesmo sistema utilizado por redes como Skype e WhatsApp. O motivo do uso dessa tecnologia é que ela torna possível alterar o ID de origem da ligação, fazendo com que o atacante sinta-se livre para escolher o ID considerado mais convincente para falar com a vítima, podendo aparentar ser uma empresa, por exemplo.

Já que o gerenciador do ataque geralmente utiliza informações relacionadas ao lado emocional da vítima, esta, por sua vez, acaba ficando nervosa e esquecendo informações passadas em treinamentos de conscientização de segurança que tenha feito, o que a torna propensa a ceder ao atacante.

Além disso, diferentemente de e-mails e fax, que não necessariamente precisam ser respondidos rapidamente, em uma ligação, o alvo sente-se mais pressionado e acaba sendo coagido a passar os dados pedidos por quem está por trás da chamada, se passando por outra pessoa ou serviço. Tudo isso acaba colaborando para o sucesso do ataque.

Whaling Attack

Considerado um ataque que acaba tomando grandes proporções, o *whalling attack* é direcionado aos "peixes grandes", trabalhadores de altas categorias de empresas, buscando conseguir acesso a suas contas de e-mail ou falsificá-las.

Ao obter êxito se passando pelo Diretor Executivo de uma empresa, CEO, o invasor tenta estabelecer contato com os funcionários do alvo, pedindo algum tipo de arquivo. Quando consegue-se falsificar um e-mail de alguém tão importante de uma empresa como um CEO, geralmente as demais informações acabam sendo conseguidas, pois a maioria dos funcionários não se recusaria a atender um pedido vindo de seu diretor, o que acaba colocando a empresa inteira em risco.

SMiShing

Essa categoria de ataque acaba sendo parecida com o *vishing*, porém, ao invés de utilizar o VoIP, faz o uso de mensagens de texto, por isso o nome é parecido com SMS. Como o golpista pode enviar essas mensagens em massa, o ataque pode atingir um amplo número de pessoas.

Nesse tipo de *phishing*, o invasor busca enganar a vítima, dizendo que ela ganhou algum tipo de prêmio ou sorteio, ou, os menos óbvios, se passam por bancos ou empresas de cartão de crédito. As informações do alvo são pegas por meio de um link, por exemplo.

É importante ressaltar que, nesse caso, acabam dando ênfase na parte da mensagem relacionada ao **tempo limitado** que a pessoa tem para acessar o link e conseguir obter seu prêmio, fazendo com que ela acabe respondendo a mensagem sem pensar muito.

De acordo com [13], no ano de 2016, relataram que 1 em cada 131 emails não eram solicitados e possuíam algum tipo de anexo que continha um *malware*. Além disso, pelas análises do mesmo ano, fez-se uma estimativa de que cerca de 269 bilhões de emails tenham sido enviados em um único dia em nível global. Com base nesses números, calcula-se que, diariamente, foram distribuídos 2.044.400.000 -dois bilhões quarenta e quatro milhões e quatrocentos mil- emails, o que mostra, com clareza, o quão amplo e comum é um ataque desse tipo.

Pharming

Como a maioria das pessoas acaba reconhecendo emails de *phishing*, o *pharming* surgiu como uma maneira mais sofisticada e de difícil reconhecimento. Essa categoria opera por meio da falsificação de Sistemas de Nomes e Domínios, DNS, em que o golpista acaba conseguindo alterar o endereço de IP de um site em específico e redirecionar o usuário a um site maligno.

A maneira mais eficiente de se proteger contra esse tipo de ataque consiste em utilizar somente sites que sejam protegidos por HTTPS quando tiver que fazer a inserção de algum tipo de informação relevante.

Ransomware Phishing

Ao contrário dos outros tipos de *phishing*, que buscam conseguir informações propriamente ditas da vítima, o *ransomware phishing* visa fazer um computador de refém, convencendo seu alvo a baixar um *malware*.

Para liberar seus arquivos e seus computadores, a maioria das pessoas acaba pagando o valor de resgate o que acaba contribuindo para a continuidade desse tipo de ataque, à medida que vai obtendo sucesso.

Baiting

É pelo *baiting* que o hacker explora a curiosidade humana. Uma de suas principais características é a realização de uma promessa que, relacionada a algo que trará benefícios ao alvo, é utilizada para enganar a vítima. Um exemplo genérico desse ataque é quando o golpista apresenta um arquivo maquiado como um suposto *update* de software ou músicas e filmes pirateados.

Além disso, existem outras maneiras para a realização desse tipo de ataque, como um hack físico, por exemplo, por meio do uso de um USB ou CD. Para isso, o hacker deve deixar o USB em um local público geralmente relacionado à empresa-alvo. Quando ele despertar a curiosidade de alguém e essa pessoa conectá-lo em seu computador, o mesmo será automaticamente infectado com o *malware*, roubando as informações pessoais da vítima ou invadindo seus sistemas.

Um exemplo de *baiting* bem conhecido que acabou ficando marcado na história da mitologia grega foi durante a batalha travada entre o exército grego e o troiano. Os gregos armaram um cenário que aparentava terem abandonado o cerco e deixado um grande presente aos troianos, um enorme cavalo de madeira comumente chamado de Cavalo de Tróia. Ao levar o cavalo para dentro dos portões da cidade, sem saber que a tropa inimiga estava escondida dentro dele, o exército de Tróia tornou possível com que os gregos obtivessem sucesso na conquista da cidade.

Quid pro Quo

Sendo um variante do *baiting*, neste caso, o hacker acaba oferecendo um serviço em troca de algum tipo de ação específica, como a passagem de informações ou o acesso à alguma plataforma desejada, geralmente incorporando o perfil de alguém que possui uma maior autoridade em assuntos ligados à tecnologia, como um técnico de TI, por exemplo.

A importante distinção do *quid pro quo* e do *baiting* é a existência de uma troca de serviços, enquanto no *baiting*, não há relação direta entre o atacante e a vítima, nesse caso a interação é necessária para que o ataque aconteça.

O mesmo Kevin Mitnick citado anteriormente, foi responsável por uma série de ataques relacionados à engenharia social utilizando esse método. Fazendo diversas ligações para os usuários das empresas em que estava interessado e oferecendo ajuda para problemas de TI tanto reais quanto falsos, ele conseguia acesso a diversos sistemas por meio de perguntas que pareciam inofensivas e cotidianas enquanto trabalhava na resolução do 'problema' do sistema alvo.

Pretexting

Esse tipo de Hacking Social consiste na prática de se apresentar como outra pessoa, incorporar um perfil diferente visando conseguir informações. É possível ir tão longe com essa suposta 'atuação' chegando a criar uma nova personalidade completa.

É importante destacar, porém, que o *pretexting* não utilizado apenas como uma ferramenta de Hacking Social, geralmente está presente no dia a dia de pessoas que trabalham com vendas, que falam muito em público ou até em profissões como videntes. A parte importante para que se obtenha êxito com esse método é a construção da confiança para com a vítima, o que acaba dando credibilidade ao personagem criado.

Trazendo para a realidade, na década de 60, Frank Abagnale convenceu funcionários da Pan Am de que ele era um piloto comercial. Após assumir a identidade de um jornalista do jornal da escola, conseguiu adquirir conhecimento sobre políticas, procedimentos e terminologias utilizadas no setor contribuindo para que, juntamente com o uniforme de piloto da Pan Am, ele conseguisse voar gratuitamente. Além disso, Frank utilizou de seu conhecimento acerca do processo bancário da Pan Am para conseguir cheques fraudulentos. Por esse exemplo é possível perceber o quão sério pode se tornar um caso de pretexting.

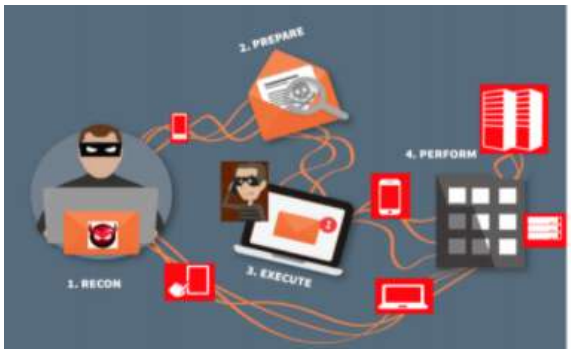


Figura 3.3: Exemplificação de Pretexting [7]

Scareware

O *scareware* está relacionado com o envio de alarmes falsos e ameaças fictícias levando o usuário a acreditar que seu sistema foi infectado por um *malware*, convencendo-os a instalar um software, que não lhes traga nenhum tipo de benefício, ou o *malware* diretamente.

Na maioria das vezes essa ação é feita por meio de anúncios pop-up que aparentam ser legítimos, indicando que o sistema utilizado pelo usuário encontra-se infectado e, utilizando o medo da vítima a seu favor, o hacker a convence a baixar um falso antivírus.

Watering Hole

Esse tipo de Hacking Social é mais utilizado por hackers e criminosos cibernéticos que, após traçarem um perfil da vítima, inserem um código malicioso em sites específicos que ela costuma visitar. São nessas páginas da internet que o *backdoor trojan* (tipo de arquivo que permite que os dados do sistema no qual ele foi instalado sejam acessados por outra pessoa, dando-lhe controle sobre a máquina do alvo) acaba sendo implantado em seu computador.

É importante ressaltar que, na maioria das vezes, o *watering hole* geralmente está bastante relacionado tanto aos atos de espionagem virtual quanto aos ataques maiores como aqueles que possuem o Estado como seu alvo principal.

Exemplos de ataques reais

É muito comum as pessoas pensarem que, nos dias atuais, como a tecnologia continua sendo desenvolvida em uma alta velocidade e apresentando cada vez mais ferramentas de segurança que são utilizadas para proteger grandes empresas de sofrerem ataques de hackers, elas deixaram de ser alvo dos golpistas cibernéticos. Porém, a realidade é outra.

No ano de 2016 a Yahoo descobriu que, três anos antes, em 2013, a companhia sofreu um ataque que divulgou mais de 3 bilhões de dados de seus usuários, desde e-mails até as respostas das perguntas de segurança. O fato de a empresa ter tomado conhecimento sobre tal ação apenas três anos mais tarde foi o que tornou tal notícia tão polêmica, afinal, não conseguiram fazer nada para tentar impedir que os golpistas obtivessem acesso a essas informações durante o ataque.

Uma notícia mais recente relacionada a essa temática envolve uma das maiores redes sociais atu-

almente, o Facebook. Ao decorrer de 2018, estima-se que a plataforma tenha sofrido, pelo menos, dois vazamentos de dados, ocorrendo em setembro e outubro, totalizando cerca de 80 milhões de informações de usuários expostas.

Além disso, um exemplo extremamente comum e muito recorrente é o envio de e-mails (geralmente feito por bots) a diversas pessoas contendo a senha delas de algum site que sofreu um ataque e acabou vazando dados dos usuários. Os hackers fazem isso na tentativa de convencer a vítima de que eles têm acesso às informações sobre ela a fim de fazê-la pagar alguma quantia em dinheiro para que não utilizem seus dados.

A falta de informação, no entanto, faz com que a pessoa acredite que aquele e-mail foi direcionado especificamente para ela e que não há tempo para alterar a senha ou algo do tipo, quando na verdade ele foi enviado para milhares de pessoas, sendo que essa senha foi retirada de um site específico. Com esse exemplo demonstra-se a importância de utilizar senhas diferentes para contas criadas nos diversos serviços online.

Figura 3.4: Exemplo de email enviado com a senha do usuário [Recebido por um dos membros do HackoonSpace]

Com isso, torna-se possível perceber que os ataques de hackers são, ainda, um problema para diversas empresas que atuam no ramo da tecnologia e precisam estar, constantemente, atualizando suas medidas de segurança para evitar com que seus usuários tenham suas informações pessoais expostas.

3.4 Utilidades

Mais do que apenas ficar familiarizado com os tipos de hacking social existentes, é necessário conhecer e entender sua aplicação prática, que pode ser encontrada em diferentes ações do dia-a-dia da sociedade.

As práticas de engenharia social são utilizadas em diversos campos, dentre eles o correspondente às investigações policiais. O filme *Big Momma's House* pode ser utilizado com um bom exemplo: nele, um membro do FBI, por meio de um disfarce, consegue emprego como babá dos filhos de um homem que é alvo de uma investigação. Dessa forma, ele se aproximou do investigado e conseguiu descobrir informações importantes para solucionar o caso por meio do uso de técnicas de hacking social, criando um personagem para si e persuadindo os demais a acreditarem nele. O filme é uma história fictícia, mas no mundo contemporâneo, muitos policiais se disfarçam para conseguirem se infiltrar em organizações criminosas com o objetivo de obter dados sobre o grupo.

A criação de um personagem também é muito comum de acontecer na política. O político precisa criar uma imagem positiva de si, coerente com os valores e ideais de sua campanha para, assim, agradar seu eleitorado. Outra prática comum é a de controlar a opinião pública por meio do uso de palavras tendenciosas ao se referir a um certo grupo. Em diversas campanhas eleitorais, a maioria das pessoas que estão concorrendo a um cargo, devido a seu interesse político, buscam diminuir e muitas vezes agregar uma imagem negativa ao grupo que demonstra oposição a sua vitória.

No entanto, como citado anteriormente no artigo, o hacking social pode ser utilizado visando atitudes maléficas. Em 2020, Barbara Corcoran, empresária e participante do programa *Shark Tank*, foi alvo de um hacker que se passou por assistente da vítima. Pelo fato de investir em imóveis, a empresária não achou estranho receber um email sobre a aprovação de um pagamento para uma reforma imobiliária (o que ela não sabia é que este teria sido enviado pelo hacker), o que resultou na perda de US$ 388.700 que havia sido depositado por uma de suas funcionárias para o suposto assistente.

Outro roubo envolvendo hacking social ocorreu em 2019 e sua vítima foi uma subsidiária europeia da Toyota. Por meio de um BEC *scam* (*phishing* ou *ransomware* avançado em grandes empresas), o hacker fingiu ser um parceiro de negócios da companhia e mandou e-mails aos membros do depar-

tamento financeiro pedindo para que depositassem uma quantia de $37 milhões em sua conta. Como a Toyota é uma grande empresa, transferências dessa quantia não costumam causar grande alarde na companhia, o que acabou facilitando o trabalho do hacker.

Por fim, técnicas de engenharia social são muito utilizadas no dia-a-dia da maioria das pessoas (as figuras abaixo ilustram as formas mais recorrentes de engenharia social), muitas vezes sem nem saberem. Médicos fazem o uso da "elicitação", que consiste em uma maneira de obter informações detalhadas sobre seus pacientes; vendedores, assim como as pessoas que trabalham com marketing, tentam persuadir os clientes a comprarem seus produtos; em um debate, mesmo que por meio da apresentação de dados que possam ser comprovados também busca-se a persuasão tanto do oponente quanto do público a acreditarem em um dos lados, entre outras.

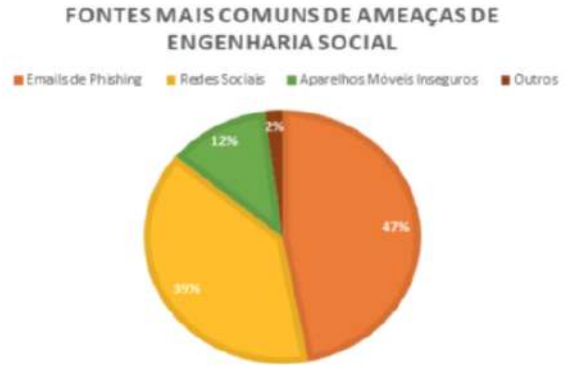


Figura 3.5: Gráfico com as principais fontes de ameaças de Engenharia Social [9]

3.5 Considerações Finais

Como a maior parte das técnicas de Hacking Social está intrinsecamente relacionada com artefatos tecnológicos, pode-se dizer que o mesmo vem, continuamente, sendo desenvolvido e aprimorado ao longo dos anos a medida que as tecnologias vão sendo aprimoradas.

Porém, mesmo passando por diversas mudanças, ainda existem medidas que podem ser usadas para evitar com que as pessoas sejam vítimas de

Figura 3.6: Gráfico exemplificando outras formas recorrentes de Engenharia Social [19]

ataques que envolvam o uso da engenharia social, medidas essas que serão citadas a seguir.

A maneira mais simples é sempre desconfiar quando alguém pedir informações que geralmente não estão disponíveis ao público ou quando uma pessoa desconhecida estiver oferecendo algo de maneira gratuita. É importante também, evitar abrir e-mails e anexos de fontes duvidosas, bem como utilizar o chamado *multi-factor authentication* para proteger as credenciais (itens mais procurados pelos hackers) quando um sistema for comprometido.

Ficar atento às informações contidas nos perfis das redes sociais também é de suma importância, tendo em vista que, caso alguém esteja tentando invadir as contas *online* de uma pessoa, pode acabar encontrando indícios que o ajude a responder as perguntas de segurança. Além disso, é preciso prestar atenção ao criar senhas para as contas, buscando sempre as mais fortes possíveis, para criar uma barreira mais forte a um possível ataque.

Por último, manter o anti-vírus do computador sempre atualizado e optar pelo uso de sites que possuam "https://"(indicando que são seguros), também é uma maneira de evitar com que o ataque seja bem sucedido.

3.6 Bibliografia

[1] J. ALBORS. Você sabe o que é um backdoor e como diferenciá-lo de um trojan? URL: `https://www.welivesecurity.com/br/2016/08/31/backdoor-e-trojan/`.

[2] BANRISUL. Cuidados com a engenharia social. URL: http://www.banrisul.com.br/bob/download/Banrisul_cuidados_com_a_engenharia_social.

[3] R. CARDOZO.

[4] C. EDU. What is social engineering? URL: https://www.forcepoint.com/cyber-edu/social-engineering.

[5] S. T. EDUCATION. The social engineering framework. URL: https://www.social-engineer.org/framework/influencing-others/pretexting/.

[6] GATEFY. A história da engenharia social na era dos computadores e da internet. URL: https://gatefy.com/pt-br/postagem/historia-da-engenharia-social-computador-internet/.

[7] GWU. Exemplificação de pretexting. URL: https://blogs.gwu.edu/gwinfosec/.

[8] T. HACKERSPACE. Engenharia social. 2020.

[9] Iconix. The security threat of social engineering. URL: https://iconixtruemark.wordpress.com/2011/09/23/the-security-threat-of-social-engineering/.

[10] IMPERVA. Social engineering. URL: https://www.imperva.com/learn/application-security/social-engineering-attack/.

[11] K. LAB. Engenharia social - definição. URL: https://www.kaspersky.com.br/resource-center/definitions/what-is-social-engineering.

[12] N. LINDSEY. Toyota subsidiary loses $37 million due to bec scam. URL: https://www.cpomagazine.com/cyber-security/toyota-subsidiary-loses-37-million-due-to-bec-scam/.

[13] S. NATHANIEL. The history and evolution of social enginee-ring attacks. URL: https://commissum.com/blog-articles/the-history-and-evolution-of-social-engineering-attacks.

[14] E. NEGROMONTE. O que é engenharia social? URL: https://sempreupdate.com.br/o-que-e-engenharia-social-principais-duvidas/.

[15] P. PAGANINI. The most common social engineering attacks. URL: https://resources.infosecinstitute.com/common-social-engineering-attacks/#gref.

[16] R. SANTOS. Qual a origem da engenharia social? URL: https://www.compugraf.com.br/origem-da-engenharia-social/.

[17] W. SECURITY.

[18] M. SILVA. Marketing pessoal para políticos: dicas para sair na frente. URL: https://neritpolitica.com.br/blog/marketing-pessoal-para-politicos.

[19] Project Sparks. Social engineering attacks and the smart grid. URL: https://project-sparks.eu/social-engineering-attacks-and-the-smart-grid/.

[20] TERRA. Dez hackers famosos e seus feitos. URL: https://www.terra.com.br/noticias/tecnologia/infograficos/hackers/hackers-06.htm.

[21] TILT. Apresentadora do shark tank perde r$ 1,7 milhão em golpe na internet. URL: https://www.uol.com.br/tilt/noticias/redacao/2020/02/27/barbara-corcoran-perde-us-400-mil-por-phising.htm.

[22] VARVARA. 10 most common phishing attacks. URL: https://resources.infosecinstitute.com/10-most-common-phishing-attacks/.

Capítulo 4

Inteligência Artificial: Da origem à aplicação

Beatriz Rogers Tripoli Barbosa
Gabrielle Bulhões Oliveira
Guilherme Fumagali Marques
Julia Ferreira da Silva
Maria Eduarda Caixeta Lella

Resumo

Através desse artigo, visamos apresentar o conceito de Inteligência Artificial, mostrando, em linhas gerais, como ela funciona e em quais contextos ela é utilizada, desde segurança até os ataques cibernéticos.

4.1 Introdução

Inteligência Artificial é um conceito que se encontra cada vez mais presente no ramo da tecnologia e vem sendo utilizado para as mais diversas funções. Assistentes virtuais como Alexa e Siri, bots usados para o atendimento de clientes em sites e filtros de spam presentes nos e-mails são alguns exemplos que mostram como a IA veio, ao longo dos anos, ganhando espaço na sociedade. Sendo assim torna-se interessante sabermos, ao menos, do que se trata e como essa tecnologia pode ser utilizada.

A definição do termo era considerada muito vaga pelo fato de apenas dizer que se tratava de "máquinas que são inteligentes". Até que, Stuart Russell e Peter Norving apresentaram uma versão um pouco mais completa, dizendo que é o *"estudo de agentes que recebem percepções do ambiente e executam ações"* [6]. Ou seja, em linhas gerais, consiste na criação de programas de computador que sejam capazes de fazer uma máquina realizar determinadas ações baseadas nas informações que ela recebe, e que geralmente precisam de algum tipo de inteligência humana.

Tendo isso em vista, é muito comum que as pessoas, ao pensarem nesse conceito, o associem a filmes nos quais o mundo é comandado por máquinas que vão destruir tudo, porém não é bem assim. A inteligência de nível humano ainda é algo que os pesquisadores não conseguiram atingir e não conseguem prever quando e se será possível de ser realizado algum dia.

Pelo fato de o estudo sobre essa área da tecnologia ser considerado muito extenso e complexo, esse artigo procura, por meio de uma linguagem mais clara e simples, desmistificar, trazer um pouco do que é, como funciona e como hackers e sistemas de segurança utilizam a Inteligência Artificial.

4.2 História

É evidente que quando se fala em "automatizar tarefas", o homem sempre desejou ter algum tipo de máquina que fosse capaz de realizar suas ações e pensar igual a ele. Como muitos estudos científicos e pesquisas, essa foi mais uma área que acabou ganhando foco e sendo desenvolvida durante a Segunda Guerra Mundial.

O primeiro indício de estudo que se tem sobre Inteligência Artificial é um artigo publicado em 1943, por Warren McCulloch e Walter Pitts, que introduziu as redes neurais por meio de um mo-

delo matemático capaz de imitar o sistema nervoso humano. Depois disso, um dos grandes conhecidos na área da computação, Alan Turing, desenvolveu um método, o Teste de Turing, consistindo em uma espécie de interrogatório que determinava se uma máquina era ou não inteligente, verificando se era possível fazer com que ela passasse por um humano em uma conversa por escrito. Esse experimento de Turing é considerado tão importante que acabou se tornando a base para os estudos de IA [21].

Mesmo depois da criação e publicação desses estudos, foi somente na metade do século XX, na Conferência de Dartmouth [21], que eles finalmente receberam um nome específico, **Inteligência Artificial**. A partir daí, conforme as pesquisas e as tecnologias foram avançando, o campo de IA foi, cada vez mais, sendo desenvolvido, tanto que em 1969, surgiu o Shakey, primeiro robô com mobilidade, fala e autonomia de ação.

Desde então, esse vem sendo um campo de bastante foco dos pesquisadores, principalmente pelo fato de possuir diversas possibilidades de aplicação que variam desde a automação de indústrias até, como citado, o desenvolvimento de assistentes virtuais capazes de reconhecerem a fala do usuário e realizarem determinadas ações demandadas.

4.3 Termos Específicos

Ao fazer uma pesquisa sobre IA, nos deparamos com diversos termos técnicos, e, antes de fazer um aprofundamento no assunto, acaba sendo necessário saber, de maneira geral, o que os principais deles significam.

Redes neurais artificiais se referem a um algoritmo capaz de simular o funcionamento cerebral, que também é relacionado ao chamado "aprendizado por meio da experiência", que consiste em aprender as coisas conforme for executando determinadas ações.

Machine learning, é o sistema que melhora sua performance por meio da aquisição e acúmulo de conhecimento, ou seja, quanto mais conhecimento e informações forem dadas à máquina, melhor vai ser seu desempenho, trata-se de uma relação proporcional.

Data science, se trata do campo responsável por conceitos e informações de maior complexidade acerca de dados, fazendo análises visando realizar algum tipo de descoberta de padrões ou absorver algum tipo de conhecimento.

Deep learning, aqui entra o reconhecimento de fala e o facial que, relacionado ao aprendizado de máquina, faz o uso das redes neurais para processar tanto as informações quanto a aprendizagem.

Por fim, outro termo também muito utilizado, é o *Big data*, que diz respeito a grandes conjuntos de dados que, por serem mais complexos, precisam ser tratados de maneira diferente dos demais.

4.4 Ramos

A Inteligência Artificial possui diversos ramos de estudo cuja junção é fundamental para o êxito de suas aplicações, parte que será tratada no próximo tópico. Para ser possível ter uma noção geral e entender um pouco de como são feitas essas aplicações, torna-se importante conhecer alguns dos ramos dessa área.

Lógica

É por meio de uma linguagem lógica matemática que um programa tem todas as informações necessárias sobre seu comportamento que, de alguma maneira, irão influenciar em seu modo de agir. Sendo um dos ramos de extrema importância da IA, é pela lógica que o programa decide qual caminho seguir e sabe o que deve fazer.

Como exemplo, se um programa recebe o símbolo de "+"para realizar uma operação, é baseado na lógica pela qual ele foi escrito que a decisão de fazer uma adição e não uma multiplicação é tomada.

Reconhecimento de Padrões

Geralmente, os programas são feitos com o objetivo de buscar por padrões, para, assim, saberem o que deve ser feito e como deve ser feito, em IA não é diferente. Em uma partida de xadrez, por exemplo, se a máquina conseguir encontrar o padrão de jogadas que é utilizado pelo jogador, ela desenvolverá suas ações tomando isso como base e conseguirá atingir uma vitória. Aqui é importante ressaltar que, por se tratar de padrões mais

complexos, os métodos utilizados para encontrá-los são diferentes dos convencionais.

Robótica

Como o próprio nome já sugere, essa é a área que fica responsável pelo projeto,produção, operação e uso dos robôs. Assim sendo, é daqui que saem os programas desenvolvidos para o controle das ações desses tipos de máquinas, também são feitos estudos e pesquisas para tentar fazer com que seja possível existir um certo grau de interação social nesses robôs.

Aprendizado de Máquina

Conhecido como *Machine Learning*, esse é o ramo responsável por fazer as máquinas "pensarem", adquirirem experiência e, por meio do uso de dados, conseguirem solucionar problemas do mundo real. Aqui entra muita matemática, para possibilitar com que a máquina seja capaz de fazer uma análise e tradução dos dados recebidos e assim, modificar seu comportamento à medida que vai adquirindo experiências.

Um exemplo bem claro de aprendizado de máquina dado pelo [3], é a tradução de um texto, que nunca pode ser feita de maneira automática, ao "pé da letra". É sempre preciso levar em consideração o contexto da frase que está sendo traduzida pelo fato de uma mesma palavra poder ter diferentes significados de acordo com o contexto no qual ela é empregada. Sendo assim, o *Machine Learning* é o responsável por fazer com que os tradutores fiquem cada vez melhores.

