CENTRO UNIVERSITÁRIO SAGRADO CORAÇÃO

RAFAEL DE OLIVEIRA GARCIA

ESTUDO DE TÉCNICAS DE INVASÃO E SEGURANÇA DE SISTEMAS DE INFORMAÇÃO ONLINE

BAURU
2021

RAFAEL DE OLIVEIRA GARCIA

ESTUDO DE TÉCNICAS DE INVASÃO E SEGURANÇA DE SISTEMAS DE INFORMAÇÃO ONLINE

Monografia Científica apresentado a Pró-Reitoria de Pesquisa e Pós-graduação do Centro Universitário Sagrado Coração, sob orientação do Prof. Me. Vinicius Santos Andrade.

BAURU
2021

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD

Garcia, Rafael de Oliveira

G216e

Estudo de técnicas de invasão e segurança de sistemas de informação online / Rafael de Oliveira Garcia. -- 2021.

39f. : il.

Orientador: Prof. M.e Vinicius Santos Andrade

Monografia (Iniciação Científica em Ciência da Computação) - Centro Universitário Sagrado Coração - UNISAGRADO - Bauru - SP

1. Segurança da informação. 2. Técnicas de invasão. 3. Técnicas de defesa. I. Andrade, Vinicius Santos. II. Título.

Elaborado por Lidyane Silva Lima - CRB-8/9602

RESUMO

É cultural que boas empresas adotem diversas regras de operação, com o intuito de se proteger de ataques cibernéticos. Estas diretrizes, antes se restringiam a controle de entrada através de níveis de autorização, fiscalização dos funcionários para impedir a saída de documentos importantes e proteção contra as invasões físicas à empresa, por exemplo. Com a evolução tecnológica, vários tipos de serviços foram criados, seja para transporte, alimentação, bancos, redes sociais etc. as informações pessoais dos usuários se encontram disponíveis dentro dos servidores das empresas donas destes sistemas, e para garantir a privacidade e a integridade dos dados, as empresas têm de tomar medidas para garantir que seus clientes estejam protegidos. Este trabalho tem enfoque em apresentar as técnicas mais comuns de invasão, quais são as falhas exploradas pelos mesmos e demonstrar as contramedidas e ferramentas usadas para a proteção das informações presentes dentro de um ambiente computacional.

Palavras-chave: Segurança da Informação. Técnicas de Invasão. Técnicas de Defesa

ABSTRACT

It is cultural that good companies adopt several operating rules, in order to protect themselves from cyber attacks. These guidelines were previously restricted to entry control through authorization levels, supervision of employees to prevent the exit of important documents and protection against physical invasions of the company, for example. With technological evolution, several types of services were created, whether for transport, food, banks, social networks etc. users' personal information is available on the servers of the companies that own these systems, and to ensure privacy and data integrity, companies must take steps to ensure that their customers are protected. This work focuses on presenting the most common techniques of invasion, which are the flaws exploited by them and demonstrate the countermeasures and tools used to protect information present within a computing environment.

Keywords: Information security, invasion techniques, defense techniques.

LISTA DE ILUSTRAÇÕES

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

DDOS – Distributed denial of servisse
DMZ – Demilitarized zone
DPI – Deep packet inspection
IOT – Internet of things
IP – Internet Protocol
NAT – Network adress translation
PDP – Packet Data Protocol
TCP – Transmission Control Protocol

SUMÁRIO

1 INTRODUÇÃO E REVISÃO DA LITERATURA

É cada vez mais comum a preocupação das empresas em manter a segurança de seus sistemas da informação, pois nestes, estão diversos dados confidenciais relacionados à própria instituição e seus clientes (usuários).

Ataques a sistemas de informações nem sempre estão relacionados ao furto de algum dado, em diversos casos, os ataques são feitos com o intuito de interromper algum serviço online, por exemplo.

Para compreender como os ataques ocorrem e como combatê-los, se faz necessário o estudo das técnicas de invasão, permitindo conhecer quais vulnerabilidades são exploradas e com isso, definir qual é a melhor forma de manter os sistemas de informações íntegros e disponíveis.