Lógica Fuzzy

Quando chegam momentos em que torna-se difícil avaliar se uma certa condição é ou não verdadeira, entra a lógica Fuzzy. Ela trata esse tipo de informação considerada "incerta"por meio da avaliação do grau de o quão verídica pode ser a hipótese, variando de 0.0 a 1.0, sendo 0.0 para falso e 1.0 para 100% verdadeiro. Com isso, ela se baseia nesse índice para decidir qual será o próximo passo a ser seguido.

Sendo assim, por tratar de informações mais vagas, a Lógica Fuzzy pode ser comparada ao pensamento das pessoas, tendo em vista que algo pode ser verdadeiro para algumas e falso para outras, também pode ser utilizada para criar estimativas sobre determinado assunto.

Figura 4.1: Campos de Uso da Inteligência Artificial [2]

4.5 Inteligência Artificial Aplicada em Ataques Hackers

Vivemos em uma era digital, e por consequência disso, expor dados na internet é algo cada vez mais comum, desde publicar fotos em uma rede social ou até fazer compras em um e-commerce usando o cartão de crédito. Isso se tornou tão cotidiano que, muitas vezes, nós nem nos preocupamos com o valor daquelas informações, nem mesmo com as consequências de caso elas venham a ser expostas a criminosos na internet.

Com a crescente quantidade de dados na internet, áreas de estudo como a segurança da informação, segurança de dados e a privacidade digital vêm sendo cada vez mais importantes para evitar ataques hackers e falhas de segurança que possam causar vazamentos de dados pessoais e privados. Por outro lado, esses hackers também pesquisam e desenvolvem métodos que buscam explorar qualquer brecha do sistema, por menor que seja, ou aprimorar suas engenharias sociais no intuito de aplicar golpes em usuários. Com isso, vemos cada vez mais a IA sendo utilizada de forma mal-intencionada como um intermédio que fortalece ataques hackers.

Ataques com IA

As motivações de cybercriminosos são sempre conseguir informações e dados da vítima mesmo que de forma ilegal, visando, na maioria das vezes, alguma quantia em dinheiro que possivelmente será paga para que ela consiga a liberação de seus dados. Para isso, sempre tentam manter a máxima discrição na rede e encontrar formas de atingir o maior número de pessoas possível. Consequentemente, cada vez mais os hackers utilizam e aprimoram o uso de Inteligência Artificial e aprendizagem de máquina para servirem de ferramenta, como forma de auxiliá-los em ataques maiores e garantir maior anonimato.

Para usar Inteligência Artificial, um sistema precisa ser instruído e implementado, por isso, utilizar essa ferramenta não é uma tarefa simples e exige mais tempo e processamento computacional comparado a outros métodos. Entretanto, uma IA mal-intencionada pode ser muito mais eficiente do que um ser humano, tornando possível atacar uma maior quantidade de dispositivos e redes, de uma forma mais inteligente. Além disso, a IA pode ser mais seletiva, e como consequência, encontrar os melhores alvos e causar mais prejuízos à vítima.

Segundo o cientista de segurança de dados da Cylance, Brian Wallace [8], o uso de IA em ataques hackers não é recente e, por mais que sua implementação seja complicada devido aos riscos gerados ao tentar atingir o máximo possível de pessoas, essas IAs continuam obtendo cada vez mais dados ao longo do tempo, fazendo com que fiquem mais fortes e difíceis de serem detectadas. Paralelamente, podemos esperar malwares e ataques mais complicados de serem reconhecidos, além de mais precisos e destrutivos, com uma maior capacidade de propagação. E sobre as engenharias sociais que induzem uma vítima a cair em golpes, pode-se dizer que a IA consegue gerar phishings, notícias falsas e clickbaits mais convincentes.

Apesar de IA ser usada de forma mal-intencionada em alguns casos, especialistas em segurança da informação também utilizam essa tecnologia para aprimorar a segurança na internet, como acontece nos detectores de phishing em e-mails, nos quais uma inteligência artificial decide se a mensagem recebida é maliciosa ou não, com base nos e-mails usuais que a pessoa recebe. No entanto, por ser um algoritmo preciso, hackers utilizam dessa mesma tecnologia para tentar fraudar essa segurança. O objetivo do algoritmo criminoso, é monitorar e-mails e mensagens de texto da vítima e, assim, tentar entender como o detector de phishing está manipulando os e-mails do usuário a ser atacado para encontrar uma maneira capaz de enganar o algoritmo de segurança. Tudo isso para que seja possível deduzir um método que tenta extrair informações pessoais da vítima com um e-mail discreto que não aparenta ser perigoso.

A partir do que foi dito, é possível concluir que a IA é uma ferramenta ampla e forte, e que pode ser preocupante se utilizada de maneira errada. Além de seu uso em phishings, ela também pode ser utilizada para fins mais destrutivos, como modificar malwares e ransomwares de uma forma rápida e inteligente com o intuito de encontrar vulnerabilidades conforme as circunstâncias do ataque. Contudo, seu poder de ataque ainda não foi muito explorado, visto que os métodos tradicionais ainda funcionam. Esse fato mostra que a IA, apesar de muito útil, ainda não é muito comum em ataques atuais.

Exemplos de IAs ofensivas

O Emotet Trojan [10], é um malware relativamente comum na internet, mas que não deixa de ser perigoso, sendo usado ilegalmente para espionagem e tendo como alvo principal corporações comerciais. Por ser um malware incluído no banco de dados dos programas de segurança-usualmente chamados de “antivírus”-, é difícil para ele se manter despercebido ao tentar infectar uma máquina. Para evitar isso, os criminosos estão usando Inteligência Artificial para fazer com que os ataques sejam bem sucedidos, transmitindo uma maior confiabilidade para os criminosos. Isso mostra que a IA é capaz de restaurar diversas vezes um malware que antes aparentava ser inofensivo aos cientistas de dados e pesquisadores de segurança digital.

"Para resolver um problema, primeiro deve-se entendê-lo". Com base nesse pensa mento, pesquisadores da área de segurança da empresa IBM desenvolveram uma técnica de inteligência artificial baseada no ramo da “aprendizagem de má-

quina", para criar programas ofensivos capazes de enganar algoritmos de segurança [18]. Esse programa conseguia manter-se suspenso na máquina, de forma que ele ia analisando a situação e pudesse atacar na hora certa. Um malware parecido é o "Stuxnet". Com funcionalidades semelhantes, ele foi usado na prática por agências de espionagem dos Estados Unidos e de Israel para atacar uma instalação de enriquecimento de urânio do Irã. Com isso é possível medir o tamanho poder destrutivo desse tipo de malware, usado até mesmo em contextos de conflitos políticos.

Um exemplo prático do uso de IA na produção de fakenews aconteceu em 2019 [20], quando criminosos utilizaram dela para criar um "Deepfake", que é um tipo de IA que pode substituir a voz e a face de uma pessoa de uma forma relativamente "natural". O alvo foi um CEO de uma empresa de energia britânica. O programa replicou sua voz, e a usou para falar sobre transferências bancárias falsas, com o objetivo de prejudicá-lo.

Com os exemplos acima, é possível perceber que, quando usada de forma maliciosa, a Inteligência Artificial é capaz de causar sérios danos e de maneiras menos perceptíveis, o que faz com que seja necessário desenvolver cada vez mais os sistemas de segurança para que eles se tornem cada vez mais poderosos e evitem graves consequências.

Experimentos Práticos e Pesquisas Acerca do Uso de IA

É muito comum pensar que os usos de IA existem apenas na teoria e em temas de estudo. Porém, uma pesquisa [11] realizada durante um evento da ConFab, no ano de 2017, questionou se os hackers criminosos utilizariam IA em seus ataques no próximo ano, e a análise revelou que 62% deles responderam que sim. Isso mostra que, mesmo lentamente e sigilosamente, a IA está cada vez mais "fora do papel", e ainda espera-se um crescimento ao longo do tempo.

Em 2016, dois cientistas de dados da empresa de segurança ZeroFOX [8], conduziram um experimento prático, que visava mensurar o poder de uma Inteligência Artificial em relação aos humanos. A experiência foi feita na rede social "Twitter" com o objetivo de forçar os usuários a clicarem em links maliciosos. Para isso, os pesquisadores ensinaram uma Inteligência Artificial a estudar o comportamento das pessoas nessa rede e, então, arquitetar e implementar uma isca de phishing própria. Como resultado, mais uma vez a IA provou ser mais eficiente do que os humanos, compondo e distribuindo "tweets" muito mais convincentes e adaptados a cada usuário alvo do algoritmo.

Inteligência Artificial na Segurança Digital

Considerando a mesma amplitude dos ataques, a Inteligência Artificial também é muito útil para o desempenho de tarefas benéficas pois auxilia no desenvolvimento de soluções no campo da segurança cibernética, reforçando que a IA pode intervir e automatizar processos que são muito mais lentos quando feito por humanos.

As utilidades da IA não se limitam a criar amostras de malwares ou mecanismos de detecção de programas maliciosos, mas, com ela, também é possível desenvolver algoritmos inteligentes que capturam possíveis versões futuras de um malware, ou outras variantes com códigos semelhantes ao dele. Além disso, mesmo que de maneira indireta, a IA também auxilia no trabalho dos pesquisadores que, ao invés de estarem fazendo as tarefas que ela executa, podem dedicar mais tempo em análises profundas de outras ameaças mais perigosas.

Atualmente, a IA para evitar ataques ainda vem sendo estudada e, com isso, existem poucas implementações realmente funcionais. Segundo o professor de segurança e diretor da empresa CyLab [7], serão necessários no mínimo 10 anos para usarmos totalmente a IA como um grande general da segurança dos computadores, porque esses algoritmos aprendem conforme vão adquirindo experiência, com tentativa e erro, ou seja, eles vão evoluindo em função do tempo. Sendo assim, a presença humana ainda é imprescindível nesse primeiro momento de evolução da cibersegurança e, além disso, não se pode descartar o fato de que as IAs não são perfeitas, e um recurso humano pode sempre ser necessário caso existam falhas.

Mesmo com grandes prós, os pesquisadores não deixam de considerar os pontos negativos de utilizar IA para evitar ataques. O investimento em procurar falhas antes do algoritmo ser implementado sempre é alto pois, para que ele funcione, é

preciso alimentá-lo com um enorme fluxo de dados que podem incluir incluem informações privadas e pessoais que são valiosos para criminosos e hackers.

4.6 Limitações da Inteligência Artificial

Quando fala-se em IA, pode surgir um pensamento de que, por ser um programa, ela pode ser adaptada para realizar toda e qualquer atividade, obtendo 100% de sucesso, porém não é bem assim.

O uso da Inteligência Artificial ainda possui uma série de limitações. Por se tratar de um algoritmo que segue determinado padrão na análise de dados em busca da solução para determinado assunto, é notável que, caso ocorra uma modificação nesse padrão, o algoritmo irá falhar. Isso mostra que é necessário ter uma interferência humana para que possa ocorrer uma análise da informação fornecida pela IA, verificando a proximidade com a resposta esperada para evitar possíveis erros.

Seguindo esse mesmo pensamento, uma IA só se desenvolve por meio da análise de dados concretos, o que significa que ela não é capaz de fazer previsões. Dito isso e trazendo a situação de pandemia atual, não seria possível fazer com que uma Inteligência Artificial previsse o momento exato em que a humanidade fosse enfrentar uma pandemia, não existem dados pelos quais é possível fazer tal inferência.

Além disso, assim como os seres humanos, a IA pode ser enganada. Para ilustrar, Carla Martínez [16] dá um claro exemplo: suponha que, por algum motivo, uma pessoa utiliza o computador de um outro alguém para fazer uma pesquisa no Google sobre bares. Pelo uso de cookies, o Google Ads pega essa informação e fará com que apareçam propagandas de bares para o dono do computador utilizado, mesmo que não seja de seu interesse, mostrando que é possível enganar uma Inteligência Artificial.

Ademais, muitas pesquisas dizem que, com o passar dos anos, os empregos irão, cada vez mais, sofrer uma redução. Porém, pelo fato de a IA não ser independente, ou seja, não consegue funcionar sozinha, é provável que, mesmo com a automatização, ocorra o surgimento de novas profissões.

4.7 Impactos da Inteligência Artificial

É visível que, de alguma maneira, a IA acabe impactando a sociedade como um todo, trazendo consequências capazes de serem sentidas por todas as pessoas, podendo elas serem boas ou ruins.

Tratando de alguns impactos positivos do uso das Inteligências Artificiais, pode-se falar em: melhora na eficiência do trabalho, como dito anteriormente, quando uma etapa é automatizada, as pessoas acabam tendo mais tempo para focar em partes que demandam maior cautela; recursos de monitoramento e diagnóstico, capazes de reduzir os custos e melhorar a área da saúde, podendo fornecer tratamentos mais individuais e personalizados; por último, a IA pode ser aplicada na Ciência Forense, mudando a maneira de solucionar os crimes, o uso do reconhecimento facial é um exemplo de como esse ramo pode ser mais eficiente com seu uso.

Contudo, também existem alguns impactos negativos que são trazidos com o desenvolvimento e amplo uso da Inteligência Artificial, são eles: diminuição no número de empregos, fruto da crescente automatização fornecida; se utilizada de maneira maliciosa, acaba fazendo com que os ataques hackers sejam mais bem sucedidos, implicando na ameaça da segurança digital; por fim, com a diminuição dos empregos, essa tecnologia influencia no aumento da desigualdade socioeconômica pelo mundo.

Assim sendo, pode-se concluir que fazer o uso da tecnologia em questão não traz apenas impactos benéficos para a sociedade. Quando for aplicá-la em alguma área, é necessário fazer uma análise das consequências que serão geradas para ver se realmente vale a pena.

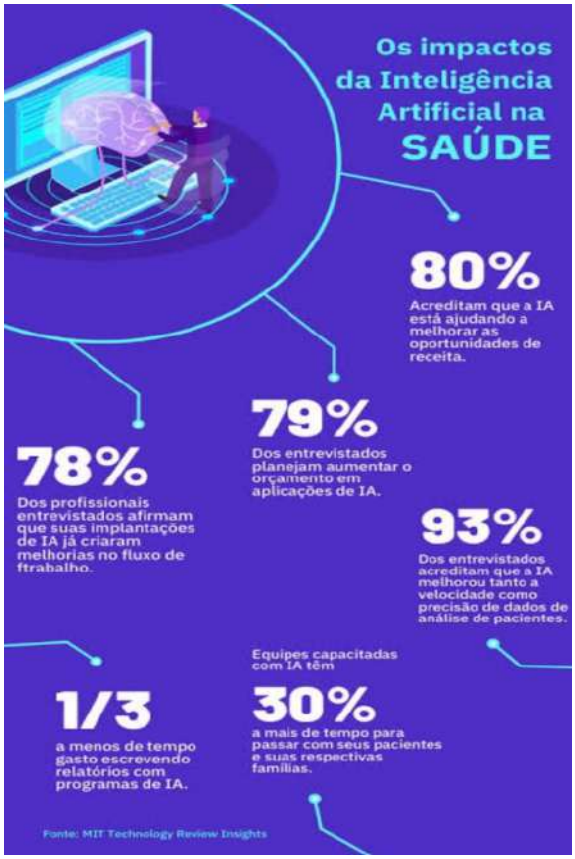


Figura 4.2: Impactos da IA na Saúde [1]

4.8 Curiosidades Sobre Inteligência Artificial

Até aqui, foram apresentados conceitos e explicações sobre como a IA funciona, bem como algumas aplicações e limitações que ela possui. Agora será introduzido um tópico interessante relacionado a curiosidades que, talvez, a maioria das pessoas nunca tinha pensado sobre.

Se for feita uma análise, será possível perceber que a maioria das IAs consideradas BOTs, as assistentes virtuais, são mulheres, como a Siri, por exemplo. Isso se deve ao fato de terem percebido que, no geral, tanto os homens quanto as mulheres, sentem-se mais atraídos por uma voz feminina.

Além disso, existem pessoas trabalhando no desenvolvimento dos chamados "pet-bots"que serão de animais de estimação robôs, ou seja, terão a mesma aparência dos convencionais, porém não precisam ser alimentados e não morrem, por exemplo.

Outras coisas interessantes que podem ser feitas pela IA [9] são: avaliação de estabelecimentos como restaurantes em plataformas como TripAdvisor, por exemplo; escrever livros; realizar a edição de vídeos; compor músicas, Taryn Southern é uma artista que fez o uso da IA para compor um álbum chamado "I AM AI"; e até sonhar, isso ocorreu com a IA do Google, gerando imagens durante seu sonho digital.

Somado a isso, com os avanços nos estudos da inteligência artificial, com ela também é possível reconhecer vozes. Se perguntarmos "Quem sou eu?"às assistentes virtuais, como a Siri, por exemplo, elas são capazes de reconhecer sua voz, que foi configurada anteriormente, e dizer seu nome.

4.9 Incógnitas Relacionadas à IA

Além de tudo o que já foi citado anteriormente, existem uma série de assuntos que ainda não foram resolvidos e que são difíceis de se chegar a uma conclusão.

Como exemplo, pode ser citado o que deve acontecer quando uma IA falha, quem deve ser responsabilizado. Para exemplificar, pode-se criar uma situação hipotética na qual um carro que está dirigindo sozinho causa um acidente. É evidente que um computador não tem como assumir a culpa, logo, quem é o responsável?

Existe também a questão da manipulação social, com o uso de algoritmos, a IA consegue disseminar propagandas que acabam sendo selecionadas por ela, ou seja, podem ser verídicas ou não, o que fica complicado, já que pode ser utilizada para espalhar fake news, por exemplo.

Acaba sendo necessário tratar de assuntos como os valores da IA, quais deles devem ser incorporados à ela? Há ainda um obstáculo para que isso seja resolvido que seria o fato de não existir um consenso universal sobre os valores dos seres humanos, dependendo da região ou país, esses valores mudam. Com isso, quais deveriam ser apresentados para a IA?

Outro fato interessante de ser pensado é como evitar a discriminação, tão presente na sociedade atual, por parte da IA, evitando com que os al-

goritmos de contratação acabem desfavorecendo uma parcela da população. Mas, como isso deve ser feito e desenvolvido para que as minorias não sejam prejudicadas por tal tecnologia, é uma incógnita que precisa ser pensada.

4.10 Considerações Finais

Fazendo uma análise de tudo que foi dito ao longo deste artigo, é possível concluir que o campo da Inteligência Artificial é muito amplo e possui uma série de aplicações diferentes que podem ser utilizadas em prol do bem coletivo, facilitando atividades feitas no dia a dia, por exemplo. Porém, assim como a maioria das coisas, a IA também possui seus pontos bons e ruins, que devem ser cuidadosamente analisados antes de tomarem a decisão de utilizá-la de alguma maneira.

Além disso, também é possível perceber que se trata de um campo relativamente novo e que ainda tem muito a ser explorado, descoberto e estudado, diversas incógnitas são encontradas no que se diz respeito à Inteligência Artificial, o que mostra que ela ainda está em processo de estudo.

Por último, foi possível perceber que não estamos falando de algo extremamente maléfico como é exageradamente mostrado na maioria dos filmes em que robôs passam a ter controle sobre o mundo como um todo, e que dispositivos que fazem o uso dessa tecnologia estão cada vez mais presentes na sociedade e, muitas vezes, a maioria das pessoas nem se dá conta disso.

4.11 Bibliografia

[1] Como a inteligência artificial está tornando a saúde 'mais humana'. URL: https://www.startse.com/noticia/nova-economia/inteligencia-artificial-saude-mais-humana.

[2] Os 7 padrões da inteligência artificial. URL: https://neigrando.com/2020/08/16/os-7-padroes-da-inteligencia-artificial/.

[3] E. ALECRIM. Machine learning: o que é e porque é tão importante? URL: https://tecnoblog.net/247820/machine-learning-ia-o-que-e/.

[4] A. ARTISTS. Unanswered questions about ai. URL: https://aiartists.org/unanswered-questions.

[5] S. BAHRAMIR. 5 interesting facts you didn't know about ai. URL: https://www.quillit.io/blog-posts/5-interesting-facts-you-didnt-know-about-ai.

[6] BUILTIN. What is artificial intelligence? URL: https://builtin.com/artificial-intelligence.

[7] C. S. COMSTOR. Como a inteligência artificial pode ajudar a parar ciberataques? URL: https://blogbrasil.comstor.com/como-a-inteligencia-artificial-pode-ajudar-a-parar-ciberataques.

[8] G. DVORSKY. Hackers já criam inteligências artificiais para usá-las como armas. URL: https://gizmodo.uol.com.br/hackers-inteligencias-artificais-armas/.

[9] R. FARINACCIO. 17 das coisas mais loucas que a inteligência artificial já pode fazer. URL: https://www.tecmundo.com.br/ciencia/121734-17-coisas-loucas-inteligencia-artificial-fazer.htm.

[10] FONTEMIDIA. Trojan bancário emotet já possui mais de 30 mil variantes em todo o mundo. URL: http://www.fontemidia.com.br/var/www/html/fontemidia.com.br/web/index.php/15-principal/homepage-blog/457-trojan-bancario-emotet-ja-possui-mais-de-30-mil-variantes-em-todo-o-mundo.

[11] HEIDEN. Artificial intelligence can be useful to hackers, too. URL: https://www.heidentechnology.com/artificial-intelligence-can-be-useful-to-hackers-too/.

[12] B. IMPACTA. Afinal, o que é data science e como atuar na área? URL: https://www.impacta.com.br/blog/afinal-o-que-e-data-science-e-como-se-atuar-na-area/.

[13] LIFARS. Hacking with artificial intelligence. URL: https://lifars.com/2019/12/hacking-with-artificial-intelligence/.

[14] MALWAREBYTES. When artificial intelligence goes awry: separating science fiction from fact. 2019.

[15] B. MARR. What is the impact of artificial intelligence (ai) on society? URL: https://bernardmarr.com/default.asp?contentID=1828.

[16] C. MARTíNEZ. La inteligencia artificial llegó para quedarse y aunque tiene grandes alcances también cuenta con limitaciones y representa retos para las empresas y la humanidad en su conjunto. URL: https://www.telcel.com/empresas/tendencias/notas/limites-de-inteligencia-artificial.html.

[17] J. MCCARTHY. What is artificial intelligence? 2007.

[18] REUTERS. Novo ataque hacker usa inteligência artificial para atingir alvos. URL: https://noticias.r7.com/tecnologia-e-ciencia/novo-ataque-hacker-usa-inteligencia-artificial-para-atingir-alvos-08082018.

[19] A. SATURNO. Glossário inteligência artificial: Entenda os principais termos usados na área. URL: https://canaltech.com.br/inteligencia-artificial/glossario-inteligencia-artificial-125084/.

[20] C. STUPP. Fraudsters used ai to mimic ceo's voice in unusual cybercrime case. URL: https://www.wsj.com/articles/fraudsters-use-ai-to-mimic-ceos-voice-in-unusual-cybercrime-case-11567157402.

[21] TECMUNDO. A história da inteligência artificial. URL: https://www.institutodeengenharia.org.br/site/2018/10/29/a-historia-da-inteligencia-artificial/.

[22] M. THOMAS. 6 dangerous risks of artificial intelligence. URL: https://builtin.com/artificial-intelligence/risks-of-artificial-intelligence.

[23] TIINSIDE. Hackers exploram sistemas com ia para potencializar seus ataques. URL: https://tiinside.com.br/21/01/2019/hackers-exploram-sistemas-com-ia-para-potencializar-seus-ataques/.

[24] N. TYAGI. 6 major branches of artificial intelligence (ai). URL: https://www.analyticssteps.com/blogs/6-major-branches-artificial-intelligence-ai.

[25] A. WOODIE. Understanding the limits of ai. URL: https://www.datanami.com/2020/04/06/understanding-the-limits-of-ai/.

Capítulo 5

Uma Introdução à Criptografia

Matheus Fernando Vieira Pinto

Resumo

Este artigo tem por objetivo apresentar conceitos básicos relacionados à criptografia, descrevendo sua história, importância, seus tipos e funcionamento. Além de exemplos de ataques hackers com a utilização da criptografia.

Palavras-chave: Criptografia; Cifra; Chaves criptográficas; Texto claro; Mensagem cifrada.

5.1 Introdução

A criptografia, do grego *kryptós* (esconder) e *grapho* (escrita), é um conjunto de técnicas utilizadas para cifrar um texto claro (mensagem comum) com o objetivo de manter a segurança das informações transmitidas entre as diferentes pessoas ou organizações durante uma comunicação. No passado a criptografia era utilizada para propósito de guerra. A partir do século XX com o surgimento do computador e posteriormente a criação da internet a criptografia passou a ter extrema importância na proteção dos dados que trafegam na rede e sem ela dificilmente poderíamos manter nossos dados protegidos contra invasores.

5.2 A história da Criptografia

A criptografia é hoje amplamente utilizada em diferentes setores da sociedade, porém essa técnica não é nada nova, dados históricos revelam que as primeiras civilizações já faziam uso de alguma técnica para esconder informações confidenciais durante períodos de guerra. [3] No Império romano, em especial durante o governo do imperador Júlio César, uma técnica chamada Cifra de César foi muito utilizada para esconder dos inimigos as mensagens trocadas entre os generais do exército romano. No século XX durante o período da 2° Guerra Mundial, a Alemanha utilizou uma máquina mecânica chamada Enigma para criptografar dados sigilosos e evitar que eles fossem parar nas mãos de seus inimigos. Hoje, com o auxílio da matemática e da computação, a criptografia moderna passou a ser implementada por algoritmos computacionais que mantém os dados protegidos de maneira bastante eficiente em diferentes aplicações.

5.3 Tipos de Criptografia

A criptografia pode ser dividida em dois tipos principais sendo eles a criptografia simétrica e a criptografia assimétrica.

Criptografia Simétrica

Nesse tipo de criptografia os dados são encriptados e decriptados com a utilização de uma única chave criptográfica, ou seja, durante a troca de informações entre dois ou mais indivíduos a mensagem é cifrada pelo(s) emissor(es) com a chave criada pelo destinatário, e este, ao receber a mensagem a decifra utilizando a mesma chave. [4] Esse método é considerado rápido, porém ele é menos seguro, pois se o conteúdo da chave for obtido por cibercriminosos todos os dados con-

fidenciais trocados durante a comunicação serão descobertos.

Figura 5.1: Processo de criptografia simétrica. [1]

Criptografia Assimétrica

Diferente da criptografia simétrica a criptografia assimétrica é composta de um par de chaves criptográficas, uma delas é utilizada para encriptar, enquanto que a outra é utilizada para decriptar os dados. [4] Durante uma comunicação o destinatário é o criador de ambas as chaves, porém ele disponibiliza apenas a chave pública para o emissor, e com esta encripta a informação da mensagem e depois envia a mensagem cifrada para o destinatário, e com sua chave privada, decripta a mensagem. Esse método é lento quando comparado com a criptografia simétrica, pois geralmente os conteúdos das chaves são muito longos, mas em contrapartida ele é mais seguro. [1]

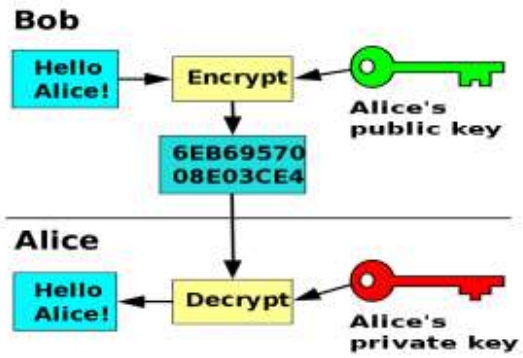


Figura 5.2: Processo de criptografia assimétrica. [9]

5.4 Algoritmos Criptográficos

Os algoritmos criptográficos são os responsáveis por encriptar um texto puro. Neste artigo serão apresentados três deles: a cifra de substituição, o RSA e a função hash.

Cifra de Substituição

Considerado um método de criptografia simples esse algoritmo do tipo simétrico realiza substituições de caracteres, no qual para cada caractere ou símbolo da mensagem clara, um novo caractere diferente é utilizado para substituí-lo e então gerar a mensagem cifrada. [8] Uma das aplicações mais conhecidas dessa técnica é a cifra de César, criada pelo imperador romano Júlio César com o objetivo de esconder as informações trocadas entre seus comandantes. Seu funcionamento consiste em deslocar uma certa quantidade de casas do alfabeto original e então substituir cada caractere da mensagem clara pelo seu representante no alfabeto deslocado. Na época esse algoritmo teve grande êxito, mas com o passar do tempo ele se tornou obsoleto, pois era fácil descobrir a informação cifrada com base em uma análise de frequência de caracteres e símbolos em um determinado idioma.

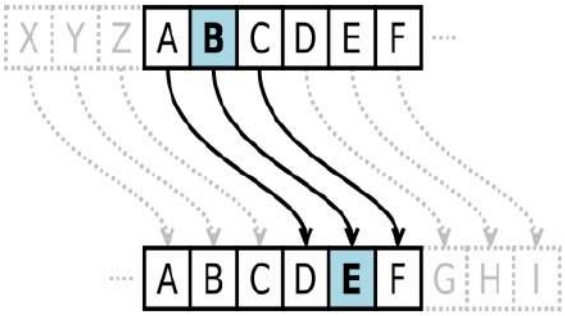


Figura 5.3: Representação do algoritmo Cifra de César. [7]

RSA

Criado em 1976 por Ron Rivest, Adi Shamir e Leonard Adleman esse algoritmo é considerado o precursor da criptografia assimétrica. Essa técnica é de extrema importância na atualidade já que atua diretamente sobre a internet. Seu fun-

cionamento consiste em criar um par de chaves criptográficas a partir da multiplicação de dois números primos grandes. [12] O processo de fatoração desse produto pode demorar centenas de anos para ser realizado por um computador e é por esse motivo que o RSA é considerada uma das técnicas de criptografias mais seguras da atualidade. Aplicações bancárias, sites de compras online, aplicativos de mensagens utilizam a criptografia RSA para manter os dados de seus usuários protegidos.

Função Hash

Uma função hash é um algoritmo matemático que mapeia um conjunto de dados em uma série de caracteres (resumo) de comprimento fixo, a saída sempre terá o mesmo comprimento independente do tamanho do conjunto de entrada. [11] A SHA-1 é uma função hash bastante utilizada na atualidade, ela converte qualquer dado de tamanho variável em um resumo fixo de 40 caracteres. O processo inverso de converter um resumo para uma mensagem clara não se aplica a essa técnica.

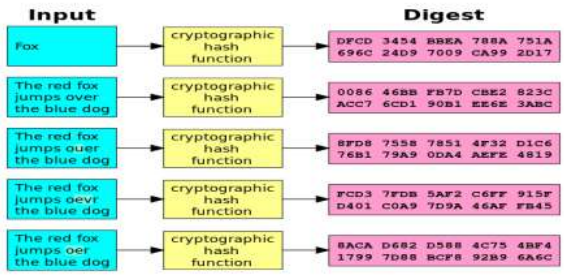


Figura 5.4: Funcionamento de uma função hash SHA-1. [11]

Muitas são as aplicações das funções hash entre elas podemos destacar a verificação de senha e a assinatura digital.

5.5 Assinatura Digital

A assinatura digital é uma técnica que utiliza a criptografia assimétrica para garantir que uma informação transmitida durante uma comunicação seja única e inalterável de modo a garantir a autenticidade da informação. [10] Seu funcionamento consiste em codificar um conjunto de dados com o uso de uma função hash que em seguida é combinada com a chave privada do emissor. Quando a mensagem é entregue ao destinatário uma comparação entre o resumo gerado pela função hash e a chave pública do emissor é realizada para então verificar a integridade da mensagem. O par de chaves criptográficas e a assinatura digital são obtidos através de certificados digitais.

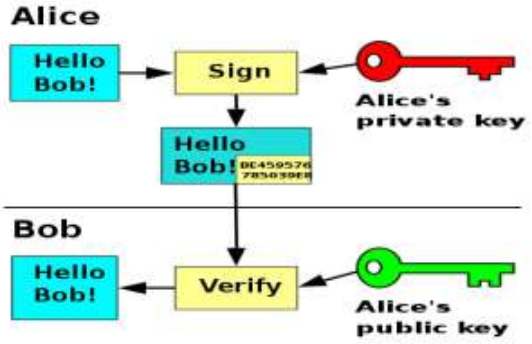


Figura 5.5: Funcionamento de uma assinatura digital. [10]

5.6 Ataques Hackers com o uso de Criptografia

Em 2017 uma ameaça chamada *WannaCry* invadiu de maneira mundial a rede de computadores, seus principais alvos eram as máquinas operadas pelo sistema Microsoft Windows. O objetivo principalmente desse cripto-ransomware era encontrar dados importantes e de valor para os usuários e encriptá-los utilizando técnicas de criptografia conhecidas, dessa maneira os afetados não poderiam ter acesso aos seus arquivos pessoais. Os dados só seriam liberados se as vítimas realizassem pagamentos em Bitcoins assim como solicitados pelos hackers.

Meses após o ataque *WannaCry* um novo cripto-ransomware chamado *NotPetya* realizou ataques sistemáticos em diferentes organizações da Ucrânia e posteriormente dos Estados Unidos, Austrália e Europa. Seu funcionamento era ainda pior que o *WannaCry*, pois ele não criptografava apenas os dados do usuário e sim toda a máquina in-

fectada, impedindo o acesso da vítima ao sistema operacional do computador.[2]

5.7 Considerações Finais

A criptografia é uma técnica para esconder os dados de terceiros ela pode ser dividida em criptografia simétrica ou criptografia assimétrica. Diferentes técnicas de criptografia foram criadas desde o seu surgimento, algumas utilizadas para propósito de guerra e outras apenas para estabelecer uma comunicação segura. A partir do século XX, com o surgimento do computador e posteriormente da internet o fluxo de dados trocados entre as pessoas e organizações aumento de maneira significativa, para cobrir essa demanda técnicas avançadas de criptografias foram criadas com o auxílio computacional entre elas destaca-se o método RSA que é utilizado em sites de compras online, internet banking, criptomoedas e aplicativos de mensagens. A criptografia também pode ser utilizada para males assim como o ocorrido nos ataques *WannaCry* e *NotPetya* em 2017.

5.8 Bibliografia

[1] DEVMEDIA. Criptografia assimétrica: Criptografando e descriptografando dados em java. URL: https://www.devmedia.com.br/criptografia-assimetrica-criptografando-e-descriptografando-dados-em-java/31213.

[2] IG. Entenda por que o ataque com ransomware notpetya é mais grave que o wannacry. URL: https://www.google.com/amp/s/tecnologia.ig.com.br/2017-06-27/ransomware-notpetya-wannacry.html.amp.

[3] D. KAHN. *The Codebreakers - The Story of Secret Writing*. The Macmillan Company, New York 1972.

[4] C. Paar and J. Pelzl. *Understanding Cryptography - A Textbook for Students and Practitioners.* Springer, 2010.

[5] CANAL TECH. Criptografia para iniciantes: o que é, como funciona e porque precisamos dela? URL: https://www.google.com/amp/s/canaltech.com.br/amp/seguranca/criptografia-para-iniciantes-o-que-e-como-funciona-e-por-que-precisamos-dela-46753/.

[6] TECHTUDO. O que é criptografia? URL: https://www.google.com/amp/s/www.techtudo.com.br/artigos/noticia/2012/06/o-que-e-criptografia.amp.

[7] WIKIPEDIA. Cifra de cesar. URL: https://pt.wikipedia.org/wiki/Cifra_de_César.

[8] WIKIPEDIA. Cifra de substituição. URL: https://pt.wikipedia.org/wiki/Cifra_de_substituiç~ao.

[9] WIKIPEDIA. Criptografia de chave pública. URL: https://pt.wikipedia.org/wiki/Criptografia_de_chave_pública.

[10] WIKIPEDIA. Digital signature. URL: https://en.wikipedia.org/wiki/Digital_signature.

[11] WIKIPEDIA. Função hash criptográfica. URL: https://pt.wikipedia.org/wiki/Funç~ao_hash_criptográfica.

[12] WIKIPEDIA. Rsa (sistema criptográfico). URL: https://pt.wikipedia.org/wiki/RSA_(sistema_criptográfico).

Capítulo 6

Concepção de Jammer de pulso eletromagnético em dardos ou flechas I

Lucas Martins Silva
lucas.silva@dcomp.sor.ufscar.br

Resumo

Este trabalho apresenta as concepções teóricas e o desenvolvimento da pesquisa com relação à possível montagem de um *jammer* de pulso eletromagnético sobre uma flecha ou dardo, de modo a utilizar os atributos disruptivos dos dispositivos de *jamming*, aliados ao alcance e agilidade de um projétil. Serão evidenciadas informações de forma a consolidar uma análise de viabilidade, bem como dados gerais acerca de assuntos correlatos ao projeto, como segurança, nuances de física elétrica, arquearia e *jamming* geral.

Palavras-chave: flecha; arquearia; *jamming*; pulso eletromagnético; PEM.

6.1 Introdução

Com sua origem datada no paleolítico [6], o uso do arco e flecha foi observado ao longo de toda a história da humanidade, sendo esta arma extremamente dominante nos confrontos bélicos, caçadas e esportes até o surgimento e popularização da pólvora e das armas de fogo. Não é estranho notar, portanto, que a arquearia tornou-se campo de estudo e admiração, avançando como arte e ciência.

Em tempos atuais, o uso do arco e flecha foi socialmente redefinido como majoritariamente civil, visto que, do advento das armas de fogo, notou-se a forte obsolescência do armamento que, já de tão longa data, acompanha a espécie humana. Em seu uso civil, os disparos de flechas são voltados à prática de tiro de precisão(como esporte), atividades de recreação e caçadas em regiões e países permissivos.

No entanto, a característica de possuir projéteis alteráveis dá ao arco e flecha, ainda hoje, uma versatilidade admirável, sendo este trabalho um esforço na direção de realizar demonstração e aproveitamento de tal característica. Para tanto, iniciase o estudo de uma alteração de projétil consistindo em um *jammer* de pulso eletromagnético, a ser utilizado, por exemplo, para desarmar artefatos perigosos com componentes eletrônicos sem a necessidade de aproximação de operadores humanos (ideia semelhante à de utilização atual de fuzis anti-materiais). Outra utilidade seria a de interceptar o funcionamento de drones, conhecidamente utilizados como transportadores de recursos para prisioneiros em penitenciárias.

Em essência, um *jammer* de pulso eletromagnético é um dispositivo simples, composto geralmente por uma espira condutora por onde trafegará uma corrente elétrica criada por uma diferença de potencial. A passagem da corrente é acompanhada da formação de um campo eletromagnético nos arredores da área do condutor [9], sendo este campo responsável por interferir no funcionamento de dispositivos eletrônicos

nesta área. O efeito é notavelmente relevante para dispositivos computacionais comerciais, visto que tais dispositivos são geralmente compostos por semicondutores de óxido metálico, altamente sensíveis a perturbações de alta tensão [5].

A utilização de *jammers* na atualidade tem uma popularidade controversa, sendo, em muitos países (como o Brasil [2]), regulamentados por lei e tendo sua utilização descontrolada definida como crime. Desta forma, o desenvolvimento desta pesquisa tem suas bases em propostas de análise científica, bem como de utilização de uma aparente sinergia entre armamento antiquado e tecnologia moderna.

6.2 Jamming e suas variantes

A prática de gerar condições disruptivas no funcionamento de dispositivos eletrônicos, popularmente conhecida como *jamming*, é composta por estratégias diferentes e se refere não apenas às disrupções causadas por dispositivos geradores de pulso eletromagnético. Na verdade, o conceito de *jamming* como um todo pode ser aplicado a quase qualquer tipo de sabotagem, mas é corriqueiramente destinado somente aos casos em que a prática é realizada tendo foco de ataque em dispositivos tecnológicos atuais.

Dentre os tipos de *jamming*, um dos mais comuns dentro da contemporaneidade é o de disrupção de comunicação entre dispositivos que se utilizam de bandas de radiofrequência para realizar comunicação. Geralmente, a interrupção na comunicação é conseguida através da propagação excessiva de ruído dentro dos intervalos de frequência utilizados na comunicação entre os dispositivos que se deseja afetar. É possível causar este efeito utilizando transmissores de rádio e antenas convencionais [4]. Este tipo de *jammer* ficou relativamente conhecido no Brasil dado ao seu uso por ladrões de carros com o intuito de desestruturar rastreadores de GPS presentes em veículos roubados. Além disso, popularmente conhecidos no país como "capetinha" ou "chupa-cabra", dispositivos semelhantes foram utilizados para gerar interferência nos aparelhos de fiscalização eletrônica automática em rodovias.

Outro tipo de *jamming*, porém de muito maior escala, é observado como armamento bélico de alguns países. Este *jamming* de longo alcance pode ser realizado de diferentes formas. A mais antiga delas é a utilização da explosão de uma bomba nuclear em alta altitude, que termina por gerar radiação gama capaz de promover o surgimento de um intenso e instantâneo campo eletromagnético, que é inofensivo às pessoas, mas potencialmente mais destrutivo a circuitos computacionais que o efeito da descarga de um relâmpago [8], tal estratégia é convencionalmente chamada HEMP, um acrônimo para pulso eletromagnético de grande altitude. A segunda e mais recente forma, consiste em um aparato bélico capaz de projetar ondas eletromagnéticas de origens diferentes, mas que confluem a um mesmo ponto, sendo seus geradores satélites em órbita [1].

Por fim, a forma mais antiga de *jamming* eletrônico e também o foco deste trabalho consiste na utilização de um pulso de campo eletromagnético gerado pela passagem de uma corrente em um condutor a fim de obstruir o funcionamento de dispositivos eletrônicos, principalmente daqueles com grande presença de circuitos computacionais. Este método é comumente chamado de pulso eletromagnético HPM, um acrônimo para micro-ondas de alta potência. Este método também aparece com a utilização de explosivos químicos de menor escala, havendo também, atrelada a tal reação, o aparecimento de um campo eletromagnético de alta intensidade e curta duração.

O problema do Jammer em flecha

É necessário ressaltar que, dentre as variantes de *jammer* previamente referidas, foi escolhida a estratégia de disrupção por eletromagnetismo, visto que, dada a existência no mercado de *jammers* de radiofrequência detentores de tamanho reduzido, o problema da criação de um projétil alterado para esta modalidade se direciona, praticamente em sua totalidade, ao problema de montagem e estabilização da flecha. Tendo em vista tal aspecto, julgou-se cientificamente mais produtiva a formulação de disrupção de pulso eletromagnético, visto que conceitos aplicados

a esta modalidade serão aplicáveis também à técnica de *jamming* por radiofrequência.

A fim de formular com maior precisão o problema de confeccionar um disruptor de PEM montado sobre uma flecha ou dardo, serão convencionadas algumas definições. Os projéteis de um arco e flecha podem, segundo tais convenções, serem classificados entre flechas portadoras, flechas de gatilho e flechas de lesão.

Flechas de lesão se referem à maior parte das flechas utilizadas por seres humanos, sendo seu objetivo primário o de causar danos em tecido vivo através de perfuração e corte. Geralmente a ponta ou cabeça da flecha determina a maior parte de sua funcionalidade e comportamento em impacto. Dentro de tais flechas também podem ser alocadas as flechas de treino e de tiro desportivo, que, por sua vez, buscam perfurar um alvo inanimado.

Flechas portadoras se referem à categoria de flechas alteradas para transportar objetos para alguma finalidade. Para tanto, flechas deste tipo sacrificam precisão de ajuste fino, sendo disparadas para atingir áreas e não alvos específicos. Pode-se visualizar o conceito de flecha portadora ao se analisar a concepção de uma flecha portando o já mencionado gerador de pulso eletromagnético. Por conseguinte, a finalidade principal de uma flecha portadora é, não surpreendentemente, o transporte.

Flechas de gatilho se referem à um pequeno grupo de flechas alteradas para provocar uma reação em um local suscetível. Dentro deste grupo, são alocadas flechas de finalidades semelhantes às das flechas incendiárias, utilizadas para desencadear incêndios ao atingir locais inflamáveis. Flechas de gatilho são convenientemente interessantes quando combinadas com flechas portadoras. Desta forma, o escopo deste projeto pode ser definido como o problema de formulação de uma flecha portadora cujo objeto transportado é um *jammer* de pulso eletromagnético. Para tal, é preciso considerar algumas limitações e características desejadas.

Dentre as limitações, a primeira a ser observada é a de que a flecha portadora desejada necessita de comprimento relativamente elevado. Tal limitação foi analisada ao início do desenvolvimento deste trabalho, tendo como uma de suas consequências a impossibilidade de utilizar uma besta de mão como arma propulsora para os projéteis alterados. Isso é justificável através da percepção de que os aparatos cogitados para a formulação do gerador de pulso comporiam um peso superior ao do próprio dardo. No entanto, testes utilizando a besta de mão podem ser realizados para estudo prático de aerodinâmica em hipóteses futuras.

Outra limitação a ser observada se encontra na impossibilidade de utilização de uma bateria única para alimentação do circuito desejado. Isso ocorre pois as menores baterias capazes de gerar a tensão elétrica desejada (por volta de 3.6 Volts) apresentam peso relativamente elevado em um espaço concentrado não distribuível ao longo do corpo da flecha, fato capaz de alterar drasticamente o balanceamento de peso e padrão de voo do projétil.

Ademais, como já mencionado anteriormente, sendo possível classificar o projétil a ser concebido neste projeto como uma flecha portadora, é observável, como uma de suas características, a redução da precisão nos disparos. Isso ocorrerá em virtude das alterações de balanceamento de peso da flecha, bem como da alteração de suas propriedades aerodinâmicas. Por conseguinte, é conveniente reduzir o alcance efetivo cogitado para disparos com este equipamento, a saber, para em torno de 10 a 15 metros de distância do atirador.

Consideradas as peculiaridades e limitações previamente mencionadas, o problema para este trabalho pode ser formulado como o do desenvolvimento teórico e, possivelmente, prático de uma flecha portadora de um circuito elétrico capaz de gerar um pulso eletromagnético, e capaz de manter precisão mediana dentro de uma distância idealizada de 10 a 15 metros a partir do atirador.

Resoluções e design

Das fortes limitações observadas, certa miríade de opções de design de equipamento foram

cogitadas, sendo esta seção destinada à discussão e apresentação de informações acerca destas opções. O principal foco da discussão revolverá em torno das características do projétil, ainda que algumas partes sejam dignas de menção no que tange à participação do armamento propulsor no desempenho esperado.

Em primeira menção, é necessário ressaltar que o armamento propulsor a ser escolhido para receber foco no desenvolvimento do trabalho consiste no arco, visto que, de maneira geral, é o dispositivo que apresenta capacidade para munições de maior escala. Em outras palavras, o tamanho médio de flechas para arcos é relativamente superior ao de dardos de bestas (bem como seu tamanho máximo), sendo sua maior extensão e consequente peso características desejáveis à estrutura do projeto.

Tendo novamente em vista os fatores limitantes, foi cogitada uma alternativa às opções de bateria única. Aproveitando-se das características da ligação de baterias em série, serão utilizadas várias baterias combinadas em série, a fim de atingir a tensão nominal desejada e esperada para o escopo do projeto. O tamanho específico das baterias ainda não foi determinado, mas a fragmentação do componente gerador de tensão será importante para que o peso possa ser distribuído de maneira mais uniforme ao longo do corpo da flecha. Para tal finalidade, o uso de várias baterias menores acopladas em diferentes posições do corpo da flecha e conectadas entre si em série pode se provar fundamental na manutenção do equilíbrio de peso sem grandes perdas na tensão nominal.

Ainda dentro do tópico de balanceamento de peso no projétil, é importante observar algumas características da física de arquearia capazes de fornecer ferramentas e direcionamentos na concepção do design desejado. Uma destas características pode ser apontada na posição do centro de massa da flecha, responsável por notáveis alterações no padrão de voo do disparo. Comumente conhecido como FOC (*Front-of-center*), o coeficiente geralmente utilizado para análise de distribuição de massa de uma flecha denota a porcentagem do peso total da flecha disposta na parte posterior do objeto, ou seja, na parte frontal da seta. O voo de um projétil com um alto FOC é caracterizado por manter a estabilidade e a trajetória do disparo, sendo, no entanto, afetado mais rapidamente pela queda propiciada pela gravidade. Em contrapartida, flechas com baixo FOC são conhecidas por trajetórias mais volúveis, mas que se mantém no ar por mais tempo [3].

Dadas as características observadas de centro de massa, a estratégia adotada para este trabalho consiste na distribuição do peso na flecha de modo que seja obtido um alto FOC. Isso significa concentrar a maior parte do peso na parte frontal da seta, o que dará maior estabilidade e precisão para os disparos dentro do alcance especificado de 10 a 15 metros. Para realizar o balanceamento correto, é possível a realização da troca da ponta da flecha, visto que o peso da ponta é relativamente elevado se comparado ao restante do projétil.

Outra peculiaridade digna de menção dentro do design do projeto é a do fenômeno conhecido como o paradoxo do arqueiro, responsável pela trajetória inesperadamente consistente de uma flecha ao ser disparada pelo arco. Tal fenômeno ocorre devido à elasticidade do corpo da seta. Ao ser impulsionada pela corda, uma flecha oscila lateralmente em sua movimentação, em movimento semelhante ao da locomoção de um peixe [7]. Tal oscilação é necessária para manter precisão dentro dos disparos e se traduz para o projeto como uma característica geral a ser mantida. Desta forma, é importante considerar que a disposição de fios, da espira e das baterias, não deve ser capaz de criar relativa rigidez no corpo da flecha, limitando, portanto, a extensão e número de voltas de fio ao redor do corpo da flecha.

Por fim, há ainda um ponto a ser explorado dentro do campo da arquearia, a ser considerado como estratégia de estabilização adicional em caso de observação de voos impróprios das setas. Este ponto reside na disposição das rêmiges das flechas, sendo possível a criação de um movimento de giro no projétil através da disposição orientada das rêmiges.

O giro tem a característica de prover estabilidade em detrimento da velocidade e distância percorrida. Como a distância almejada para o equipamento gira em torno de 10 a 15 metros, tal ônus é aceitável dentro do escopo deste trabalho.

Desta forma, além dos já mencionados materiais e estratégias, vale ressaltar a presença de um componente elétrico adicional para este primeiro design. Tal componente se refere a um gerador/conversor de alta tensão, capaz de amplificar a tensão gerada pelas baterias.

A idealização inicial é composta por um circuito a ser fechado no momento em que o pulso eletromagnético é desejado, estando o momento da ativação ainda em deliberação e estudo. Algumas opções simples são a de ativação no impacto da flecha e ativação anterior ao disparo.

Conclui-se, portanto, a idealização básica do design para a flecha portadora desejada, bem como a apresentação das discussões e justificativas a corroborar com as decisões tomadas. Há ainda extenso campo de aprimoramento e estudo, a ser o foco da continuidade do trabalho.

6.3 Estado atual e continuidade

Dada a condição de pandemia e seu rigor encontrado em território brasileiro no momento do desenvolvimento deste trabalho, são dificultados os testes práticos com equipamento já atualmente disponível. Como foi mencionado, a utilização de uma besta de mão para extrapolar condições de aerodinâmica é um dos pontos de experimentação a se explorar dentro do projeto. Além disso, a obtenção de equipamento confiável, assim como de espaço para realização de testes (dada a característica destrutiva do pulso eletromagnético) são empecilhos à parte da experimentação desejada.

No entanto, há ainda extenso espaço para o aprofundamento teórico no âmbito desta pesquisa, visto que há nuances e informações desejadas e ainda não exploradas dentro dos assuntos envolvidos. O direcionamento da pesquisa, a partir de então, terá por foco entender alguns funcionamentos de dispositivos selecionados, bem como tornar cientificamente mais previsíveis os fenômenos a serem observados nos testes práticos futuros. Dentro desta perspectiva, está, por exemplo, o problema de determinar previamente a extensão, alcance e relativa densidade do campo eletromagnético ao redor do campo de impacto (ou alvo).

Aliada à busca teórica, haverá continuidade na análise de equipamento disponível e obtenível, de forma a possibilitar validação ou alteração nos âmbitos de design estipulados até então, conforme eficiência e disponibilidade. Desta forma, este texto se consolida como o primeiro passo em direção à concepção do objeto desejado, formulando as bases sobre as quais as pesquisas futuras se assentarão.

6.4 Conclusões

A criação de uma flecha portadora aliada a um gerador de pulso eletromagnético se apresenta, até o momento, como factível, fato sustentado pelas pesquisas feitas até o momento e discussões e aplicações de conceitos relacionados à física que se consolida na união de duas áreas relativamente bem trabalhadas pela humanidade.

Há ainda grande espaço de desenvolvimento a fim de tornar mais precisos e significativos os testes práticos que possivelmente serão realizados. Desta forma, conclui-se o presente trabalho tendo em vista sua continuidade e aperfeiçoamento.

6.5 Bibliografia

[1] United States Paten (10). Patent no.: Us 8,785,840 b2 - apparatus for producing emp.

[2] Anatel. Resolução nº 308, de 11 de setembro de 2002. URL: https://www.anatel.gov.br/legislacao/resolucoes/2002/257-resolucao-308.

[3] Easton Archery. What is f.o.c. and how does it affect arrow flight? URL: https://eastonarchery.com/2014/06/foc/.

[4] P. Buckley, B.Bush, G.Coen, and C.Reis. Hack-a-thon:bb-rc jammer. URL: https:

//shareok.org/bitstream/handle/11244/302142/oksd_coen_HT_2018.pdf?sequence=1&isAllowed=y.

[5] C.N. Ghosh. *EMP weapons, Strategic Analysis*. 2000.

[6] George Agar Hansard. *The Book Of Archery: Being The Complete History And Practice Of The Art, Ancient And Modern*.

[7] Real World Physics Problems. Physics of archery – archer's paradox. URL: https://www.real-world-physics-problems.com/physics-of-archery.html.

[8] Clay Wilson. Crs report for congress, high altitude electromagnetic pulse (hemp) and high power microwave (hpm) devices: Threat assessments. Technical report.

[9] Hugh D. Young and Roger A. Freedman. *Física III - Eletromagnetismo*.

Capítulo 7

Concepção de Jammer de pulso eletromagnético em dardos ou flechas II

Lucas Martins Silva
lucas.silva@dcomp.sor.ufscar.br

Resumo

Este trabalho apresenta a continuidade do desenvolvimento da pesquisa com relação à possível montagem de um *jammer* de pulso eletromagnético sobre uma flecha ou dardo, de modo a utilizar os atributos disruptivos dos dispositivos de *jamming*, aliados ao alcance e agilidade de um projétil. Serão evidenciadas as descobertas e decisões referentes ao projeto, bem como justificativas para os caminhos seguidos. Neste trabalho também poderão ser encontradas novas interpretações úteis a quaisquer futuros interessados em desenvolvimentos correlatos

Palavras-chave: flecha; arquearia; *jamming*; pulso eletromagnético; PEM.

7.1 Introdução

Dado o interesse no aproveitamento da notável característica do arco e flecha como um armamento de projéteis alteráveis, foi realizada pesquisa na direção da concepção de um *jammer* de pulso eletromagnético acoplado a uma flecha, sendo parte dos conhecimentos obtidos e aplicados no escopo deste projeto também aplicáveis a dardos de bestas ou mesmo projéteis semelhantes de maior escala. Certo progresso foi feito e neste texto se concentram informações acerca do prosseguimento da já referida pesquisa, que visa aliar tecnologia atual a um armamento antigo [6].

Sendo este o segundo documento referente ao projeto, espera-se do leitor a leitura da primeira publicação, visto que parte dos conceitos será retomada, enquanto outra parte será presumida conhecida para fins de fluidez textual e redução da carga de leitura.

É necessário retomar no entanto, que, em essência, um *jammer* de pulso eletromagnético é um dispositivo simples, composto geralmente por uma espira condutora por onde trafegará uma corrente elétrica criada por uma diferença de potencial. A passagem da corrente é acompanhada da formação de um campo eletromagnético nos arredores da área do condutor [8], sendo este campo responsável por interferir no funcionamento de dispositivos eletrônicos nesta área. O efeito é notavelmente relevante para dispositivos computacionais comerciais, visto que tais dispositivos são geralmente compostos por semicondutores de óxido metálico, altamente sensíveis a perturbações de alta tensão [5].

Ademais, é necessário ressaltar que há vários tipos de *jammers*, sendo o escopo deste projeto voltado a um *jammer* baseado no pulso eletromagnético gerado por uma corrente em um condutor. É importante mencionar também, que a utilização de disruptores ou *jammers* em território brasileiro é regulamentada por lei e sua utilização desregulada é definida como crime [1]. Desta forma, o desenvolvimento desta pesquisa

tem suas bases em propostas de análise científica, bem como do aproveitamento das já mencionadas características do armamento a ser modificado para fins úteis à sociedade, como desarme de dispositivos perigosos, munidos de componente eletrônico, sem a necessidade de aproximação de um operador, ou mesmo desativação de drones em regiões onde não são permitidos (como penitenciárias e conglomerados semelhantes).

Por conseguinte, são evidenciados neste documento, recursos adicionais para a continuação do projeto de formulação de uma flecha portadora de um *jammer* eletromagnético. Espera-se também consolidar conceitos aplicáveis a outros tipos de flechas portadoras.

7.2 Adições teóricas

Seguindo o princípio adotado ao início deste estudo, novas pesquisas foram realizadas em campos da arquearia e da física, a fim de consolidar a viabilidade do projeto e alicerçar de maneira coesa, através do que se conhece da ciência referente às nuances do projeto, as concepções de design a serem adotadas para esta flecha portadora, bem como para trabalhos futuros de quaisquer interessados. Serão explicitados nesta seção os resultados de pesquisas adicionais relevantes ao projeto.

É necessário retomar, a princípio, o conceito do "paradoxo do arqueiro", que pode ser visto como a movimentação inesperadamente consistente de uma flecha ao deixar o arco, propelida pela corda [7]. Analisando de maneira mais profunda o conceito e as forças físicas envolvidas no fenômeno, podemos apontar algumas informações relevantes ao aprimoramento de voo de quaisquer flechas, constituindo, portanto, imprescindíveis dados à concepção de flechas portadoras, visto que os artefatos atrelados a esse tipo de flecha necessariamente alterarão questões físicas referentes a seu percurso no ar.

O fenômeno conhecido como paradoxo do arqueiro é fortemente relacionado à elasticidade do corpo da seta, sendo esta acometida por uma oscilação ao deixar o arco. Pode-se desmembrar esta oscilação em grandes duas forças, a força aplicada à flecha pela corda que a propele e a força da inércia presente no mesmo corpo. Ao ser propelida pela corda, as forças em contraste fazem com que o corpo do projétil seja deformado em uma curvatura, visto que a força de inércia concentrada na parte dianteira do artefato se opõe à força propulsora aplicada pela corda. O resultado é a oscilação que ocorre de forma análoga à movimentação de um peixe na água, sendo esta oscilação fortemente dependente da elasticidade do corpo da flecha.

As forças analisadas no paradoxo são responsáveis por diferentes efeitos na trajetória de um projétil a sofrer a propulsão de um arco. Se a força de propulsão do arco, transmitida pela corda, sobrepõe a força de inércia concentrada na parte posterior da seta, o efeito é uma deformação na trajetória da flecha ao deixar o arco, que se mantém ao longo de seu voo, consistente na alteração da trajetória para o lado esquerdo. Caso o oposto seja verdadeiro, ou seja, no caso em que força de inércia presente na ponta da flecha sobrepõe a força de propulsão proveniente da corda, a alteração de voo do projétil será observada no direcionamento da seta ao lado direito. Estas afirmações são válidas considerando um arco de postura destra, não composto. Em caso de postura canhota, os direcionamentos são invertidos dada a posição da seta em relação ao arco e à corda.

Tendo em vista tais conceitos, é importante apresentar os conceitos de espinha de flecha. Esta medida se refere à capacidade que um projétil de arco possui com relação à deformação. É chamada espinha estática de uma flecha a medida de desvio do corpo do artefato, quando submetido à pressão de um peso em um ponto considerado central para a flecha em posição de disparo [3]. A espinha estática refere-se, primordialmente, à constituição do corpo da seta, sendo um fator necessário à observação para ajustes de voo. É chamada espinha dinâmica uma aproximação da flexão real de uma flecha, sendo resultado das interações físicas oriundas da espinha estática, comprimento da seta e tamanho e peso de sua ponta. Quanto menor a espinha dinâmica, maior a flexibilidade da flecha, e mais suscetível ela se

apresenta à curvaturas para a esquerda.

Adicionalmente, para o escopo deste trabalho, é necessário ressaltar uma interessante propriedade do campo magnético gerado por uma corrente em uma espira ou bobina. Neste caso, é importante compreender que o direcionamento deste campo se atém ao plano perpendicular ao plano da espira ou bobina, sendo seu sentido determinado pelo fluxo da corrente. A partir de tais informações adicionais, é possível reaver o problema do *jammer* de PEM em uma flecha.

7.3 O problema do Jammer em flecha

Em consonância à publicação predecessora a este trabalho, busca-se formular o problema a ser enfrentado na concepção da flecha portadora de um disruptor de pulso eletromagnético. Consideram-se ainda, no escopo deste texto, as restrições e objetivos apresentados previamente, bem como somam-se àquele conjunto nova miríade de observações.

O desenvolvimento da flecha portadora a que se refere este projeto, tem, ainda como foco, a utilização efetiva em torno de 10 a 15 metros do equipamento a ser projetado. Por conseguinte, são esperadas situações imperfeitas de voo, a serem minimizadas pelos estudos feitos até o momento, mas capazes de garantir precisão dentro do alcance proposto. Ademais, é necessário ressaltar que o objetivo se reafirma no que tange à utilização do tipo de disruptor constituído pelo pulso eletromagnético gerado pela passagem de uma corrente em um condutor.

No entanto, após pesquisas realizadas em mercado real, pôde-se mensurar de maneira mais eficiente uma possível configuração de equipamentos e itens à constituir o aparato do projeto. Foram considerados, para a formulação do problema, dimensões e pesos correspondentes aos artefatos pesquisados, de forma a identificar possíveis falhas no design cogitado e em suas possíveis variantes. Ademais, as adições teóricas ao escopo do trabalho permitiram a visualização de novas restrições e detalhes, bem como alternativas à possíveis problemas encontrados após formulação dos primeiros protótipos. A saber, os conceitos mencionados previamente neste texto inspiram maior preocupação com relação ao comprimento e material da flecha, bem como a atribuição de pontas e seus diferentes tipos. Além disso, o estudo das características do campo eletromagnético requisitam decisão voltada ao posicionamento e orientação da espira ou bobina condutora, relacionando-se ao plano determinado pelo corpo da seta.

Por conseguinte, o problema para este trabalho persiste como a formulação de uma flecha portadora de um disruptor de pulso eletromagnético gerado por corrente, que seja capaz de manter precisão eficiente dentro do alcance de 10 a 15 metros de distância do atirador.

7.4 Resoluções e design

Das fortes limitações observadas, bem como opções encontradas, a inicial miríade de opções de design tornou-se mais restrita. Considera-se, no entanto, uma restrição proveitosa, baseada em suposições teóricas capazes de fornecer maior segurança ao desenvolvimento do sistema proposto.

As formulações se voltam, ainda, ao que pode ser alcançado utilizando-se os projéteis de um arco e flecha, preferencialmente recurvo ou longo, capaz de oferecer pouco empecilho aos lançamentos das flechas portadoras e seus adereços capazes de alterar sua aerodinâmica. Foram mantidas também as decisões referentes aos dispositivos eletrônicos(múltiplas baterias em série, disposição uniforme de peso, etc), bem como foram adiantadas pesquisas em mercado sobre opções viáveis de montagem de protótipo.

No que tange à arquearia e suas concepções teóricas, novos fatores foram adicionados ao escopo de estudo do trabalho, capazes de alterar significativamente as suposições anteriores. No entanto, parte das decisões de design propostas ainda podem ser mantidas, como o elevado FOC (*Front-of-Center*), caracterizado pela disposição maior de peso em parte posterior do projétil, sendo responsável por uma trajetória mais concisa, mas menos duradoura [4]. A utilização de

rêmiges espiraladas ainda se apresenta como possibilidade de estabilização do projétil em voo.

Dentre as adições teóricas referentes à arquearia, a concepção de espinha de flecha se mostrou fortemente ligada ao desempenho de voo da seta, o que direcionou ao tema grande foco de estudo. Dentre as descobertas a afetar o design da flecha, a mais importante se apresenta na correlação intrínseca entre a espinha dinâmica e o nível de desvio da flecha produzido pelo arco em posição de disparo. Desta forma, registra-se neste documento, para fins de futuros trabalhos e demais pesquisadores, que testes devem ser realizados para questões variáveis para cada arqueiro (desvio do arco, orientação destro-canhoto, tamanho de puxada, etc).

A formulação concreta do design ainda se aterá aos eventos e informações advindos dos testes com protótipos e produtos desejados, mas até o presente momento, boa parte do que foi previamente estipulado em design para o projétil persiste.

Dentre as alterações mais importantes para este trabalho, pode-se destacar a utilização da espira ou bobina condutora disposta de maneira a circundar o corpo da flecha, criando um campo eletromagnético direcionado à ponta da seta, voltado ao mesmo plano do corpo do projétil, sendo seu sentido determinado pelo sentido da corrente e passível de controle pelo formulador do sistema. Além disso, as concepções referentes à espinha de flecha resultaram, para o projeto, no direcionamento da disposição das baterias, orquestradas em ligação em série, para o centro da flecha, a fim de possibilitar menores alterações de espinha dinâmica.

O detalhamento em torno de equipamentos e produtos utilizados será tornado secundário no escopo deste trabalho, tendo em vista a variabilidade de fatores presentes para cada conjunto de artefatos utilizado por cada atirador. No entanto, espera-se ter apresentado fatores importantes no direcionamento de testes feitos por quaisquer outros interessados na formulação de flechas portadoras. Conclui-se, portanto, a idealização básica do design para a flecha portadora desejada, bem como a apresentação das discussões e justificativas a corroborar com as decisões tomadas. Há ainda extenso campo de aprimoramento e estudo, a ser o foco da continuidade do trabalho.

7.5 Estado atual e continuidade

Ainda dentro do período de pandemia global, é válido ressaltar que a disponibilidade de artefatos necessários à constituição do projeto se mostra defasada. No entanto, avanços foram realizados neste sentido e, é esperada a obtenção de um arco de 45 libras para início da realização de testes e prototipagem.

Além disso, é importante mencionar que o desenvolvimento do projeto e as pesquisas teóricas persistem, a fim de pavimentar o caminho da união de conceitos de tecnologia e arquearia para quaisquer outros pesquisadores que tenham o intuito de ingressar em projetos de mesmo cunho.

7.6 Indicações a futuros projetos

Considera-se benéfico, ainda dentro do escopo deste projeto, estimular formulações e aplicações reais referentes aos tópicos de flechas portadoras. Por conseguinte, nesta seção são descritas algumas recomendações àqueles interessados em formular flechas portadoras como a deste trabalho ou de demais variedades.

Inicialmente, recomenda-se a realização de avaliação teórica de viabilidade do projeto cogitado, baseando-se em informações referentes às peculiaridades do que se deseja portar na flecha (como o disruptor de pulso eletromagnético). Além disso, é importante considerar equipamentos disponíveis na região onde será desenvolvido o projeto.

Dentro dos tópicos de arquearia, ambos os trabalhos referentes a este tipo de flecha portadora podem trazer nuances passíveis de boa observação no desenvolvimento de um design apropriado para demais trabalhos correlatos. Sugere-se um curso de ação baseado no estudo em ordem, bem como decisão do que pode ser aplicado ao trabalho a ser desenvolvido, dos seguintes temas: funcionamento do tipo de arco

escolhido, treinamento e forma consistentes de tiro referentes à escolha anterior, consideração de disponibilidade e capacidade (o que é obtenível ao projetista no momento e suas capacidades físicas para manuseio do equipamento), possíveis ambientes de uso e formulação do problema (como o objetivo de se garantir precisão dentro de 10 a 15 metros), conceitos de *tuning* de flecha que se adéquem às escolhas e propostas anteriores (FOC, espinha, comprimento, peso, tipo de ponteira,etc) e, por fim, concepções de arquiteturas capazes de resolver os dilemas propostos e análise de possíveis vantagens e desvantagens, bem como pontos de aprimoramento.

Adicionalmente, é relevante ressaltar que os conceitos de arquearia são, atualmente, fortemente estudados e conceitualmente explicados de diversas maneiras em organizações e praticantes dentro da internet. Um exemplo, possivelmente útil a futuros projetistas, é o de uma calculadora online de espinha de flecha [2]. Reafirma-se, portanto, a necessidade do estudo teórico relacionado às concepções e aplicações de flechas portadoras, sendo este o alicerce principal deste mesmo projeto.

7.7 Conclusões

A criação de uma flecha portadora aliada a um gerador de pulso eletromagnético se apresenta ainda como factível, fato sustentado pelas pesquisas feitas até o momento e não desencorajada por novas descobertas. Pelo contrário, as novas adições teóricas constituem ferramental precioso na formulação de testes para o projeto, bem como sua posterior análise e possíveis alterações de design deles advindas.

Há ainda grande espaço de desenvolvimento a fim de tornar mais precisos e significativos os testes práticos que possivelmente serão realizados, bem como aperfeiçoar para cada caso a tomada de decisão para design, avaliando características de instrumentos e atiradores. Desta forma, concluise esta etapa do trabalho, tendo em vista sua continuidade e aperfeiçoamento.

7.8 Bibliografia

[1] Anatel. Resolução nº 308, de 11 de setembro de 2002. URL: https://www.anatel.gov.br/legislacao/resolucoes/2002/257-resolucao-308.

[2] 3Rivers Archery. Spine calculator. URL: https://www.3riversarchery.com/dynamic-spine-arrow-calculator-from-3rivers-archery.html.

[3] Easton Archery. Making sense of arrow spine. URL: https://eastonarchery.com/2014/07/making-sense-of-arrow-spine/.

[4] Easton Archery. What is f.o.c. and how does it affect arrow flight? URL: https://eastonarchery.com/2014/06/foc/.

[5] C.N. Ghosh. *EMP weapons, Strategic Analysis*. 2000.

[6] George Agar Hansard. *The Book Of Archery: Being The Complete History And Practice Of The Art, Ancient And Modern*.

[7] Real World Physics Problems. Physics of archery – archer's paradox. URL: https://www.real-world-physics-problems.com/physics-of-archery.html.

[8] Hugh D. Young and Roger A. Freedman. *Física III - Eletromagnetismo*.

Capítulo 8

Machine Learning: Uma breve introdução ao futuro

Matheus Vargas Volpon Berto
matheusvvb@hotmail.com

Vitor Ribeiro Guimarães Gomes
vitor.ribeiro0803@gmail.com

Resumo

Este documento apresenta uma caracterização geral dos principais conceitos envolvidos no processo de *Machine Learning*, além de discutir a aplicação dessa tecnologia em outras áreas do conhecimento e da computação, através da análise de trabalhos e contextualização com acontecimentos recentes.

Palavras-chave: *Machine Learning*; Técnicas; Aplicação; Computação; Contextualização.

8.1 Introdução

Com o advento da revolução técnico-científico-informacional, a quantidade de dados armazenados e em trânsito presentes na nova Era Digital cresceu exponencialmente - *Big Data* - tornando-se cada vez mais difícil a organização e o processamentos dessas informações de forma manual. Foram criadas, então, ferramentas e tecnologias capazes de tal feito, como é o caso da *Data Mining* e o *Machine Learning*.

O *Machine Learning* – ou "aprendizado de máquina" – surgiu como um ramo da Inteligência Artificial e está relacionado ao reconhecimento de padrões por meio de técnicas matemáticas e estatísticas, possibilitando a predição ou classificação de eventos reais. Em outras palavras, esse recurso é um método capaz de modificar seu comportamento de forma independente, tendo como base a sua própria experiência.

Sua aplicação está presente em diversas atividades cotidianas, por exemplo, ao pesquisarmos pelo significado da palavra "pena" em um site de buscas, o serviço pode optar por exibir resultados relacionados ao sentimento de piedade ou a uma condenação judicial. Para decidir por quais dos possíveis resultados pesquisar, o software de busca pode se utilizar do histórico de pesquisa do próprio usuário, fazendo uma predição do significado mais provável.

A situação contextualizada anteriormente é uma aplicação simples da capacidade do aprendizado de máquina, mas este também é utilizado em situações complexas e, inclusive, de alto risco, como administração de um banco de dados autônomo, combate à fraudes em sistemas de pagamentos, entre outras formas de automatização de processos.

Ademais, diversas outras áreas da ciência se beneficiam da aplicação do *Machine Learning*, catalisando suas pesquisas. Por exemplo, a Biologia Marinha e a Oceanografia realizam análises de fitoplânctons em amostras de águas marinhas, usando-as como indicadores da qualidade

da água do mar, utilizando o aprendizado de máquina para a contagem automática dos plânctons [2].

Outro exemplo do potencial da *Machine Learning* é sua contribuição na área da saúde. No Chile, por exemplo, foi desenvolvido um projeto de classificação automática de anamnese. Dessa forma, as técnicas de ML permitiram a identificação rápida de enfermidades relevantes na população, possibilitando a elaboração de políticas públicas adequadas [2].

8.2 Sistema básico de aprendizagem

Em qualquer processo de aprendizado de máquina, a fonte primária de informações ou atributos - *dataset* - também deve ser bem definida, podendo esta ser variável de acordo com o projeto em questão. As entradas de um algoritmo de ML se dividem em: numéricas, as quais são passíveis de aplicação de operações matemáticas, escalas racionais e intervalares, além disso, se dividem em contínuas (reais) ou discretas (inteiros); e categóricas, cujos conteúdos remetem a qualidades, podendo sofrer manipulação ordinal e nominal.

Os *datasets*, definidos previamente, podem ser implementados em diferentes tipos de aprendizado e suas respectivas técnicas, explicados a seguir [10]:

- *Aprendizado supervisionado*: caracterizado por realizar tarefas preditivas, usando funções, modelos ou hipóteses formados. Pode ser utilizados para fins de regressão ou classificação. Exemplos: Árvore de Decisão, Floresta Aleatória, KNN [3], entre outros.
- *Aprendizado não-supervisionado*: não há expectativa padrão de saída, apenas a representação dos dados. Além disso, é possível compactar os dados - sumarização - identificando atributos e similaridades desejadas. Dentro deste método, são utilizados ainda o agrupamento e a associação. Exemplos: *Clustering*, Detecção de Anomalias [1], Particionamento, Hierarquia e Densidade de Dados [7], etc.
- *Aprendizado por reforço*: o algoritmo descobre, por tentativa e erro, quais ações geram melhores resultados. Ele possui três componentes: o agente - tomador de decisões - ambiente e as ações. Exemplos: *Q-Learning*, Método de Aproximação [12] e outros.

Por fim, a escolha de cada componente do processo de ML, enunciado acima, resulta na estrutura básica de um sistema de aprendizado, formada pelos quatro campos a seguir [13]:

- *Ambiente*: fornece informações para a parte de aprendizado do sistema.
- *Aprendizado*: revisa a base de conhecimento utilizando as informações do ambiente.
- *Base de conhecimento*: pode ser um vetor de características, sentenças lógicas, regras de modelo de produção, entre outros.
- *Execução*: é o núcleo de todo o sistema, pois é a parte operativa, cujo foco é o aperfeiçoamento das ações de aprendizado.

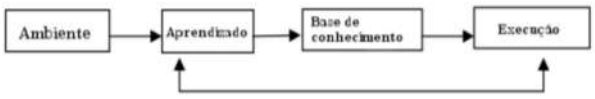


Figura 8.1: Estrutura básica de um sistema de aprendizado [13]

8.3 A iniciativa Open-Source e o Machine Learning

Na história da computação,o movimento *Open Source* - Código Aberto - foi uma forma de unir diversos profissionais, especializados ou não, para construir juntos ferramentas acessíveis e de propósitos gerais.

Com o *Machine Learning* não é diferente, um dos principais projetos de código aberto relacionado com ML, é o *TensorFlow*, uma biblioteca de funções e API's administrada pela *Google Brain Team*, criada em 2015, e que foi disponibilizada para que a comunidade corrigisse erros e desenvolvesse novas ferramentas, alcançando sua versão definitiva no ano de 2017. Dentre

os recursos presentes, a utilização da biblioteca possibilita o treinamento e execução de redes neurais profundas para classificação manuscrita de dígitos, reconhecimento de imagens, incorporação de palavras, modelos de tradução automática e outros. Hoje, possui suporte para todas as plataformas mais importantes, como *Linux*, *MacOS*, *Windows* e *Android*.

Conforme esta biblioteca foi sendo aprimorada, diversas empresas realizaram implementações que perduram até hoje. A Coca-Cola, por exemplo, se utilizou dessas ferramentas para realizar a criação de um algoritmo para auxiliar durante o escaneamento de códigos gravados em tampas e lacres dos produtos da empresa. Este algoritmo foi utilizado para predizer quais as letras e dígitos que estão presentes no produto através de uma foto tirada pelo próprio consumidor. Logo em seguida, a pessoa é redirecionada para o site de cadastro na promoção e, simultaneamente, o software já efetua o preenchimento dos códigos para que o usuário necessite apenas confirmar o cadastro.

8.4 O uso de Machine Learning para a segurança digital

Ao longo dos anos, o número de ocorrências relacionadas à questão de segurança só cresce. Hodiernamente, nos defrontamos com diversos tipos de ameaças de segurança promovidas por criminosos profissionais, que se utilizam desde ameaças escondidas em downloads até *bots* de rede. Tais ações mal intencionadas são empregadas por atividades ilegais, como roubo de informações sigilosas, disseminação de mensagens de spam, além da propagação e infecção por vírus ou softwares suspeitos.

Sucintamente, a segurança digital pode ser vista como um processo cíclico, o qual começa com a descoberta de novas ameaças. Pensando-se dessa forma, pode-se ilustrá-lo como na figura 2.

Entretanto, com o aumento na automatização dos ataques, esse ciclo está ficando desequilibrado. Das novas ameaças, apenas uma parte é detectada, e destas, somente algumas conseguem ser estudadas para a prevenção.

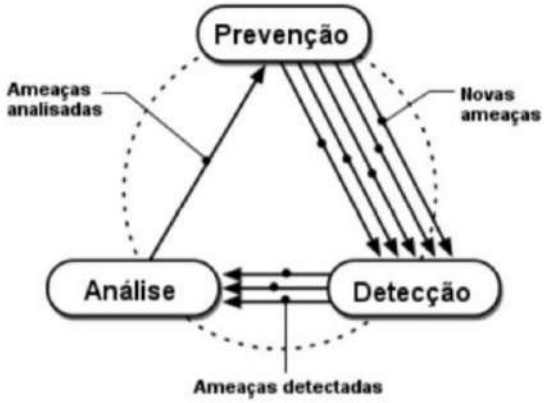


Figura 8.2: Processo cíclico da segurança de computador [13]

Deste modo, as técnicas de *Machine Learning* vêm sendo muito aplicadas também a essa área, visto que elas são capazes de analisar automaticamente os dados e providenciar decisões a tempo.

Exemplo prático: Detecção de intrusão em redes de computadores

Detecção de intrusão em redes são utilizadas para identificar atividades maliciosas sobre a confidencialidade, integridade ou violação da disponibilidade do sistema [8]. As técnicas implementadas para tal fim são comumente diferenciadas em dois tipos: assinatura e anomalia. A detecção por assinatura compara o tráfego com uma base de dados de ataques conhecidos, enquanto a detecção por anomalia compara os dados coletados com registros de atividades consideradas normais no sistema.

Ambas as abordagens possuem desvantagens, sendo que a primeira exige atualização frequente dos registros para garantir uma boa detecção, enquanto a última sofre de uma alta taxa de falsos positivos. Desta forma, o desafio é encontrar uma solução que resolva estes dois problemas, oferecendo tanto uma boa detecção, quanto uma baixa taxa de falsos alarmes.

E então, no trabalho [15], foi formulada uma proposta de detecção de ataques baseando-se em três níveis de classificação. No primeiro, a classificação é realizada por modelos gerados por um mesmo algoritmo base. No nível 2, um algoritmo de comitê é aplicado a vários modelos do classificador do nível 1. Finalmente, no terceiro nível, os resultados do anterior são combinados por um segundo algoritmo de comitê, gerando um agrupamento de modelos.

A utilização conjunta de algoritmos de ML e a base de banco de dados DARPA KDDCUP'99 simulou um ambiente fictício de uma rede militar, bombardeada por múltiplos ataques, tendo por objetivo analisar em quais casos e circunstâncias o aprendizado de máquina poderia ser vantajoso na identificação e classificação desses ataques, vantagem essa medida por meio de taxas de detecção e taxa de falsos positivos.

Por fim, os experimentos realizados demonstraram que esse modelo em três níveis apresenta melhores resultados do que a aplicação individual de algoritmos e a aplicação de apenas um nível de comitê. Quando comparado com outras propostas, o modelo mostrou-se superior em vários aspectos. No entanto, é importante notar que a aplicação de um terceiro nível de classificação exige uma maior quantidade de processamento, aumentando o tempo para realizar a classificação. Tal fato torna o modelo inviável para bases de dados muito grandes, mas adequado para sistemas que requerem alto nível de precisão na detecção ou que possuem uma quantidade média de dados a serem analisados.

Exemplo prático: Lavagem de dinheiro na rede Bitcoin

A criptomoeda *Bitcoin* consiste em um sistema monetário complexo, no qual mais de 130 bilhões de reais são movimentados diariamente, tornando-se maior do que o PIB anual de mais de 95 países em 2019, segundo a *World Bank*.

Em contrapartida, o *Bitcoin* também se tornou amplamente famoso por ser a moeda utilizada em atividades maliciosas e até criminosas, como sequestro de dados, lavagem de dinheiro e terrorismo digital. A realização de tais delitos são incentivadas pelo pseudo-anonimato fornecido pelo próprio *Bitcoin*, motivando ataques cibernéticos e mercados negros de armas e pirataria.

Especificamente sobre o processo de lavagem de dinheiro por meio do *Bitcoin*, o mesmo ocorre de maneira simples: os criminosos convertem o dinheiro ilícito em *Bitcoin*, tornando suas operações financeiras na rede anônimas, através de serviços de misturadoras – *BitCoin Mixers* – e, após certo tempo, comutam os *Bitcoin* de volta para dinheiro concreto, agora legal, justificado como supervalorização monetária.

Entretanto, assim como dito anteriormente, a rede *Bitcoin* não torna suas atividades totalmente anônimas, visto que as mesmas ficam permanentemente registradas. Além disso, o *Bitcoin* utiliza o modelo de saídas não-gastas (*Unspent Transactions Outputs – UTXO*), gerando vínculos entre todas as transações armazenadas no histórico da rede, o que permite a construção de um grafo direcionado para representar tal movimentação (figura 3). Dessa forma, mesmo que as misturadoras tentem omitir a origem ilegal dos recursos, é possível rastreá-las por meio de técnicas de aprendizado de máquina, assunto este abordado no trabalho [16].

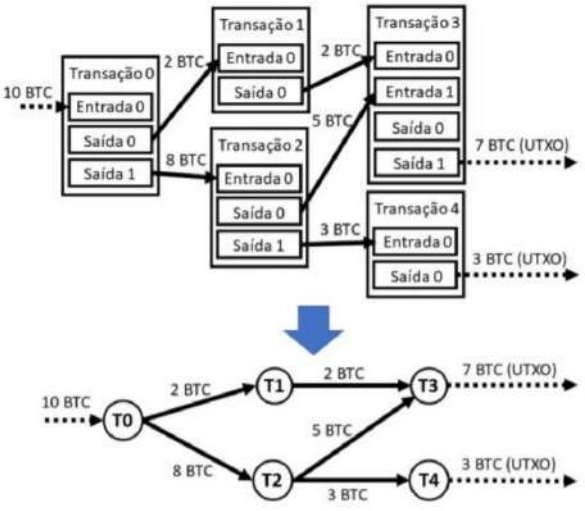


Figura 8.3: Grafo gerado por transações de criptomoeda *Bitcoin* [16]

A lavagem de dinheiro faz parte da classe de conjunto de dados desbalanceados, classe esta na qual o *dataset* é composto por amostras raras e sub representadas, dificultando sua identificação quando comparada a objetos comuns. O propósito do artigo foi verificar o desempenho, através de experimentos, da detecção de lavagem de dinheiro obtida por diferentes técnicas aplicadas em um conjunto de dados específico: o *Elliptic Dataset*.

Dentre os experimentos descritos no trabalho, o principal deles foi realizado para se comparar o desempenho de diferentes algoritmos de ML, como árvore de decisão, regressão logística, redes neurais probabilísticas e floresta aleatória; todos implementados na plataforma *KNIME*. Os resultados dos classificadores revelaram três diferentes níveis de desempenho: a regressão logística, com desempenho perto do aleatório, a árvore de decisão, com bom desempenho, e os algoritmos de redes neurais probabilísticas como o melhor resultado.

Além disso, um outro experimento efetuado mostrou uma melhora da classificação após aumentar-se o tamanho das partições do *dataset* destinadas ao treino, em relação àquelas reservadas para teste, sendo a proporção 90/10 a mais indicada.

Como recente e real exemplo de roubo envolvendo criptomoedas, a empresa *Binance*, uma das maiores no ramo de criptomoeda, teve mais de 7.000 *bitcoins* roubados em 2019, totalizando mais de US$ 40 milhões e impactando em cerca de 2% do estoque total da empresa. Através de violações de segurança, foi possível que os hackers obtivessem dados sigilosos de usuários cadastrados na empresa, como o código da autenticação de dois fatores, permitindo o acesso à conta do usuário e a manipulação seus dados bancários.

Um dos motivos apontados pela empresa como a causa do roubo foi a alta valorização do *Bitcoin* uma semana antes do ocorrido e que, por consequência da obtenção de dados dos usuários, seria possível que os hackers controlassem o mercado da criptomoeda, ou seja, influenciar o preço da mesma, tanto para valorizá-la quanto desvalorizá-la. [6]

Exemplo prático: Fraude em cartões de crédito

Os cartões, tanto de crédito quanto de débito, se tornaram importante e recorrente forma de pagamento no comércio mundial, seja ele presencial ou não. Com eles é possível efetuar pagamentos, realizar e aprovar transações, sacar dinheiro em espécie, entre outros; o que atrai criminosos fraudadores.

Segundo a Associação Brasileira das Empresas de Cartões de Crédito e Serviços (Abecs), as compras com cartão aumentaram 14,1% no primeiro trimestre de 2020, com um volume transacionado de R$ 475,7 bilhões, sendo 29% destas por meio do *e-commerce* [11]. Fora a imensa procura por este tipo de pagamento, a quantidade de novos tipos de fraudes também é crescente, compreendendo: fraude virtual, *phishing scam*, clonagem, *botnets* e *pharming* [9].

Especificamente em relação à técnica *phishing scam* - um método que consiste em criar uma página falsa, mas extremamente similar a original, para enganar o consumidor e roubar os dados pessoais inseridos pelo mesmo - casos como este são comuns no Brasil, como o exemplo da empresa *Nubank* em 2020. Desta vez, para convencer cada vez mais o consumidor de que se trata de uma página oficial e autorizada da *Nubank*, foi pedido não apenas dados pessoais, como CPF e senha, mas também uma foto do RG ou CNH da vítima, utilizando um sistema de detecção facial em conjunto ao documento (prática usual durante o cadastro oficial do cartão *Nubank*) [18].

Por se tratar de um sistema financeiro capaz de trazer tantas facilidades e recursos, é preciso ser altamente seguro. Para isso, a detecção, análise e solução de fraudes deve ser feita de maneira eficaz e simultânea, logo, é passível da aplicação de técnicas de aprendizado de máquina. Este foi o tema proposto pelo trabalho [4], no qual foram realizados experimentos avaliativos sobre a eficiência de algoritmos de ML na análise de

fraudes em cartões bancários.

Os algoritmos classificadores utilizados foram KNN, Floresta Aleatória, Regressão Logística, entre outros. Todos foram implementados através da linguagem *Python*, com o uso de bibliotecas específicas para ciência e mineração de dados, além de aprendizado de máquina, álgebra e randomização; sendo as principais delas *Numpy*, *Pandas* e *Scikit-Learn*. O *dataset* utilizado apresentava mais de 94 mil dados, com 16 campos explícitos, como valor da transação, horário e local da mesma, além de 13 campos ocultos, encriptados e protegidos por lei. O objetivo do uso de ambos os tipos de dados foi comprovar o potencial do ML na detecção das fraudes, mesmo as mais encobertas.

As medidas de avaliação utilizadas foram a precisão de casos positivos, taxa de falsos negativos, o *recall* - taxa de casos positivos em relação a todas as entradas positivas reais, incluindo os falsos negativos - e a média harmônica ponderada entre estas duas. Os resultados obtidos demonstraram que mediante proporção de teste e treino desigual, o algoritmo de Floresta Aleatória alcançou melhores resultados em relação ao KNN.

Entretanto, ao aumentar-se demasiadamente o tamanho do *dataset*, a Floresta Aleatória perdeu eficiência, visto que, com mais dados, o tempo necessário para particioná-los em subconjuntos se estende. Já com a proporção de dados próxima de 1/1, ambas as técnicas tiveram desempenhos melhores do que anteriormente, apresentando comportamento semelhante, ou seja, início ótimo e redução lenta e gradual conforme o crescimento do *dataset*, mas com a Floresta Aleatória ainda mantendo vantagem sobre o KNN

8.5 Considerações finais

O aprendizado de máquina é uma ferramenta efetiva e poderosa, que pode ser implementada em diversas áreas, além da computação. Por proporcionar tratamento real e constante às diversas situações, além de automatizar processos, análises e decisões, o ML está sendo tão estudado e utilizado ultimamente. Embora complexo e de alto custo, o uso desse recurso pode produzir excelentes resultados.

Dentre suas aplicações, como foi tratado mais ao fim deste artigo, está a segurança digital. Durante muitos anos, o enfoque dessa área residiu sobre a defesa de perímetros. Entretanto, um novo contexto delimitado pela agilidade dos atacantes e alto índice de proliferação de ameaças tornam essa abordagem ineficaz. Enfrentar esta situação exige um modelo de segurança mais moldável e dinâmico, baseado na gestão e resposta rápida aos incidentes de segurança.

Nesse sentido, o aprendizado de máquina é capaz de reduzir a quantidade de informações a serem estudadas por equipes de segurança e monitoramento, além de dividir e filtrar os alertas em subconjuntos. Tais subconjuntos preservam as principais informações necessárias para um profissional de segurança, construindo um mecanismo de redução de dimensionalidade que mantém o valor semântico dos dados. Ademais, tal abordagem permite não apenas detectar ataques conhecidos, como também indicar ataques que não faziam parte de sua massa de dados de aprendizado.

8.6 Bibliografia

[1] Aliger. Entenda o aprendizado não supervisionado no machine learning. [Acessado em: 09 de julho de 2020]. URL: `https://www.aliger.com.br/blog/machine-learning-entenda-o-que-e-aprendizado-nao-supervisionado`.

[2] Héctor Allende-Cid. Machine learning: Catalisador da ciência. *Revista Sociedade Brasileira de Computação nº 39*, page 15. [Acessado em: 02 de julho de 2020]. URL: `https://www.sbc.org.br/images/flippingbook/computacaobrasil/computa_39/pdf/CompBrasil_39_180.pdf`.

[3] Ethem Alpaydm. *Introduction to Machine Learning Ed. 2*. The MIT Press Cambridge, 2010. [Acessado em: 09 de julho de 2020].

URL: https://lamfo-unb.github.io/2017/07/27/tres-tipos-am/.

[4] BOURLA G. CHEN S. KASHYAP M. & PUROHIT S. BANERJEE, R. *Comparative Analysis of Machine Learning Algorithms through Credit Card Fraud Detection*. New Jersey's Governor's School of Engineering and Technology, 2018.

[5] Erhan. BUCZAK, Anna L; GUVEN. *A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection Vol. 18*. Number 2. IEEE COMMUNICATIONS SURVEYS & TUTORIALS, 2016.

[6] Olhar Digital. Hackers conseguiram roubar 40 milhões de dólares da binance. [Acessado em: 09 de julho de 2020]. URL: https://olhardigital.com.br/fique_seguro/noticia/hackersconseguiram-roubar-40-milhoes-de-dolares-em-bitcoin-da-binance/85564.

[7] Miriam Lúcia C. S. Domingues. *Mineração de Dados Utilizando Aprendizado Não-Supervisionado: um estudo de caso para bancos de dados da saúde*. UFRGS - Universidade Federal do Rio Grande do Sul, Instituto de Informática, Programa de Pós-graduação em Computação, 2003.

[8] Vitaly Ford and Ambareen Siraj. *Applications of Machine Learning in Cyber Security*. 27th International Conference on Computer Applications in Industry and Engineering, 2014.

[9] Manoel Fernando A. Gadi. *Uma comparação de métodos de classificação aplicados à detecção de fraude em cartões de crédito*. USP - Universidade de São Paulo, Instituto de Matemática e Estatística, 2008.

[10] Hugo Honda, Matheus Facure, and Peng Yaohao. Os três tipos de aprendizagem de máquina. [Acessado em: 09 de julho de 2020]. URL: https://lamfo-unb.github.io/2017/07/27/tres-tipos-am/.

[11] InfoMoney. Compras com cartão crescem 14,1% no 1º trimestre de 2020; ecommerce sobe 23%. [Acessado em: 09 de julho de 2020]. URL: https://www.infomoney.com.br/minhas-financas/compras-com-cartao-crescem-141-no-1o-trimestre-de-2020-e-commerce-sobe-23/#:~:text=S%C3%83O%20PAULO%20%E2%80%93%20As%20compras%20com.

[12] Eugênio P. F. D. Junior. *Aprendizado por reforço sobre o problema de revisitação de páginas Web, cap. 03*. Programa PPG em informática. Pontifícia Universidade Católica do Rio de Janeiro, 2012. [Acessado em: 09 de julho de 2020]. URL: http://www2.dbd.puc-rio.br/pergamum/tesesabertas/0912826_2012_cap_3.pdf.

[13] Cláudio Toshio Kawakani. *Segurança de computadores e Aprendizado de Máquina*. Universidade Estadual de Londrina, Londrina, 2014.

[14] Alessandro L. Koerich. Aprendizagem de máquina. *PUC-Paraná*. [Acessado em: 02 de julho de 2020]. URL: http://www.ppgia.pucpr.br/~alekoe/AM/2012/1-Introducao+EstudodeCaso-AM2012.pdf.

[15] Alex L. Ramos and Cícero dos Santos. *Combinando Algoritmos de Classificação para Detecção de Intrusão em Redes de Computadores*. Mestrado em Informática Aplicada – Universidade de Fortaleza (Unifor), Fortaleza - CE.

[16] Gabriel F. Rebello, Yining Hu, Kanchana Thilakarathna, Gustavo E. A. P. A. Batista, Aruna Senenviratine, and Otto Duarte. *Melhorando a Acurácia da Detecção de Lavagem de Dinheiro na Rede Bitcoin*. Universidade Federal do Rio de Janeiro, RJ, Brasil. University of New South Wales, Sydney. University of Sydney, Australia.

[17] Cynthia Rudin and Kiri L. Wagstaff. *Machine learning for science and society*. 2014.

[18] TecnoBlog. Nubank: anúncio no facebook levava a site falso e tentava roubar. [Acessado em: 09 de julho de 2020]. URL: https://tecnoblog.net/322518/nubank-anuncio-facebook-sitefalso-tentava-roubar-dados-phishing/.

[19] Lane Terran and E. Carla Brodley. *An Application of Machine Learning to Anomaly Detection*.

School of Electrical and Computer Engineering Purdue University, West Lafayette, 1997.

Parte II

Projetos

Capítulo 9

Ajuda Senhas

Maurício Cândido de Souza

9.1 Conceito

Projeto simples em C#, que utiliza a lista "RockYou" para auxiliar com a criação de senhas. Foi criado no intuito de ajudar usuários mais leigos, fazendo-os reconhecer (mesmo que um pouco) a importância de ter uma senha complexa.

9.2 Recursos utilizados

Além de terem sido utilizadas todas bibliotecas padrão, foi também utilizado o arquivo "rockyou", com o intuito de análise e leitura pela aplicação.

9.3 Tutorial

Apesar da maior parte do projeto ter sido realizada fora do código (com a análise das senhas), o passo a passo se caracterizou da seguinte forma:

1. Download da lista *RockYou.txt*.
2. Utilização de um script para adquirir informações e estatísticas sobre as senhas da lista.
3. Desenvolvimento do programa baseado em 5 estatísticas da etapa anterior.
4. Criação do "roteiro" de diálogos para a interação com o usuário.
5. Desenvolvimento do Design.
6. Adaptação ao Github.

- Devido ao limite de 25-100mb de arquivos, infelizmente o arquivo RockYou teve de ficar de fora, obrigando assim ao programa necessitar do arquivo à parte, apesar do mesmo estar disponibilizado aqui na página.

9.4 Instalação e Execução do Projeto

1. Baixe o projeto.
2. Baixe a lista Rockyou.
3. Após extrair ambos os arquivos, abra o *AjudaSenha.exe* dentro de /AjudaSenha/Bin/Debug.

9.5 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em: `https://github.com/yrmeww/Ajuda_Senhas`


Ajuda Senhas
? ? @ ?
@aaa
Testar!


Ajuda Senhas
? ? @ ?
lmao420
Muito cuidado! A senha que acabou de introduzir está em uma lista de senhas que já foram expostas!
OK

Capítulo 10

Among Us no Minecraft - Uma conversão de um jogo 2D para um ambiente 3D

Danilo Isamu Inafuku
Marcus Vinícius Natrielli Garcia
Fernando Favareto Abromovick
Michel Ribeiro Koba
Gustavo de Jesus Rodrigues Silva
Vinícius Venturini
Caio César Brandini da Silva
Mauricio Cândido de Souza
Felipe Bertoni Salvati

10.1 Conceito do projeto

Este projeto trata-se de uma conversão das mecânicas e sistemas do popular jogo de dedução social *Among Us* para o ambiente do jogo sandbox *Minecraft*. Para isto, muitas técnicas e implementações comuns à linguagens de programação tiveram de ser abstraídas, simuladas e traduzidas para o sistema nativo de "blocos de comando"que o jogo possui.

Para um resumo das regras, sistemas e mecânicas de *Among Us*, vide *Guia de Among Us* [1] para um guia em português e *A Long Guide Among Us* [4] para uma explicação mais completa (porém em inglês).

Para uma explicação sobre o jogo *Minecraft*, é recomendado visitar o site *Minecraft Wiki* [6].

10.2 Pré-requisitos e recursos utilizados

O grupo utilizou o videogame *Minecraft* [9] para o desenvolvimento principal, com o jogo *Among Us* [8] e o site *Minecraft Wiki* [5] como principais recursos de consulta.

Outros recursos utilizados foram:

- *WebFx* [10] para a obtenção e conversão de cores no jogo;
- *Gamewith* [3] para um melhor detalhamento do sistema de Tarefas do jogo *Among Us*;
- *PlanetMinecraft* [7] para mais informações sobre o sistema de *Bossbars* de *Minecraft*.

10.3 Passo a passo

- O servidor-ambiente de jogo foi criado
- O mapa do jogo foi examinado, simulado e construído no ambiente de jogo
- O *setup* inicial ("código"executado para inicializar o jogo) foi iniciado, assim como o ambiente de desenvolvimento
- O sistema de *cooldown* foi implementado pela primeira vez
- O sistema de votação foi implementado
- O sistema de escotilhas foi implementado e construído

- O sistema de morte foi implementado
- O sistema de configurações do jogo (*setup* antes do jogo) foi implementado
- O sistema de condições de vitória foi implementado
- Os quatro sistemas de Sabotagem (Nuclear, Elétrica, O2 e Comms) foram implementados
- O gerenciador de Tarefas, criado para distribuir e controlar o sistema de Tarefas no início e decorrer do jogo, foi implementado
- Os treze sistemas de Tarefa (Admin, Divert, Upload, Medscan, Manifolds, Sample, Shields, Align Output, Steering, Fuel, Garbage, Asteroids e Wiring) foram implementados
- O sistema de Escuta, criado para substituir o sistema de Câmeras no jogo original, foi criado e implementado
- O quinto sistema de Sabotagem (Portas) foi implementado
- O sistema de Compasso, criado para ajudar a orientar os jogadores pelo mapa, foi implementado
- O sistema de cosméticos, criado para maior customização dos jogadores por meio de "máscaras", foi implementado

Tais passos foram intercalados por diversos *bugfixes* e testes coletivos. Destaque ao sistema de votação, que de longe foi o sistema com mais problemas durante o desenvolvimento do projeto.

10.4 Instalação

Acesse o servidor oficial do *Hackoonspace* de *Minecraft* (versão atual - 1.16.3), e siga para o lobby de *Among Us* sinalizado na área inicial.

10.5 Execução

1. Reúna 5-12 jogadores no *lobby* do jogo.
2. Pressione o botão de configurações para modificar certas regras do jogo.
3. Pressione o botão no centro da sala para iniciar o jogo.

A maioria das regras segue idêntica ao jogo original. Todas as diferenças estão listadas sob o botão *Regras*.

10.6 Implementação

A implementação foi feita principalmente pelo uso de "blocos de comando", que podem ser comparados a linhas de código em linguagens de programação convencionais. Foi feito também uso de *redstone*, o sistema elétrico do jogo, para a implementação de portas lógicas, certos *loops* e, de certo modo, funções.

Figura 10.1: *Redstone* utilizada para formar a porta lógica *AND*

Comandos principais

Dezenas de tipos de comandos foram utilizados, mas alguns merecem destaque por sua utilidade e similaridade a estruturas de programação:

- *Execute*: similar a um *if*; também usável para a atribuição de certas variáveis.
- *Scoreboard*: o sistema de *scoreboard* foi essencial para o jogo, sendo responsável quase inteiramente por gerenciar as variáveis e as constantes do jogo, assim como certas formas de *input* e operações lógicas envolvendo elas.
- *Tellraw*: com o uso deste comando, foi possível utilizar o formato JSON do jogo para formar mensagens personalizadas e imprimi-las para jogadores específicos na área de *chat* do

jogo. Em essência, serviu como tanto o *printf* quanto o *scanf* do jogo.

Estruturas de blocos de comando

No decorrer do projeto, várias técnicas foram descobertas para facilitar a implementação de certas funcionalidades do projeto por meio de diferentes arranjos de combinações de blocos com comandos.

A função

A estrutura mais básica e mais essencial do projeto foi, sem dúvida, a função. Apesar de *Minecraft* oferecer seu próprio sistema de funções (com o commando */function*), este necessita a instalação de *datapacks*. Por isso, um sistema mais rústico foi utilizado, formado por um comando "chamador"da função, que posiciona um bloco ativador na frente dos blocos de comando, e os blocos de comando em si, que formam a função.

Figura 10.2: Comando chamador da função

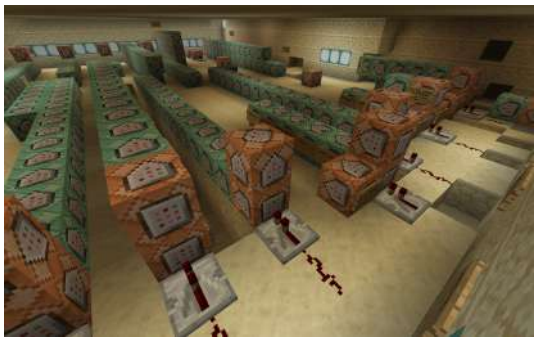


Figura 10.3: Função

O *do/while*

Para criar a mecânica de de votação, *loops* tiveram de ser criados para se adaptar à quantidade variável de jogadores no jogo. Assim, um sistema semelhante à um *loop do/while* foi criado, formado pela função principal, um bloco-condição, que checa uma condição usando execute e chama blocos de comandos ajudantes que invocam a função principal novamente após uma curta espera, e um bloco-quebra, que é chamado ao fim do *loop*.

Figura 10.4: Estrutura marcada com placas para sinalização

O checador

Minecraft fornece uma opção de blocos de comando contínuos, que são invocados a todo *tick*. *Minecraft* também oferece a opção de blocos condicionais, que são chamados apenas quando o anterior for bem-sucedido. Apesar de muito úteis, estes, quando pareados, às vezes não são suficientes, pois em uma situação de *switch/case*, ou todos os comandos são executados a todo *tick* (assim gastando processamento), ou nenhum é chamado. Para evitar este problema, a estrutura *checador* foi criada, que troca os blocos condicionais por um único bloco contínuo com o comando *execute*, que, quando bem-sucedido, chama uma função com o resto dos comandos.

A cadeia de conjuntos de comando

Para o sistema de distribuição de tarefas, foi necessário sortear um jogador aleatório para a obtenção de cada Tarefa do jogo, repetindo a função até que todos os jogadores tiverem a quanti-

Figura 10.5: O checador, sinalizado com placas

dade necessária de Tarefas. A obtenção de tarefas para um jogador, porém, necessita que diversos comandos sejam executados em um mesmo jogador sorteado aleatoriamente. Para criar isto, a cadeia de comandos foi criada. A cada "nódulo"da cadeia, um jogador aleatório recebe uma *tag Target* para receber a Tarefa do nódulo seguinte, enquanto que o jogador com a *tag* do nódulo anterior recebe a sua própria tarefa. Assim, os comandos rodam de forma eficiente e facilmente expansível - para adicionar uma nova tarefa ao jogo, é necessário apenas criar um novo nódulo.

Figura 10.6: A cadeia de conjuntos de comando

10.7 *Bugs*/problemas conhecidos

Devido à diferença intrínseca de ambos ambientes e às limitações do projeto, algumas mecânicas do jogo original tiveram de ser retiradas ou pesadamente modificadas para se adaptar ao ambiente 3D. Estas incluem:

- Fantasmas (jogadores mortos) não podem realizar Tarefas. Ao invés disso, Tarefas de fantasmas são completas a cada 20 segundos, a partir do momento de suas mortes. Esta funcionalidade foi retirada devido aos diversos problemas causados pela presença e influência física dos jogadores mortos no jogo.

- O sistema de câmeras foi substituído por um sistema de escuta que transmite *pings* de jogadores em certos pontos do mapa para a sala de segurança. A substituição foi feita com relutância, pois o sistema de câmeras seria tecnicamente possível: apesar de *Minecraft* não suportar a criação de câmeras em tempo-real, um sistema alternativo utilizando invisibilidade, teletransporte e o uso de uma criatura na posição original do jogador [2] possibilitaria a implementação de tal mecânica. Decidimos, porém, que tal funcionalidade seria esdrúxula demais para o design da simulação, e retiramos o sistema.

10.8 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/MasterProjectLC/among_us_minecraft`

10.9 Bibliografia

[1] Among us: Guia completo (pt-br). URL: `https://steamcommunity.com/sharedfiles/filedetails/?id=1908555816`.

[2] Chau, hamish. among us (1.16.3). URL: `http://phoenixsc.me/download-links/among-us-1-16-3/`.

[3] Gamewith. tasks list - common & visual tasks list. URL: `https://gamewith.net/among-us-wiki/article/show/22106`.

[4] A long guide among us. URL: `https://steamcommunity.com/sharedfiles/filedetails/?id=2220357997`.

[5] Minecraft wiki. URL: https://minecraft.gamepedia.com/Minecraft_Wiki.

[6] Minecraft wiki pt-br. URL: https://minecraft-pt.gamepedia.com/Minecraft.

[7] Shelluser. setting up a bossbar in minecraft 1.13. URL: https://www.planetminecraft.com/data-pack/setting-up-a-bossbar-in-minecraft-1-13/.

[8] Site oficial de among us. URL: http://www.innersloth.com/gameAmongUs.php.

[9] Site oficial de minecraft. URL: https://www.minecraft.net/pt-pt.

[10] Webfx. hex to rgb. URL: https://www.webfx.com/web-design/hex-to-rgb/.

Capítulo 11

Ferramenta Beans

Bruno Sacconi Peres

11.1 Conceito do projeto

Este projeto se trata de um utilitário de interface gráfica para ajudar no uso dos softwares *i3* [3], *Polybar* [4] e, com sorte, muitos outros programas relacionados a *ricing*[1] que dependem de arquivos de texto para personalização e configuração.

Atualmente, *Beans* está sendo desenvolvido para funcionar no *Arch Linux*, mas ele também pode ser executado em outras distros do *GNU/Linux*.

11.2 Observações importantes

Este projeto ainda está em uma fase bem primitiva, e eu considero que ele é apenas uma prova de conceito, por enquanto. Atualmente, o *Beans* possui apenas uma interface gráfica de configuração em desenvolvimento para *Polybar* [4].

11.3 Passo a passo

- O *Beans* foi criado do zero, usando um *script* em *Perl* simples para analisar os arquivos de configuração existentes. Sua interação era feita por uma interface CLI básica.

- Então, comecei a melhorar a interface, além da funcionalidade, criando menus para o usuário escolher os valores para as opções fornecidas pelo wiki de configuração do *Polybar* [6]. Comecei usando *Xdialog* [7] como interface principal, mas problemas de formatação de texto começaram a aparecer.

- Decidi migrar para o terminal *Dialog* [2] baseado em *ncurses*. Nesse ponto, tudo estava bem no que dizia respeito à interface. Mas o projeto estava ficando cada vez mais difícil de manter, pois sua base de código crescia cada vez mais rápido. Em um certo ponto, haviam mais de 1500 linhas de código em um único *script Perl* que nem sequer estava na metade (me refiro apenas ao parser de configuração do *Polybar*). Estava uma bagunça, e algo tinha que mudar.

- Dividi as funcionalidades do *Beans* em módulos. Ele possuía um *back-end* que servia par analisar os arquivos de configuração e gerava arquivos JSON com base neles. Logo desisti dessa ideia.

- Comecei o desenvolvimento da interface gráfica com o *GTK* [5] usando *C++*. Como eu não tinha certeza de como integrar meu parser *Perl* com meu código C++, decidi portar meu parser para *C++* e abandonei meus *scripts Perl* para sempre. Agora, tudo parece estar bem integrado e eu espero estar no caminho certo.

- Consegui tornar a configuração de cores do *Polybar* perfeitamente funcional!

[1] *Rice/ricing* - prática de customizar ou aprimorar o visual de uma área de trabalho, principalmente por meios não convencionais ou padrões [1]

- A interface gráfica com *GTK* sofreu mudanças severas em relação ao design original e agora parece muito melhor.

11.4 Dependências

- gtkmm 3.24.2
- meson >= 0.54.3
- ninja >= 1.10.0
- gcc >= 10.1.0
- GNU make >= 4.3

Demais dependências serão adicionadas no repositório do GitHub.

11.5 Problemas Conhecidos

Atualmente, eu ainda estou aprendendo a usar o *GTK*. Dessa forma o desenvolvimento da interface gráfica será lento, mas pelo menos já começou!

Como o projeto ainda está em andamento, o programa ainda não pode ser usado completamente.

11.6 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/mdk97/Beans`

É importante notar que a versão mais recente do código está na *branch dev* do repositório.

11.7 Imagens

Figura 11.1: Interface de escolha de software

Figura 11.2: Interface de escolha de seção do *Polybar*

Figura 11.4: Configuração de nova cor

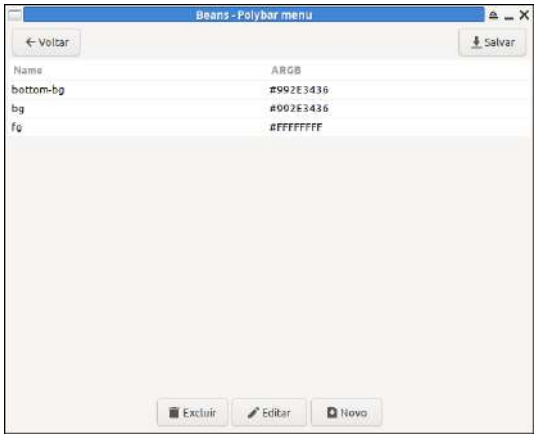


Figura 11.3: Lista de cores configuradas

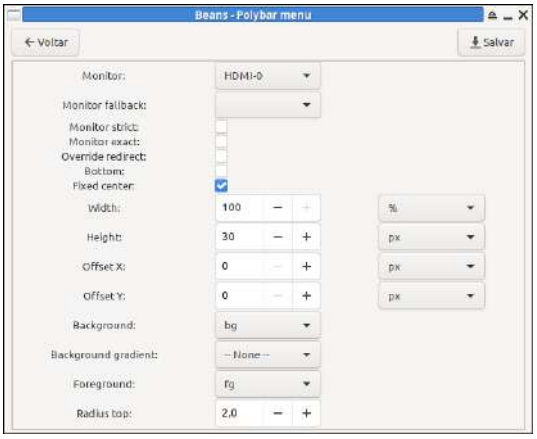


Figura 11.6: Configuração de nova interface

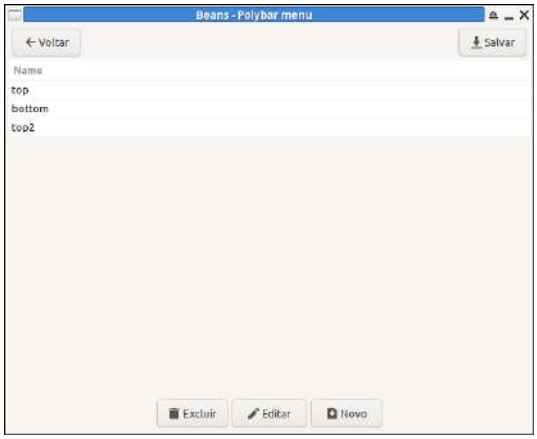


Figura 11.5: Lista de interfaces criadas

11.8 Bibliografia

[1] Definição de rice/ricing. URL: https://www.reddit.com/r/unixporn/wiki/themeing/dictionary#wiki_rice.

[2] Dialog. URL: https://invisible-island.net/dialog/.

[3] i3. URL: https://i3wm.org/.

[4] Polybar. URL: https://github.com/polybar/polybar.

[5] Site oficial do gtk. URL: https://www.gtk.org/.

[6] Wiki de configuração do polybar. URL: https://github.com/polybar/polybar/wiki/Configuration.

[7] Xdialog. URL: http://xdialog.free.fr/.

Capítulo 12

Bravely Default Index Injector

Vítor Ribeiro Guimarães Gomes

12.1 Conceito

A aplicação permite criar um arquivo binário Mestre chamado Crowd.fs e um arquivo Index.fs para o jogo Bravely Default de Nintendo 3DS.

O arquivo *Crowd.fs* possui estrutura lógica similar a um arquivo em .zip, ou seja, este formato é uma compactação proprietária de arquivos menores feita pela empresa *Square Enix* para este jogo em específico, os sub-arquivos devem ser indicados pelo usuário em um diretório escolhido para que sejam compactados.

Além disso, o projeto também permite editar o arquivo *Index.fs* feito pelos desenvolvedores, este arquivo contêm informações sensíveis ao *Crowd.fs*, tais como o tamanho dos arquivos compactados e a posição de onde se iniciam dentro do *Crowd.fs*, dessa forma, esta função pode modificar estas duas sequências de bytes para que seja possível carregar os arquivos traduzidos de forma correta dentro do jogo.

12.2 Pré-requisitos e recursos utilizados

O Programa feito em C# usando as Funções nativas da linguagem conhecidas como *BinaryReader/BinaryWriter* para o jogo *Bravely Default* de Nintendo 3DS. Além disso, foi utilizado a Biblioteca *System.Threading* para inserir comandos de *Delay* no console de *Debug*, ficando mais simples de verificar as operações feitas e as bibliotecas padrão do *Windows Forms*, permitindo fazer uma interface para o projeto.

Para *Debug* do programa, foi criado a exibição de um Console para que o usuário possa verificar o que está ocorrendo com o programa em tempo real, sendo mais simples de localizar erros em arquivos, caso haja.

12.3 Passo a passo

- Extraí a *RomFS* do jogo *Bravely Default* e descompactei os arquivos *Crowd.fs* e *Index.fs* originais para que fosse gerado os sub-arquivos do projeto
- Estudei a estrutura de ponteiros e textos dos arquivos do jogo para realizar a engenharia reversa nos arquivos Mestre
- Implementei uma função com o objetivo de compactar os sub-arquivos e gerar um arquivo chamado *Crowd.fs*, utilizando da estrutura estudada
- Implementei uma função com o objetivo de gerar o arquivo *Index.fs* com as sequências de bytes modificadas referentes ao tamanho dos arquivos traduzidos e da posição de onde se iniciam na compactação do novo *Crowd.fs*
- Criei uma interface gráfica para utilizar o programa de comunicação de forma mais intuitiva, adicionando um menu de Ajuda com tutoriais para utilizar o programa e abas de

referência ao repositório do código e meu perfil no Github.

12.4 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/MrVtR/Bravely_Default_Index_Injector`

12.5 Imagens


```
private void BtnIndex_Click(object sender, EventArgs e)
{
    try
    {
        folderBrowserDialog1.Description = "Selecione o diretório com o Crowd.fs, index.fs e arquivos antigos extraídos: ";
        DialogResult dr = folderBrowserDialog1.ShowDialog();
        string path = folderBrowserDialog1.SelectedPath.ToString();

        string crowdFsAntigoCheck = path + @"\antigo\Crowd\crowd.fs";
        string crowdFsNovoCheck = path + @"\novo\Crowd\crowd.fs";
        string indexFsAntigoCheck = path + @"\antigo\Crowd\index.fs";
        string indexFsNovo = path + @"\novo\Crowd\index.fs";

        string[] dirAntigo = Directory.GetFiles(path + @"\antigo");//args[1]
        string[] dirNovo = Directory.GetFiles(path + @"\novo");//args[2]

        if (File.Exists(crowdFsAntigoCheck) && File.Exists(crowdFsNovoCheck) && File.Exists(indexFsAntigoCheck) && dirAntigo.Length > 0 && dirNovo.Length > 0)
        {
            string[] vetTamanhoAntigo = new string[dirAntigo.Length];
            string[] vetTamanhoNovo = new string[dirNovo.Length];
            string[] vetCrowdAntigo = new string[dirAntigo.Length];
            string[] vetCrowdNovo = new string[dirNovo.Length];
            string fileName = indexFsNovo;//args[3]
            byte[] byteIndex = ReadAllBytes(indexFsAntigoCheck);//args[4]
            byte[] byteCrowdAntigo = ReadAllBytes(crowdFsAntigoCheck);//args[5]
            byte[] byteCrowdNovo = ReadAllBytes(crowdFsNovoCheck);//args[6]

            criaArquivoCopiarVetor("index.fs", fileName, byteIndex);
            setVetor(dirAntigo, byteCrowdAntigo, vetTamanhoAntigo, vetCrowdAntigo);
            Console.WriteLine("Tamanhos antigos alocados em vetor: Feito");
            Console.WriteLine("Criado vetor do crowd.fs antigo");
            Console.WriteLine("-----------------------------------------------");
            setVetor(dirNovo, byteCrowdNovo, vetTamanhoNovo, vetCrowdNovo);
            Console.WriteLine("Tamanhos novos alocados em vetor: Feito");
            Console.WriteLine("Criado vetor do crowd.fs antigo");
            Console.WriteLine("-----------------------------------------------");
            Thread.Sleep(200);
            imprimeIndex(byteIndex, "Original");
            Thread.Sleep(200);
            Console.WriteLine("-----------------------------------------------");
            reinsereByte(byteIndex, vetTamanhoAntigo, vetTamanhoNovo, fileName);
            Console.WriteLine("-----------------------------------------------");
            reloceaByte(byteIndex, vetCrowdAntigo, vetCrowdNovo, fileName);
            Console.WriteLine("-----------------------------------------------");
            Thread.Sleep(200);
            byteIndex = ReadAllBytes(indexFsNovo);//args[4]
            imprimeIndex(byteIndex, "Modificado");
            Console.WriteLine("\nPrograma finalizado");
            MessageBox.Show("Index.fs criado com sucesso", "Concluído");
        }
    }
```

```
1 referência
private void BtnCrowd_Click(object sender, EventArgs e)
{
    try
    {
        folderBrowserDialog1.Description = "Selecione o diretório com o Crowd.fs, index.fs e arquivos antigos extraídos: ";
        DialogResult dr = folderBrowserDialog1.ShowDialog();
        string path = folderBrowserDialog1.SelectedPath.ToString();
        string crowdFsCheck = path + @"\antigo\Crowd\crowd.fs";
        string indexFsCheck = path + @"\antigo\Crowd\index.fs";
        string[] dir = Directory.GetFiles(path + @"\novo");

        if (File.Exists(crowdFsCheck) && File.Exists(indexFsCheck) && dir.Length>0)
        {
            DirectoryInfo di = Directory.CreateDirectory(path + @"\novo\Crowd");
            //Cria crowd.fs novo com Append dos arquivos novos
            string fileName = path + @"\novo\Crowd\crowd.fs";
            FileStream writeStream = new FileStream(fileName, FileMode.Create);//Criação do arquivo

            for (int j = 0; j < dir.Length; j++)
            {
                byte[] lerArquivo = File.ReadAllBytes(dir[j]);//Pega cada arquivo de dirs
                for (int k = 0; k < lerArquivo.Length; k++)//pega cada byte do arquivo coletado
                    writeStream.WriteByte(lerArquivo[k]); //Coloca os dados dos arquivos novos em um Crowd.fs
                Thread.Sleep(20);
                Console.WriteLine("Inseriu: " + Path.GetFileName(dir[j]));
            }
            Console.WriteLine("\nPrograma finalizado");
            MessageBox.Show("Crowd.fs criado com sucesso","Concluído");
            writeStream.Close();
        }
        else
            MessageBox.Show("Arquivos necessários não encontrados","ERRO");
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message,"ERRO");
    }
}
```

Capítulo 13

Bravely Default Text Injector

Vítor Ribeiro Guimarães Gomes

13.1 Conceito

A aplicação permite injetar os textos modificados e filtrar os ponteiros dos arquivos do jogo Bravely Default.

O programa feito permite criar um arquivo binário em codificação UTF-16 com os textos fornecidos pelo usuário (tendo o texto \n em seu formato ou não) para serem inseridos no jogo *Bravely Default* do console Nintendo 3ds, dessa forma, permitindo que a tradução do jogo de Inglês para Português Brasileiro seja feita de forma automatizada durante a fase de *Romhacking* do jogo. Também está incluso uma função de filtragem de ponteiros corrompidos (ponteiros que são desnecessários para o jogo rodar normalmente), dessa forma, é possível otimizar o processo de tradução com o aplicativo *Kruptar7*.

13.2 Pré-requisitos e recursos utilizados

Programa feito em C# com a Biblioteca *Komponent* (para fins de uso do comando *BinaryWriter* customizado) do programa *Kuriimu2* [2].

Além disso, foi utilizado as bibliotecas padrões do *Windows Forms*, permitindo fazer uma interface para o projeto. Para *Debug* do programa, foi criado a exibição de um *Console* para que o usuário possa verificar o que está ocorrendo com o programa em tempo real, sendo mais simples de localizar erros em arquivos, caso haja.

13.3 Passo a passo

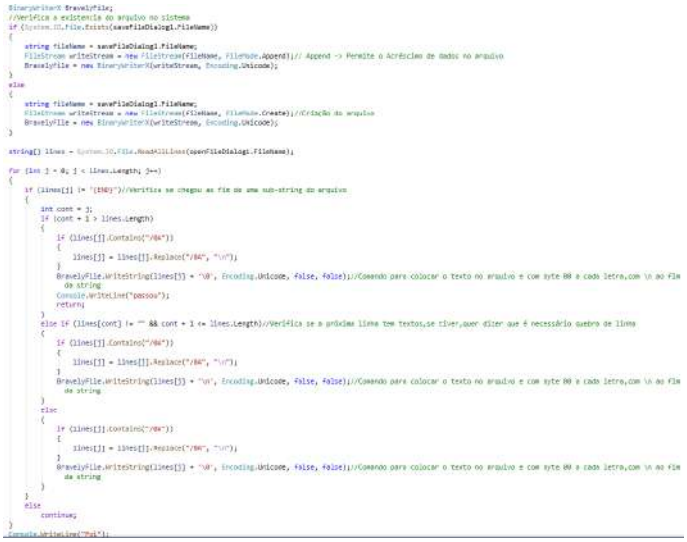
- Extraí a *RomFS* do jogo *Bravely Default* e descompactei os arquivos *Crowd.fs* e *Index.fs* originais para que fosse gerado os sub-arquivos do projeto
- Estudei a estrutura de ponteiros e textos dos arquivos do jogo para realizar a extração usando o aplicativo russo *Kruptar7* [1]
- Criei projetos em extensões .kpx dentro do *Kruptar7* a partir da engenharia reversa dos ponteiros do jogo para poder extrair e injetar os textos com arquivos .txt
- Implementei uma função com o objetivo de injetar os textos traduzidos, convertendo do formato UTF-8 para UNICODE (Codificação de texto aceito pelos arquivos do *Bravely Default*)
- Implementei uma função com o objetivo de filtrar os ponteiros corrompidos dos arquivos de texto do jogo, dessa forma, o processo de tradução é otimizado, juntamente com o carregamento dos futuros *patchs*, visto que os arquivos serão mais leves utilizando o filtro.
- Criei uma interface gráfica para utilizar o programa de comunicação de forma mais intuitiva, adicionando um menu de Ajuda com tutoriais para utilizar o programa e abas de referência ao repositório do código e meu perfil no Github.

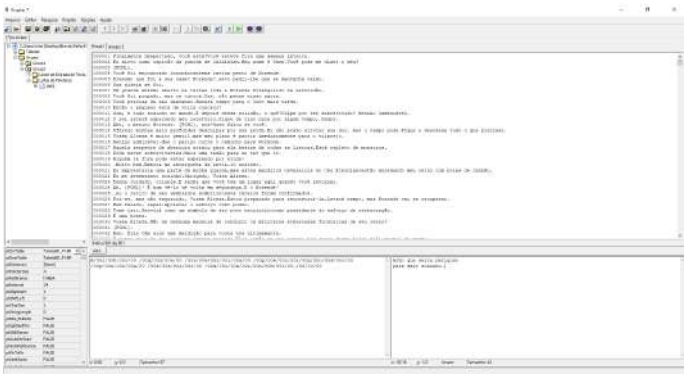
13.4 Repositório

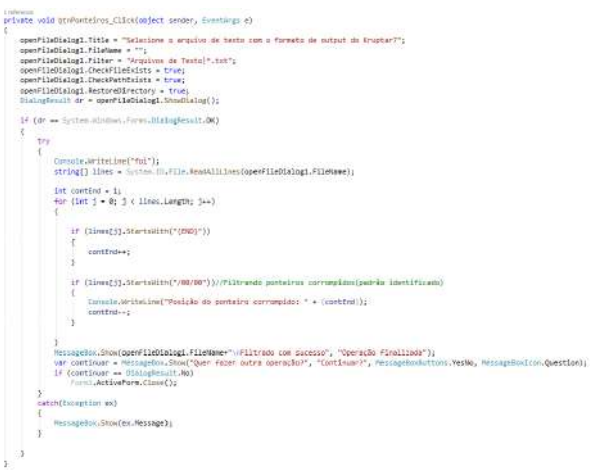
O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/MrVtR/Bravely_Default_Text_Injector`

13.5 Imagens







13.6 Bibliografia

[1] Kruptar7. URL: `http://magicteam.net/?page=programs&show=Kruptar%207`.

[2] Source do kuriimu2 com a biblioteca usada. URL: `https://github.com/FanTranslatorsInternational/Kuriimu2`.

Capítulo 14

Cheat The Gungeon

Guilherme Henrique Rodrigues
Rafael Jyo Kondo

14.1 Conceito

O projeto consiste de uma interface que permite a usar "cheats" (truques especiais durante o jogo) em um jogo chamado Enter the Gungeon. Para isso, foi necessário implementar a interface, e o cheat do jogo.

14.2 Pré-Requisitos e Recursos Utilizados

O grupo utilizou a linguagem C# para desenvolver a implementação geral do projeto, além de importar as seguintes bibliotecas: Memory.dll.x64 e Memory.dll.x86 ambas desenvolvidas por NeWaGe, hollow87. Para ajudar a entender melhor sobre o assunto, foram pesquisadas várias informações sobre Como CHEATS e HACKS de jogos são criados? para compreender a implementação do cheat.

14.3 Passo a Passo

- Foi baixado o material disponível em Cheat Engine;
- Usando a ferramenta Cheat Engine, os endereços dos dados do jogo foram procurados;
- Então, foi implementado um programa em C# que detecta cada endereço dos dados e modifica-os;
- Foi implementada uma interface de usuário para facilitar o uso do programa.

14.4 Instalação

- Tenha instalado o jogo Enter The Gungeon em sua máquina;
- Baixe os arquivos executáveis no Release, disponível no repositório e extraia.

14.5 Execução

- Abra o Cheat The Gungeon como Administrador;
- Abra o jogo Enter The Gungeon;
- Escolha o cheat que deseja utilizar.

14.6 Bugs Conhecidos

Existe a possibilidade de falha na alteração da memória. O cheat utiliza de uma base de endereço somado a offssets, logo, existe a possibilidade que o resultado dessa soma chegue há um endereço nao relacionado ao jogo, possibilitando bugs no cheat. O cheat foi desenvolvido na versão 1.1.3 do jogo, portanto só funcionará nesta versão. Em qualquer outra versão o cheat não irá funcionar.

Exemplo: O cheat usa como base a memória que está sendo executada durante o jogo, ou seja,

para alterar a quantidade de dinheiro, é injetado um valor diretamente na memoria, durante a execução do jogo. Se o endereço não for o correto, a quantidade de dinheiro não será alterada. O mesmo acontece com a vida e os blanks.

14.7 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/nyanham/Cheat-the-Gungeon`

14.8 Imagens

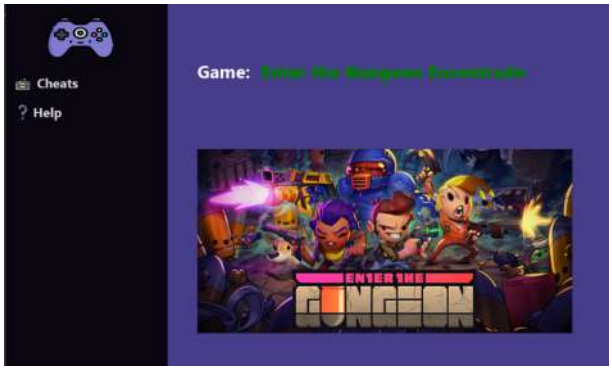
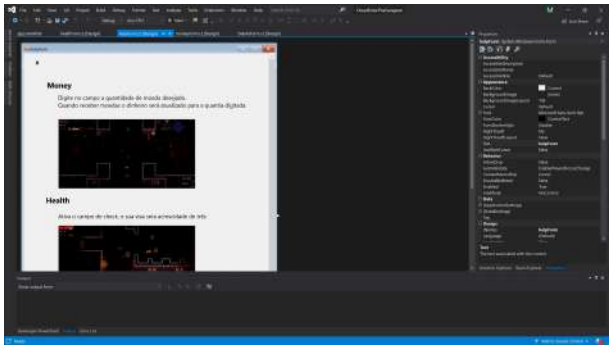


Figura 14.1: Tela inicial, aonde será exibido se o processo do jogo foi encontrado




```
if (moneyTextBox.Text != " ")
    m.WriteMemory("base+00FAF7B0,c,1c,38,40,18,f8,14", "int", moneyTextBox.Text);
```

Figura 14.2: Uso das bibliotecas, utilizando do método write para alterar a memória em execução.

```
procOpen = m.OpenProcess("EtG");
```

Figura 14.3: Uso das bibliotecas para encontrar o jogo em execução

Capítulo 15

R0 da COVID-19 na cidade de Sorocaba

Guilherme Milani de Oliveira
Jean Wylmer Flores
Gabriel Kyomen

15.1 Conceito

O projeto visa calcular e projetar o Número de Reprodução Básico (R0) da COVID-19 em Sorocaba-SP, um dado que, por muitas vezes, é deixado de lado por orgãos públicos, porém tem sua eficácia comprovada em analisar o contágio da doença e mensurar os esforços no controle da pandemia, visto que nos mostra quantas pessoas serão infectadas a partir de um único indivíduo portador do patógeno.

15.2 Pré-requisitos e Recursos Utilizados

Foi utilizada a linguagem PHP na implementação geral do projeto, com uso também de HTML, CSS, JavaScript e JSON, além de testar o site em servidor Apache local por meio do pacote XAMPP. Foi importada a biblioteca Charts.js em JavaScript para montagem de gráfico e os seguintes tutoriais como base para trechos do código:

- Efficiently counting the number of lines of a text file
- How to rewind() an http stream file in PHP
- php.net/manual/strtok

15.3 Passo a Passo

- Foi feito um processo de pesquisa sobre o cálculo do R0 e busca de fontes de estudo epidemiológicas confiáveis. (Caso deseje mais informações sobre esse passo entre em contato com os autores);
- Foi criado um protótipo em C que lia um arquivo .csv providenciado pela Prefeitura e calculava o R0;
- Protótipo foi transformado em código PHP e um formato base em HTML (index.php) também foi criado;
- As funções que tratavam explicitamente do R0 foram colocadas numa API (api.php);
- Guardar os valores previamente calculados num arquivo JSON (`r0_values.json`);
- Estilização do site criada (style.css);
- Foi adicionado um gráfico alimentado pelo json criado, que depois foi ajustado para se tornar responsivo;
- Um domínio foi comprado e hospedado para divulgação do site com a comunidade. Pode ser encontrado em `http://covidsorocaba.online` até Julho de 2021.

15.4 Instalação e Execução

- Baixe a pasta com os arquivos e instale XAMPP na sua máquina;

- Execute, no terminal, o comando: `php -S localhost:4000`

- Abra seu navegador em "localhost:4000/(caminho até o arquivo index.php)".

15.5 Bugs Conhecidos

Alguns problemas envolvendo o armazenamento dos dados no arquivo `r0_values.json`, como a atribuição de valores nulos quando o arquivo é manipulado de forma específica e erros em atualizar automaticamente o arquivo. Possíveis causas não são claras mas há chances do problema estar na função `store_r0` dentro do arquivo api.php.

15.6 Demais Anotações e Referências

O que é R0?

O Número Básico de Reprodução, mais conhecido como R0 (pronuncia-se "R-zero"), nos diz o número de pessoas que irão contrair a doença a partir de uma única pessoa que já está com o vírus, ou seja, este indivíduo contaminado servirá como fonte da doença. Por exemplo, se o R0 é estimado em dois, cada pessoa doente transmitirá para outras duas aproximadamente. O R0 é calculado quando se tem uma população não vacinada, sem contato prévio com o patógeno e quando não há formas de controlar sua dispersão. O "novo coronavírus" SARS-CoV-2 se encaixa nestes pré-requisitos.

Os dados utilizados como fonte são providenciados pela Prefeitura de Sorocaba no seguinte link: `http://www.sorocaba.sp.gov.br/coronavirus/painel-grafico/`

O R0 foi calculado utilizando como base este artigo: `https://hal.archives-ouvertes.fr/hal-02509142v2/file/epidemie_pt.pdf`

Uma simples explicação sobre o Número Básico de Reprodução: `https://www.luciacangussu.bio.br/entenda_o_r0_na_covid-19_e_suas_consequencias/`

15.7 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/GuiMilani/covid-sorocaba`

15.8 Imagens

```php
function calculateR0($info)
{

    $initial_cases = (float) $info["casesStart"];
    $final_cases = (float) $info["casesEnd"];
    $r0 = $final_cases / $initial_cases;
    $r0 = log($r0);
    $r0 = $r0 / 7;
    $r0 = (1 + $r0 / 0.25) * ($r0 + 1);

    return $r0;
}
```

Figura 15.1: Função que calcula o R0 a partir dos valores dados

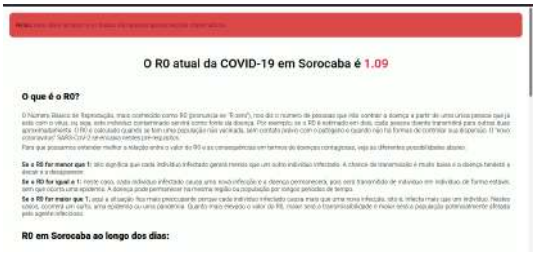


Figura 15.2: A página inicial do site

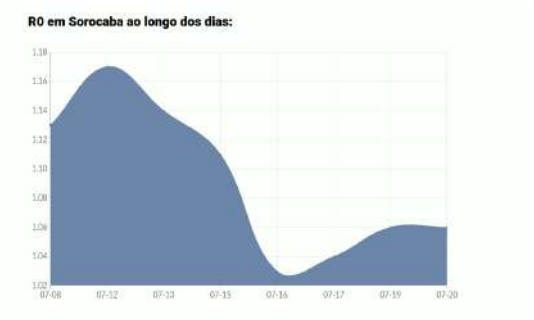


Figura 15.3: O gráfico do R0 na versão desktop

Figura 15.4: Um dos autores contatando Atila Iamarino no Twitter

Capítulo 16

CriPython

Gregório Fornetti Azevedo

16.1 Conceito

CriPython é um projeto sobre criptografia que tem o intuito de traduzir ou encriptar mensagens. O usuário que estiver usando o programa tem liberdade para escolher uma das cifras disponíveis e utiliza-la para encriptar/traduzir um texto, podendo ver na prática como que funcionam algumas cifras.

Além disso, nesse programa existem os utilitários, que são implementações que tentam desvendar uma mensagem encriptada sem a sua chave de tradução. E para finalizar, outro objetivo desse projeto é explicar um pouco sobre o assunto cifras de criptografia.

16.2 Pré-requisitos e recursos utilizados

Para o desenvolvimento desse projeto foi utilizado a linguagem *Python* e algumas bibliotecas, citadas logo abaixo:

1. *PySimpleGUI* [1], utilizado para criar a interface gráfica do programa

16.3 Passo a passo

1. Aprendi um pouco sobre a biblioteca PySimpleGUI e suas funções, métodos e objetos através da documentação da biblioteca [1] e por um vídeo [2]
2. Com os conhecimentos adquiridos através do passo anterior, comecei a criar a interface gráfica do programa.
3. Depois de criar uma interface básica, comecei a estudar mais sobre algumas cifras simples.
4. Com um pouco de conhecimento sobre programação e criptografia, implementei a lógica das cifras disponíveis no programa até agora.
5. Com algumas cifras implementadas, comecei a implementar os utilitários desse programa (força bruta César e Adivinhador César).

16.4 Instalação

Todos os arquivos do projeto estão disponíveis no repositório do mesmo, ou seja, clique em *Clone or download* e baixe os arquivos para instalar o programa em sua maquina.

16.5 Execução

Basta clicar duas vezes no executável *main.exe*

16.6 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/GregorioFornetti/CriPython`

16.7 Demais anotações

No repositório do projeto há uma wiki que explica um pouco sobre as cifras disponíveis e opções do menu da interface gráfica do programa. Ela está disponível em:

`https://github.com/GregorioFornetti/CriPython/wiki`

16.8 Imagens

Figura 16.1: Tela principal

Figura 16.2: Tela do menu encriptar

Figura 16.3: Tela do menu César

16.9 Bibliografia

[1] Pysimplegui, 2020. URL: https://pysimplegui.readthedocs.io/en/latest/.

[2] Pysimplegui - criando uma interface gráfica com python, 2020. URL: https://www.youtube.com/watch?v=Et0fYeA2XxY.

Capítulo 17

Emulador VSGBE

Adriano Emídio

17.1 Introdução

Este é um pequeno projeto que foi iniciado por hobby. Tudo começou com uma pequeno videogame caseiro baseado em CPLD e Microntrolador ARM Cortex-M4F que o autor realizou há alguns anos atrás, apesar de não ser poderoso, foi notado que talvez seria possível a emulação de algum videogame de 8 bits. Foi escolhido o GameBoy justamente pela relativa simplicidade do hardware e por ser um videogame que o autor possuí acesso ao hardware verdadeiro, sem contar a quantidade de informações disponíveis na internet. A primeira versão deste emulador foi escrita em Python, mas por uma questão de performance, foi decidido utilizar a linguagem C++.

17.2 Sobre o Emulador

Antes de mais nada, esse emulador é um projeto pessoal, e por isso, não se encontra muito organizado. Simples assim, o código é mal documentado, cheio de bugs e da forma que está, pouquíssimos jogos são compatíveis. Esta versão suporta apenas Linux, mas há a expectativa de fazer um MakeFile que suporte o sistema operacional Windows em breve.

17.3 Público Alvo

Qualquer um que entenda de C++ e queria mexer em um código de emulação para isso, o código está em licença z-lib, ou seja, pode ser utilizado para quais quer propósito. Da forma que está, este sofware não é um produto comercial, desta forma, se deseja apenas jogar um jogo sem se importar com detalhes técnicos, este emulador não é para você.

17.4 Bugs Conhecidos

Como já foi dito, este código é bagunçado e possui todos os tipos de problemas que um programa de computador pode ter entre eles, mas não limitado há: estouro de pilha, exceções não tratadas, ponteiros selvagens e etc. Este programa não deve ser utilizado de maneira alguma em modo de super usuário (`root`), e o autor não se responsabiliza por quaisquer danos ou perda de dados que possam acontecer devido à utilização deste programa.

17.5 Compilação

Para a compilação é necessário o compilador g++ e a biblioteca SDL na sua versão 2, dentro da pasta principal apenas digite `make` e arquivo de saída será `vsgbe.out`. O caminho para ROM a ser executada deve ser dado na linha de comando para chamar o arquivo. Ex: `./vsgbe.out minharom.gb`

17.6 Controles

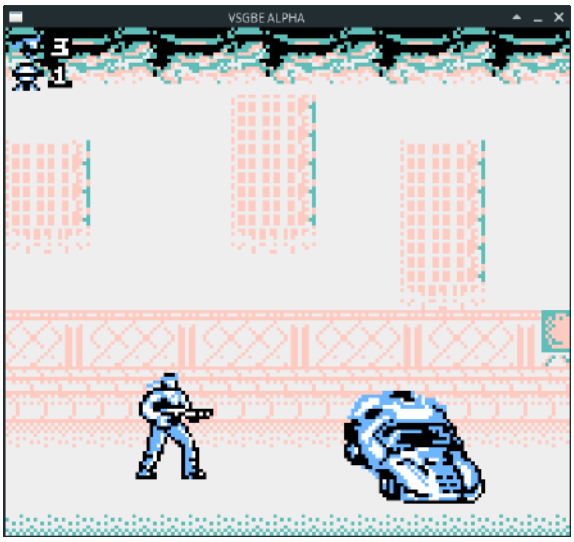
1. Direcional: Teclas direcionais;
2. Botão A: Tecla A;
3. Botão B: Tecla S:

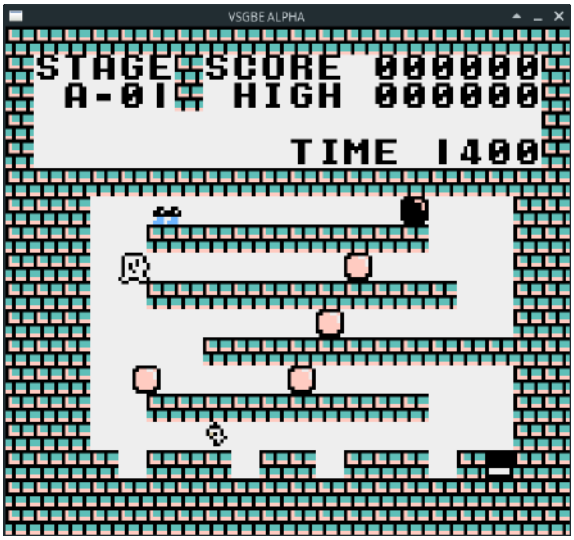
4. Start: Tecla X;
5. Mode: Tecla Z;

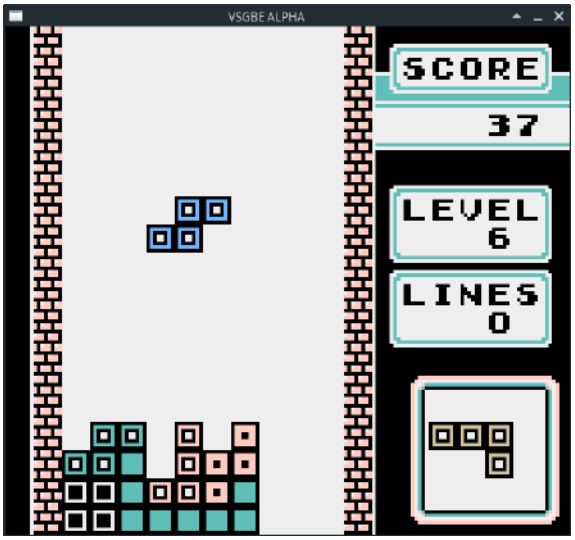
17.7 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em: `https://github.com/adrianoemidio/VSGBE`

17.8 Imagens







Capítulo 18

Impressora 3D DVD

Adriano Emídio
Anderson Pinheiro Garrote
Marcus Vinícius Natrielli Garcia
Vinícius Carvalho Venturini

18.1 Conceito

Este projeto se baseia no criação, desenvolvimento e montagem de uma impressora 3D, reaproveitando motores e drivers de leitores de CD/DVD e utilizando a placa Arduino Uno com um shield CNC.

O conceito inicial surgiu durante a atividade de extensão Hackerspace, da UFSCar Sorocaba, em 2019. O andamento do projeto ocorreu no ano seguinte, dentro da mesma atividade.

Esperamos, no final, ter uma impressora funcional que realize a produção de pequenos modelos 3D.

Os detalhes mais específicos de cada parte do projeto se encontram na Wiki do projeto.

18.2 Mecânica

A parte mecânica da impressora se baseia na utilização de diversos motores de passo, tanto os encontrados nos drivers de CD/DVD, quanto avulsos, reproduzindo os eixos X, Y e Z necessários para a impressão em 3D. Um motor de passo também será usado para movimentar o filamento para a extrusora.

18.3 Eletrônica

Para a eletrônica, a impressora funcionará principalmente a partir de uma placa Arduino Uno, a qual irá controlar os motores, sensores e demais componentes utilizando um shield CNC V3 e quatro drivers DRV8825.

18.4 Programação

A parte de controle por software será usada nas etapas de modelagem o objeto 3D em um programa de CAD, no processo de manufatura assistida por computador realizado por um programa de CAM e no firmware do Arduíno, responsável por transformar as instruções de movimentação do bico extrusor em sinais para os drivers dos motores.

18.5 Extrusora

Para a extrusão dos filamentos da impressora, será usado um módulo extrusor, chamado de HotEnd V6, e um motor de passo, depositando o filamento na extrusora. Utilizaremos materiais de até 1.75mm, respeitando os limites impostos pelas peças.

18.6 Passo a Passo

- Foram realizadas algumas reuniões com o grupo, procurando escolher as melhores tec-

nologias e abordagens para a fabricação da impressora;

- O grupo escreveu uma lista com os materiais que já possuíam, incluindo componentes eletrônicos, leitores de CD/DVD e motores;
- Divisão do projeto em partes, sendo elas: mecânica, eletrônica, de programação e de extrusão. Assim cada um poderia trabalhar separadamente e, aos poucos, unir cada uma das partes;
- Foram feitos testes com os motores de passo dos leitores, compreendendo como eles funcionavam e quais eram suas limitações;
- O grupo testou o shield CNC e o Arduino UNO, usando os motores de passo comuns que já possuiam (além dos leitores);
- Busca de quais são as melhores bibliotecas de Arduino para controlar o shield CNC e o circuito eletrônico;
- O grupo estudou os softwares necessários para realizar a impressão de uma peça 3D, passando por modelagem digital do objeto, conversão do mesmo para G-Code e comunicação com o Arduino;
- Foi comprado o módulo extrusor, uma das poucas peças que o grupo não possuia e que não poderia ser reaproveitada de outros meios;
- Foi testada a medição da temperatura do módulo extrusor

18.7 Bugs Conhecidos

Por enquanto, não foram encontrados bugs, erros ou falhas no projeto. O maior problema encontrado é a dificuldade na integração das partes, devido a necessidade do grupo trabalhar à distância e online durante a pandemia do CoVID-19. Muitas peças de uma mesma parte estão espalhadas entre os membros do projeto.

18.8 Resultado

Devido a pandemia atual, não o grupo não teve a oportunidade de juntar presencialmente todos os componentes e peças. No entanto, diversos bons resultados em cada parte separada do sistema foram obtidos.

O projeto está em fase de testes tanto para a extrusão quanto para a programação, buscando entender os detalhes de ambos e, futuramente, interligá-los.

A parte eletrônica está praticamente pronta, com todas as peças essenciais já em posse. Fala, todavia, progresso na união com as outras partes.

Por fim, a mecânica é a parte mais estagnada. Montar e planejar os eixos e a movimentação das peças necessita da união presencial dos membros do projeto.

18.9 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/AndersonGarrote/impressora_3D_DVD`

A wiki do projeto se encontra em:

`https://github.com/AndersonGarrote/impressora_3D_DVD/wiki`

Capítulo 19

Ins N' Outs

Maurício Cândido

19.1 Conceito

O projeto foi originado da ideia de substituir as velhas planilhas de cálculo em Excel e transformá-las em um aplicativo intuitivo e agradável, acessível para todos. Apesar do projeto não funcionar à base do Excel, a utilização do banco de dados (tanto online quanto offline) funciona basicamente na mesma maneira, apenas de uma forma não visual, que é traduzida posteriormente quando o aplicativo está funcionando.

Apesar de muitas *features* ainda estarem em um estágio inicial, o intuito é que o aplicativo sempre seja melhorado para a acessibilidade de todos, liberando-o (algum dia) para o público de forma gratuita.

19.2 Recursos Utilizados

- Android Studio como a plataforma principal, responsável pelo front-end e funcionamento do projeto;
- Foram utilizadas inúmeras bibliotecas base do próprio android studio, que podem ser encontradas em todas atividades;
- Firebase, como banco de dados online;
- SQLite, como banco de dados local.

19.3 Passo a Passo

- Projeção do Banco de dados Online;
- Projeção do Banco de dados Local;
- Criação do Design de cada página/feature;
- Criação dos algoritmos de autenticação e cálculo;
- Correção de bugs.

19.4 Instalação e Execução

- Baixe o projeto;
- Faça a build do projeto no Android Studio;
- Linkar o aplicativo com algum projeto no Firebase;
- Execute o projeto em um emulador;
- Tranforme o projeto em um APK e transfira-o para um dispositivo mobile.

19.5 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/yrmeww/InsNOuts`

19.6 Imagens


28/11
R$320.00
INSERIR
Adicionar
nova conta
Editar contas
InsNOuts™ 2020


Dívida
0
Anterior
FINALIZAR


Adicionar uma Conta
Nome da conta
Renda
Dívida
CRIAR

Capítulo 20

Macro-Keylogger

Guilherme Brante (@brantenosh)
Lucca Marques (@yllumi)

20.1 Conceito

ESTE PROJETO NÃO É MALICIOSO E TEM INTENÇÕES DIDÁTICAS.

O projeto consiste na implementação de um keylogger embutido em um software de macro com o intuito de camuflar as intenções do programa e roubar os dados dos usuários (vítimas) que utilizarem o software. As informações capturadas envolvem tudo que o usuário digita em suas aplicações e informações sobre o seu computador.

20.2 Pré-requisitos e recursos utilizados

Foi utilizado C# para a implementação do Keylogger.

Destaque para as seguintes bibliotecas:

- FluentFTP, utilizada para tratar as requisições FTP.
- Windows DLL:
- user32.dll, responsável pelos Hooks
- kernel32.dll, responsável pelos módulos dos Hooks

Foi utilizado Node.js para a implementação do back-end do website (painel de controle), além dos recursos de HTML/CSS. Destaque para as seguintes bibliotecas:

- Node.js
- Express;
- Consign;
- Promise FTP.

HTML:

- Bootstrap
- Font Awesome

20.3 Passo a passo de implementação

Pesquisa dos códigos-fonte de keyloggers em C/C++ e C#; Estudo de como os códigos-fonte funcionavam; Implementação do algoritmo de keylogger em C#; Implementação do painel de controle (website) estruturado pelo Node.js.

20.4 Instalação

- Abra a pasta *keylogger*;
- Execute o *setup.exe*;
- Confirme as janelas seguintes.

20.5 Execução

Quando instalado, o Keylogger irá abrir em sequência e juntamente a isso criará um registro no caminho `\HKEY\_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run`, definindo que o mesmo abrirá com a inicialização do Windows

20.6 Implementações

- Clipboard listener. Responsável por monitorar a movimentação do clipboard (crtl+c, ctrl+x);
- Screenlogger;
- Deixar menos detectável;
- Refatoração dos métodos de acesso FTP;
- Exibir e formatar no painel de controle os aruivos de log;
- Macro para mascarar o Keylogger

20.7 Bugs

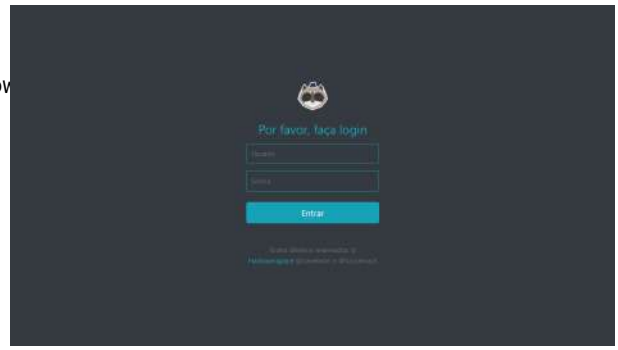
Caso o servidor FTP fique indisponível a aplicação C# e a Node.js não se conectarão ao mesmo, não enviando logs e gerando exceções nas requisições FTP na aplicação Node.js; Windows Defender reconhece como vírus;

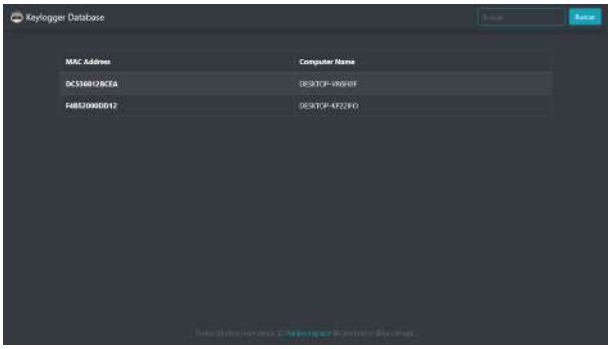
20.8 Repositório

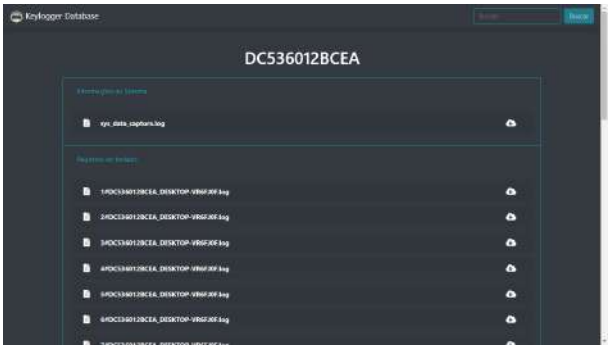
O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/cavebran/macro-keylogger`

20.9 Imagens







Capítulo 21

Predição de Diabetes com Machine Learning

Matheus Vargas Volpon Berto

21.1 Conceito

Esse é um projeto de Ciência de Dados e Machine Learning que visa predizer se um indivíduo possui diabetes tipo 2 utilizando algoritmos simples de classificação, como Árvores de Decisão, KNN e SVC. Esse projeto foi realizado para propósitos de estudo e para a atividade de extensão do HackoonSpace.

21.2 Pré-requisitos e recursos utilizados

A implementação se utiliza da linguagem Python com o Google Colab e as bibliotecas NumPy, Matplotlib, Pandas e Scikit Learn. Os modelos são treinados a partir da seguinte base de dados:

- *Early stage diabetes risk prediction dataset* — disponível em UCI Repository

21.3 Passo a Passo

- Coleta de dados e análise exploratória;
- Pré-processamento de dados, programação e regularização;
- Treinamento dos modelos;
- Avaliação e validação;

21.4 Instalação

- Baixe o repositório;
- Abra e execute os arquivos `.ipynb` no Google Colab ou Jupyter Notebook. É possível fazer testes e mudar os hiperparâmetros dos modelos para testar melhores resultados.

21.5 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/MathVolps/ML-Diabetes-Prediction---HackoonSpace-2020-2`

Um documento com mais explicações teóricas sobre o projeto pode ser encontrado no mesmo repositório, em:

`https://github.com/MathVolps/ML-Diabetes-Prediction---HackoonSpace-2020-2/blob/master/Report%20(pt-br).pdf`

Capítulo 22

Monika bot

Marcus Vinícius Natrielli Garcia

22.1 Conceito

O projeto Monika bot se baseia na criação de um bot de propósito geral para a plataforma Discord, utilizando-se de do módulo discord.js, em Node.js, para acessar a API da mesma e fornecer diversos comandos úteis para os usuários (principalmente do HackoonSpace).
Os principais intuitos deste programa são:

- Aumentar o engajamento dos usuários nos servidores do Discord.
- Facilitar a interação entre usuários e/ou bots.
- Automatizar o uso de ferramentas já disponíveis em outros meios computacionais.
- Desenvolver novas funcionalidades úteis em geral.

Além disso, a persona do bot é uma referência a personagem Monika, do jogo Doki Doki Literature Club! (2017). O autor do projeto seguiu todos os direcionamentos e recomendações presentes no site da desenvolvedora do jogo, a Team Salvato, sobre o uso de seus personagens em outras mídias e/ou formatos, bem como entrou em contato com a mesma para esclarecer eventuais dúvidas sobre o mesmo tema.

O autor deste projeto deixa claro que não possui quaisquer direitos oficiais sobre a personagem Monika e/ou o jogo DDLC. O uso de quaisquer informações, imagens, referências ou detalhes deles neste software se baseiam apenas no desejo de contribuir com a comunidade de fãs deste jogo, não desejando prejudicar a empresa detentora dos direitos legais, Team Salvato. O autor sempre estará aberto para conversar e resolver quaisquer possíveis conflitos existentes no uso destas propriedades intelectuais com a respectiva desenvolvedora.

22.2 Funcionalidades

Dentre as funcionalidades atualmente implementadas, a Monika pode:

- Cumprimentar usuários, enviar abraços, escrever mensagens e realizar outros comportamentos aleatórios.
- Entrar e sair de canais de voz.
- Mostrar uma lista completa de comandos (e fornecer mais detalhes sobre cada um eles).
- Mostrar a previsão do tempo para um determinado lugar.
- Enviar gifs de uma seleção aleatória.
- Realizar um ping TCP em um endereço na Web (e em uma determinada porta, se especificado).
- Rolar dados de diversos lados.
- Tocar músicas a partir de links do Youtube.
- eletar pessoas e trazê-las de volta (referência ao jogo DDLC)

A cada nova atualização, mais funcionalidades serão implementadas. Desta forma, a lista e a quantidade de recursos/dependências poderá se expandir.

22.3 Recursos utilizados

Para o desenvolvimento deste projeto, o recurso utilizado mais importante foi o framework Node.js, que possibilita o uso da linguagem Javascript para aplicações servidoras. No entanto, o gerenciador de pacotes npm também foi de suma importância, possibilitando a instalação de módulos fundamentais para a implementação dos comandos propostos.

Segue a lista atual de dependências do projeto:

- @discordjs/opus.
- axios.
- discord.js.
- ffmpeg-static.
- pg.
- remove-accents.
- tcp-ping
- ytdl-core-discord.

Mais detalhes sobre as versões utilizadas de cada dependência e do framework em questão se encontram no arquivo package.json do repositório do projeto.

Outros recursos valiosos para o desenvolvimento deste projeto foram o tutorial disponível em Discord.js Guide, para melhor compreensão de como começar a construir bots para a plataforma Discord, e a documentação disponível em `discord.js.org`, para consultar mais informações sobre o módulo discord.js.

Algumas funcionalidades utilizam acesso a banco de dados para armazenar, registrar e consultar alguns dados dos usuários dos servidores aos quais o bot servirá. A tecnologia usada para tal fim foi o PostgreSQL, a partir do módulo pg, citado na lista de dependências. Atualmente, as credênciais de acesso ao banco se encontram no arquivo database.js, no diretório functions.

Também é importante observar que o comando de previsão do tempo necessita de um token para a API do site OpenWeather. Atualmente, isto pode ser adicionado ao modificar a o texto MYAPIID no arquivo weather.js no diretório commands/utils.

22.4 Passo a passo

Para o desenvolvimento deste projeto, foram realizados os seguintes passos:

- Uma versão inicial do programa foi criada seguindo o passo a passo detalhado no guia Discord.js Guide.
- O primeiro comando, de envio de arquivos gif, foi criado, buscando-se compreender o funcionamento da API do Discord.
- Demais comandos básicos foram sendo construidos, como os de ajuda e cumprimento de usuários.
- Refatorou-se o arquivo principal index.js, de maneira a colocar cada função do bot em um módulo JavaScript separado.
- A conexão com banco de dados foi criada, armazenando dados sobre as interações dos usuários do servidor do HackoonSpace.
- O sistema de níveis e pontos de experiência foi implementado, visando a existência de comandos exclusivos para determinados níveis.
- Comandos mais complexos, como de conexão com a API de previsão do tempo e de deletar usuários foram sendo implementados, usando recursos mais avançados do Node.Js.
- Uma microeconomia, baseada em moedas ficticias (HackoonCoins) foi criada, possibilitando que alguns comandos só pudessem ser utilizados se o usuário consumisse algumas dessas moedas.
- Alguns tratamentos mais avançados para Promises e comportamentos assíncronos foram adicionados.

Diante da continuidade deste projeto, outros passos, estudos e reformulações ainda podem acontecer.

22.5 Instalação

1º Passo:

Baixe e instale o Node.js, preferencialmente na mesma versão apresentada no arquivo package.json. Se tudo der certo, o gerenciador de pacotes npm será instalado automaticamente junto do Node

2º Passo:

Clone/copie todos os arquivos deste repositório em um diretório na sua máquina

3º Passo:

Execute o comando npm install para instalar todas as dependências presentes em package.json

4º Passo:

Preencha todos os campos necessários no arquivo config.json.

5º Passo:

Preencha os demais campos com tokens e chaves de configuração do projeto:

- process.env.BOT_TOKEN, em index.js, para autenticar o bot na API do Discord.
- process.env.ROLE e process.env.WELCOME_CHANNEL, em index.js, para configurar qual cargo padrão novos usuários irão receber e qual mensagem de boas-vindas deverá aparecer.
- (OPCIONAL) process.env.process.env.BOTS_ONLY_CHANNEL, em index.js, caso o bot necessite responder apenas em um único canal exclusivo.
- Os campos de acesso ao banco de dados, em functions/database.js, para usufruir do sistema de níveis e moedas.
- O campo MYAPIID, na chamada da API Open Weather, em commands/utils/weather.js, para o comando de previsão do tempo

6º Passo:

Crie as tabelas necessárias no mesmo banco de dados PostgreSQL das credenciais fornecidas na etapa anterior.

Os códigos SQL necessários para a criação das tabelas do banco de dados se encontram no arquivo database.sql.

Observação: espera-se, em futuras versões deste projeto, simplificar algumas etapas de instalação.

22.6 Execução

Execute os comandos *npm start* ou *node .* para hospedar o bot na sua máquina.

Se tudo der certo, a mensagem "Estou na sua realidade!"deve aparecer no seu terminal.

22.7 Bugs e Problemas Conhecidos

Diante do desenvolvimento contínuo de novas funcionalidades para o bot, existem grandes chances de bugs e problemas futuros serem encontrados.

Até o momento, os problemas mais notórios a serem resolvidos são:

- O comando *-myinfo* possui alguns problemas de visualização em dispositivos móveis.
- O comando *-help*, a medida em que novas funcionalidades estão sendo implementadas, retorna uma lista extensa de comandos, não sendo muito amigável para ler.
- A busca por localidades com o comando *-weather* pode, em algumas situações, não funcionar. Isto ocorre por problemas envolvendo acentos e traduções de nomes de países, cidades e demais lugares, dado que a API que fornece os dados de previsão do tempo foi feita originalmente em inglês.
- Instalação e configuração do projeto possui alta complexidade. Diversos dados necessitam ser reorganizados para simplificar este processo.

22.8 Repositório

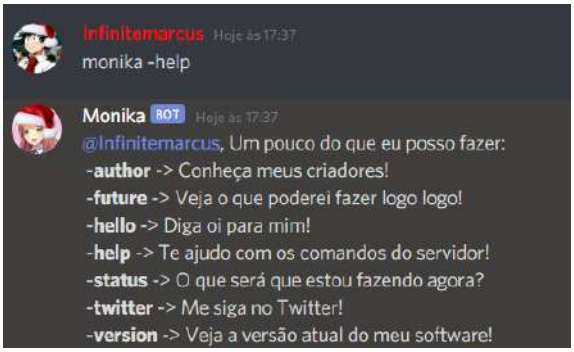
O repositório com os arquivos e demais informações sobre o projeto se encontra em:
`https://github.com/Infinitemarcus/Monika-bot`

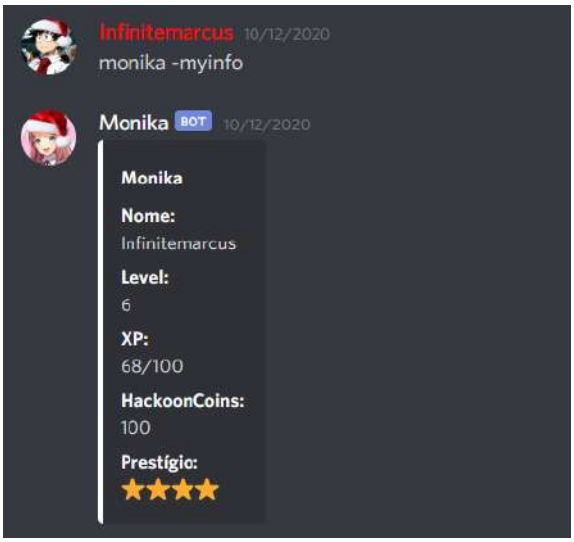
22.9 Demais anotações e referências

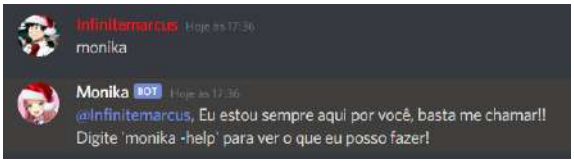
Seguem algumas últimas informações e observações sobre o projeto:

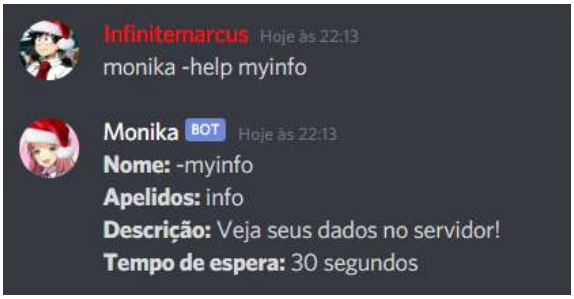
- Não existem planos concretos, por parte do autor do projeto, de hostear o bot publicamente no momento. Existem diversos gastos envolvidos e otimizações necessárias que ainda não são pertinentes.
- Este README pode se tornar obsoleto ao longo do tempo, em vista da necessidade de documentar novas alterações. Novas atualizações virão com o tempo, entretanto provavelmente irão demorar.
- O autor do projeto, como qualquer outra pessoa, não é perfeito. O projeto sempre está aberto para sugestões, recomendações, alterações, correções e afins. Então sinta-se livre para entrar em contato direto ou utilizar as próprias ferramentas do GitHub para tal.

22.10 Imagens/screenshots









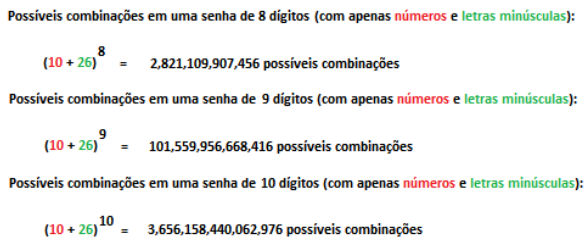
Capítulo 23

Obscrypto

Fernando Favareto Abromovick (kyleflick124)

23.1 Conceito

O Projeto foi desenvolvido mais como uma introdução para o mundo da criptografia e segurança. Embora fazer uma senha pessoal pareça ser algo extremamente seguro e específico, existem diversos softwares, maliciosos ou não, que conseguem obter quaisquer senhas em questão de horas. Sabendo disso, uma solução fácil porém nada agradável a um usuário da internet seria fazer uma senha com cada vez mais e mais caracteres "aleatórios". Pelos conceitos básicos da matemática aplicados em combinações, aumentar apenas um caractere em uma senha já aumenta consideravelmente o número de possíveis senhas, como observado a seguir:



Como pode ser visto, existem enormes possibilidades de senhas apenas com números e letras minúsculas, e caso fosse usado um número infinito de caracteres, quem sabe infinitas possibilidades existiriam?

No entanto, você poderia estar se perguntando: Como então tantas pessoas são hackeadas em tantos sites que se dizem seguros, se existem tantas possibilidades de senha assim?

A resposta para isso está no jeito que criamos senhas: Não fazemos senhas com caracteres aleatórios pois precisamos lembrar da senha para usá-la em algum lugar. Por exemplo, é bem mais fácil lembrar da senha iloveyou ou nicole123 do que 1ft3kr5t648ref5 ou qualquer outra senha desse tipo. Assim, os softwares de força bruta (softwares que tentam descobrir senhas por tentativa e erro) costumam usar wordlists, como a rockyou.txt, pois possuem senhas comuns usadas por muitas pessoas.

Pensando nisso, o projeto de criptografia obscura busca mostrar um "novo"caminho de tornar as senhas cada vez mais seguras: A partir do fornecimento de uma palavra ou de uma senha comum usada por alguma pessoa, criptografar as senhas de uma forma que façam uma cadeia consideravelmente grande de caracteres "aleatórios"gerado a partir dessa palavra, usando diferentes métodos de criptografia para tal.

23.2 Pré-requisitos e recursos utilizados:

A programação foi feita 100% em python, tudo que é necessário é ter uma versão do python 3, recomendo que seja a mais recente. A interface do programa foi construída dentro do código, com

a importação de uma biblioteca já existente chamada Tkinter

23.3 Passo a passo:

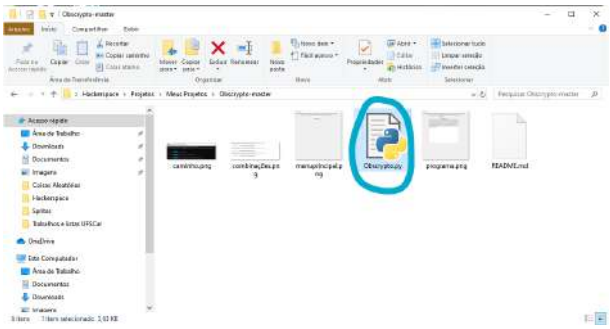
- Estudei um pouco como funcionava a biblioteca do Tkinter e como eu poderia usar para deixar meu programa simples e agradável
- Estudei alguns tipos de criptografia que geram números não importando a senha inserida, para que qualquer senha possa ser usada mesmo levando em conta caracteres especiais, gerando como se fosse um padrão/base para não dar conflito com os diferentes tipos de criptografia.
- Implementei esses tipos em uma sequência que gerasse uma longa cadeia de caracteres bem encriptados, parecendo como "aleatórios"
- Depois, adaptei tudo para as funcionalidades do Tkinter, como um botão chamar uma função, exibir as caixas de texto, etc.

23.4 Instalação

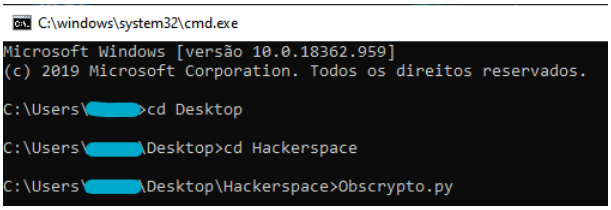
Após instalar o python, usando o GitHub pelo seu browser, clique no botão verde "Code"e em "Download ZIP". Após a conclusão do download, extraia a pasta para a sua área de trabalho.

23.5 Execução

O jeito mais fácil é clicando duplamente no programa Obscrypto.py, na pasta dos arquivos.



Outra forma também é abrindo seu terminal de comando, mude o caminho até chegar na área de trabalho, depois para a pasta do arquivo, e digite Obscrypto.py, apertando Enter em seguida: Isso



abrirá automaticamente o programa, siga as instruções exibidas na interface dele para usá-lo.

23.6 Bugs/problemas conhecidos

Por enquanto não há nenhum bug visual ou de sintaxe no programa que eu tenha percebido, mas qualquer problema com o programa basta mandar uma mensagem por email ou pelo discord/whatsapp do HackoonSpace caso faça parte. `https://hackoonspace.com`

23.7 Demais anotações e referências:

Uma boa parte do programa está comentado, o que ajuda as pessoas a entenderem como cada parte funciona sem precisar de explicação.

Os métodos de criptografia usados para converter a senha foram, nesta ordem, ascii, octal e base64.

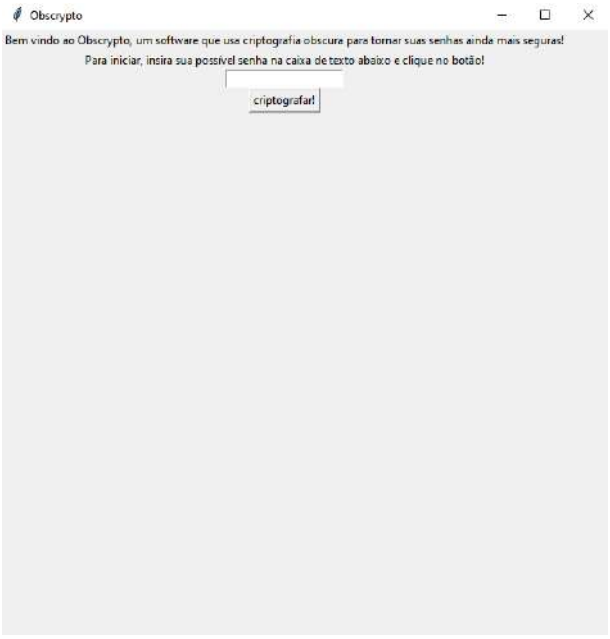
Um dos vídeos que mais me ajudou a entender como funciona o Tkinter foi esse, disponível em inglês, feito por Robert Jomar Malate e pelo curso de Harvard CS50, que inclusive recomendo muito o canal deles para quem quer aprender programação de maneira interativa, o conteúdo é todo em inglês mas a maioria das coisas possuem legenda em português.

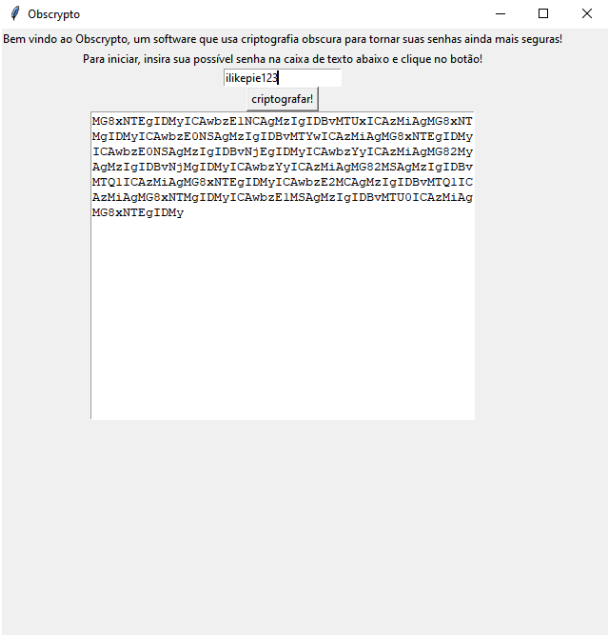
23.8 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/kyleflick124/Obscrypto`

23.9 Imagens


Obscrypto
Bem vindo ao Obscrypto, um software que usa criptografia obscura para tornar suas senhas ainda mais seguras!
Para iniciar, insira sua possível senha na caixa de texto abaixo e clique no botão!
criptografar!


Obscrypto
Bem vindo ao Obscrypto, um software que usa criptografia obscura para tornar suas senhas ainda mais seguras!
Para iniciar, insira sua possível senha na caixa de texto abaixo e clique no botão!
ilikepie123
criptografar!
MG8xNTEgIDMyICAwbzE1NCAgMzIgIDBvMTUxICAzMiAgMG8xNT
MgIDMyICAwbzE0NSAgMzIgIDBvMTYwICAzMiAgMG8xNTEgIDMy
ICAwbzE0NSAgMzIgIDBvNjEgIDMyICAwbzYyICAzMiAgMG82My
AgMzIgIDBvNjMgIDMyICAwbzYyICAzMiAgMG82MSAgMzIgIDBv
MTQlICAzMiAgMG8xNTEgIDMyICAwbzE2MCAgMzIgIDBvMTQlIC
AzMiAgMG8xNTMgIDMyICAwbzE1MSAgMzIgIDBvMTU0ICAzMiAg
MG8xNTEgIDMy

Capítulo 24

Projeto Site Hackeável HackerSpace UFSCar

Vinícius Carvalho Venturini
Marcus Vinícius N. Garcia

Este projeto consiste em um site onde todo o conteúdo do HackerSpace será armazenado e, juntamente a isso, terá também uma página denominada "Try Hack Me"onde o usuário poderá tentar hackear aquela parte do site (ainda não implantado).

24.1 Conceito

O projeto teve como base, ser um lugar onde o conteúdo do HackoonSpace ficasse armazenado, juntamente com ser um local onde os usuários pudessem treinar suas habilidades de hacking. O projeto foi um ótimo lugar para treino das habilidades com HTML 5 e CSS 3 e para entender quais são os perigos nos quais se deve atentar ao criar um site para que não fique vulnerável a certos tipos de ataque.

24.2 Problemas com o projeto

Por o projeto ter sido feito quando o autor tinha pouco conhecimento com web (principalmente no back end), o site está bem superficial com páginas apenas em HTML puro, por conta disso a página do *Try Hack Me* não foi feita, o que tende a ser alterado no futuro, com possíveis novas implementações e uma remodelagem do site para uma melhor experiência ao usuário.

24.3 Hospedagem do site

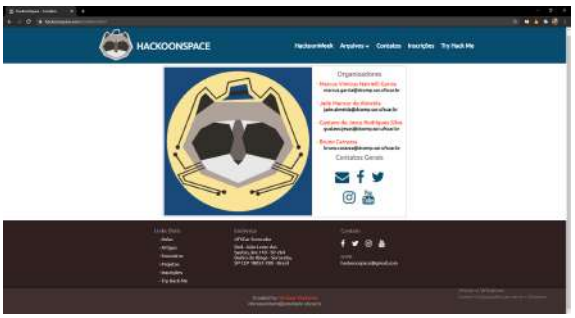
O site está hospedado no Domínio:

`https://hackoonspace.com`

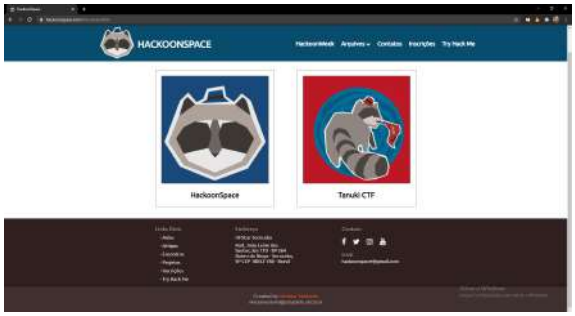
24.4 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/Vinicius-Venturini/Projeto-HackerSpace`




HACKOONSPACE
APRESENTAÇÃO DA ATIVIDADE
PRIMEIRO
ENCONTRO


HACKOONSPACE
HackoonSpace
Tanuki CTF

Capítulo 25

Verificador de Senhas

Caio César Brandini da Silva

25.1 Conceito do projeto

Este projeto foi desenvolvido com o intuito de ajudar pessoas em seu dia a dia a verificar a eficácia de suas senhas, assim, encontrando a forma mais segura e prática de manter a integridade de seus aparelhos, contas e arquivos pessoais.

Para isto, foi implementado um programa que testa a força de senhas e gera senhas seguras.

25.2 Dependências

Foi utilizada a linguagem *Python* para desenvolver a implementação geral do projeto, com a necessidade de importar os seguintes módulos:

1. *time.py*
2. *random.py*
3. *tqdm.py*
4. *_thread.py*
5. *os.py*

25.3 Passo a passo

1. Buscamos uma necessidade pública para ser selecionada
2. Pesquisamos recomendações de requisitos para a avaliação da força de senhas em empresas confiáveis como Google e Microsoft
3. Procuramos listas de senhas mais comuns, tanto nacionais quanto internacionais, e juntamos em um único arquivo .txt
4. Implementamos as funcionalidades na linguagem *Python*

25.4 Instalação

1. Baixe a pasta contendo o código e a lista
2. Abra o terminal na pasta do arquivo
3. Execute através do terminal com *Python Verificador de senha.py*

25.5 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

`https://github.com/caiobrandini/HackoonSpace`

25.6 Imagens

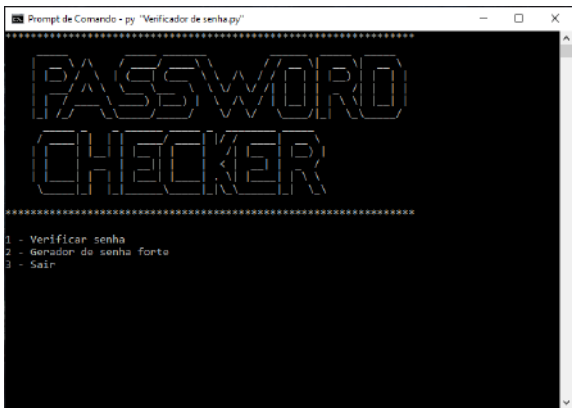


Figura 25.1: Tela de início

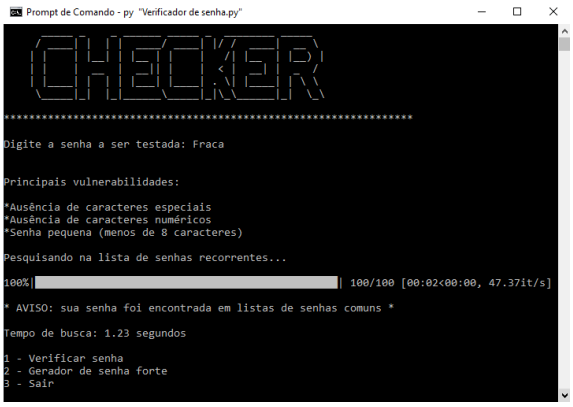


Figura 25.2: Tela de verificação de senha

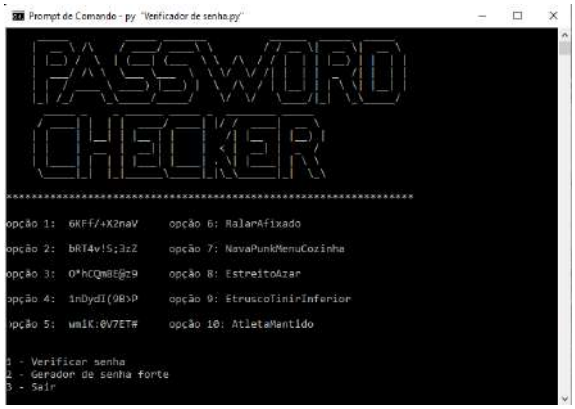


Figura 25.3: Tela de geração de senha forte

Capítulo 26

Vitto

Marcus Vinícius Caruso Leite

26.1 Conceito

Um sistema de assistente pessoal que faz recomendação de filmes de acordo com a sugestões de pistas de vontade para assistir, executando uma pesquisa dentro de um banco de dados de mais de 400 mil filmes para encontrar a produção melhor enquadrada as pistas entregues.

26.2 A TMDB

A The Movie Database é uma comunidade aberta feita para reunir diversos dados sobre filmes e séries gratuitamente desde 2008. Com isso, a plataforma consegue abrangir diversos campos da indústria do cinema com notas, títulos, sinopses, cartazes e pôsteres em alta resolução, fanarts, descrições, data de lançamento e vários tipos de dados que servem para aplicações comunitárias e de ajuda social, como o Vitto!

26.3 O Sistema

O sistema foi feito como uma aplicação web que engloba tanto o back-end quanto o front-end, usando PHP com uma API de banco de dados de filmes chamada "The Movie Database"(TMDb). Além disso, implementei animações e recursos estáticos com HTML, CSS, Javascript, JQuery e fiz requisições assíncronas do algoritmo back-end por meio de Ajax.

O algoritmo executa um tratamento de textos com espaçamentos, vírgulas e caracteres de separação para formar, em cada palavra, uma pesquisa dentro da database da API e retornar o ID de tipo de filme.

Todo o programa feito em PHP utiliza o cURL como ferramenta e organiza a engine de pesquisa por meio de ID associada a cada espaço preenchido:

- Gênero.
- Tipo.
- Elenco.
- Duração.
- Época.

26.4 Disposições finais

Link do site em funcionamento:

`https://vitto-filmes.herokuapp.com/`

26.5 Repositório

O repositório com os arquivos e demais informações sobre o projeto se encontra em:

https://github.com/SorrisoPraFoto/design_vitto

Figura 26.1: Página da aplicação mostrando o filme com a nota, sinopse e título.

Capítulo 27

WormVirus

Caio César Brandini da Silva
Fernando Favareto Abromovick
Vinícius Carvalho Venturini

27.1 Conceito do Projeto

O Projeto foi desenvolvido como um teste para observar as capacidades que um "vírus caseiro"tem de penetrar nos arquivos e diretórios de um computador. Para fazer o projeto, pensamos em incorporar o vírus como um executável, que seria baixado por um usuário qualquer da internet.

Para ilustrar os casos nos quais isso ocorre, o vírus foi inserido em uma versão "crackeada"do jogo Among Us, algo que alguém possivelmente baixaria um arquivo zip ou rar e rodaria o executável sem conferir todas as pastas no arquivo baixado. Sendo um vírus de ação direta, a proposta do vírus é simples: Receber informações sobre a localização e a provedora de internet do usuário e executar uma série de comandos em seu computador (a partir da biblioteca OS do python), quando o jogo for executado, a partir de um arquivo batch (.bat) rodando minimizado na pasta /Users .

A partir desses comandos, a ideia seria procurar por diversos arquivos importantes baseados em seus tipos (txt, json, etc.) e enviar uma cópia deles para o computador do servidor. No entanto, para tornar o programa menos perigoso para testes, ele apenas procura por um txt específico no computador do usuário, nomeado por flagHackoon.txt. Ao ser encontrado, o conteúdo do arquivo é copiado para um txt no servidor, nomeado pelo IP do cliente.

27.2 Pré-requisitos e recursos utilizados

A programação foi feita 100% em python, mas como o arquivo foi transformado em um executável, na teoria não seria necessário ter Python instalado. no entanto, como o programa não foi testado em computadores sem python, pode ser necessário ter uma versão do python 3, recomendamos que seja a mais recente.

27.3 Passo a passo

1. Estudamos e aprendemos sobre as seguintes bibliotecas e aplicações do python: import socket. import os. import sys. import glob. import random.

2. Construímos 2 scripts de python: 1 para o lado do cliente e outro para o lado do servidor. Os conteúdos de cada script serão disponibilizados nas imagens ao fim do repositório.

3. Os comandos possíveis de se usar em Windows para a cópia e transferência do conteúdo de arquivos foram estudados pelo site oficial da Microsoft [1].

4. O programa foi convertido para um executável usando a biblioteca pyinstaller [2].

27.4 Instalação

Após instalar o python, usando o GitHub pelo seu browser, clique no botão verde "Code"e em "Download ZIP". Após a conclusão do download, extraia a pasta para a sua área de trabalho. OBS: O arquivo com o Among Us é muito grande para ser exibido no github.

27.5 Execução

Para executar o programa, é apenas necessário dar clique-duplo no executável do Among Us, e o worm abriria no computador do cliente.

27.6 Bugs/problemas conhecidos

O programa deveria ser executado com a execução do atalho do Among Us, mas o atalho não funciona por motivos desconhecidos, provavelmente no caminho do arquivo. Para resolver, é só executá-lo pelo AmongUs.exe, dentro do AmongUs_Files. Qualquer problema com o programa basta mandar uma mensagem por email ou pelo discord/whatsapp do HackoonSpace caso faça parte, ou para o GitHub de algum dos autores do projeto.

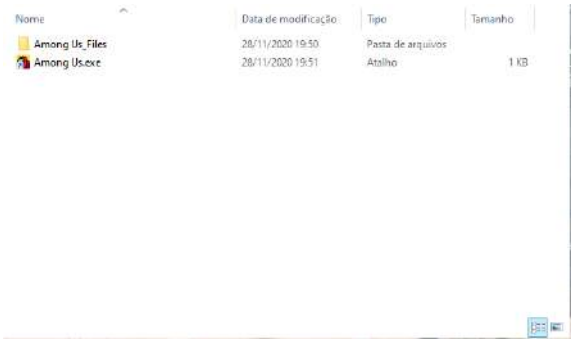
IMPORTANTE: O projeto foi feito apenas pelos integrantes do hackoonspace para ser testado e usado pelos mesmos. Por esse motivo, o projeto teve seu IP e porta de hospedamento alterados e removidos, para segurança e privacidade dos criadores do vírus.

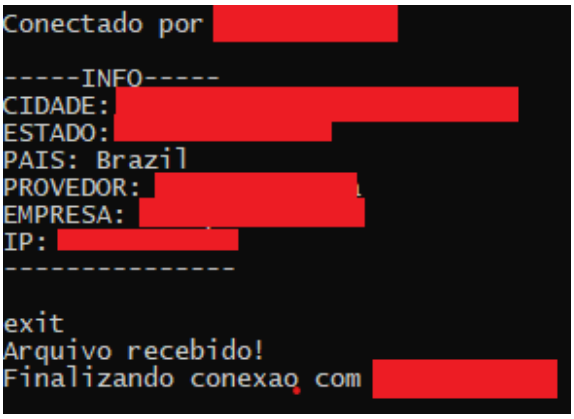
27.7 Repositório

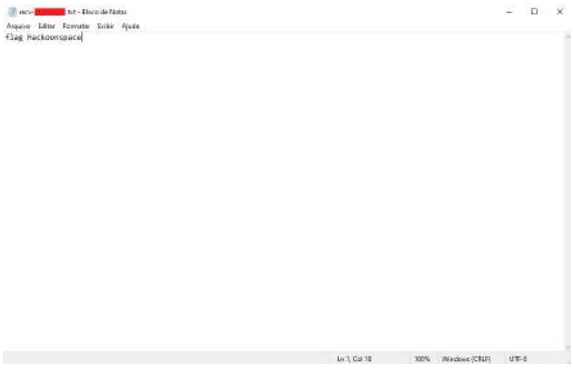
O repositório com os arquivos e demais informações sobre o projeto se encontra em:

https://github.com/kyleflick124/WormVirus

27.8 Imagens e Screenshots do programa







27.9 Bibliografia

[1] Comandos do windows - microsoft. URL: https://docs.microsoft.com/pt-br/

windows-server/administration/windows-commands/windows-commands.

[2] Using pyinstaller. URL: https://pyinstaller.readthedocs.io/en/stable/usage.html.