Sendo assim, o conteúdo das próximas seções fundamentam este projeto de IC, visando o melhor compreendimento de conceitos fundamentais para a continuidade da pesquisa.

1.1 SEGURANÇA DA INFORMAÇÃO

Para Lyra (2008), quando falamos de segurança da informação estamos falando de medidas tomadas para que uma série de princípios e aspectos sejam cumpridos, como, disponibilidade, integridade, confidencialidade entre outros. Tais aspectos serão explicados a seguir.

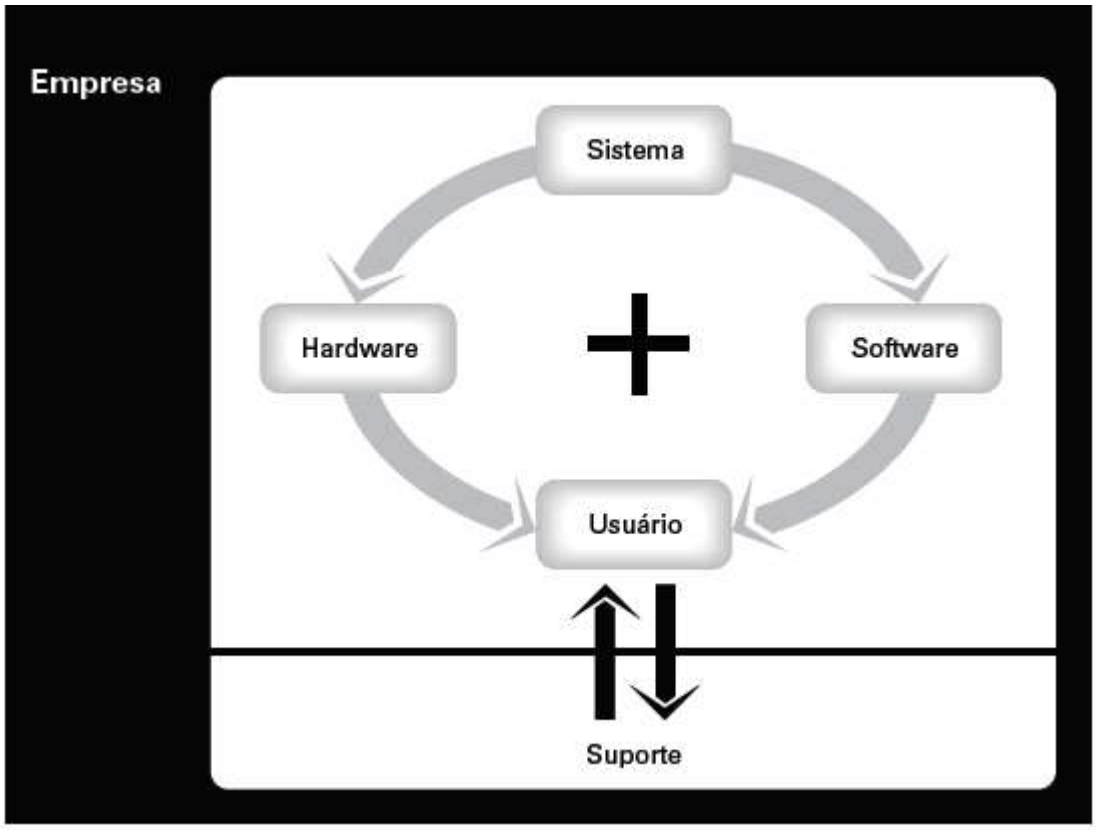
a) Aditável: nenhum sistema é imune a invasões ou ataques, um bom sistema é capaz de registrar o que foi feito e quem o fez, esta medida permite detectar mal uso por parte dos funcionários e identificar ataques realizados contra o sistema.
b) Disponibilidade: o sistema sempre tem de estar disponível e entregar a informação requisitada pelo usuário.
c) Confidencialidade: os dados são uns dos bens mais preciosos de uma empresa, por isso, garantir que apenas aqueles que realmente devem ter acesso a mesma, é extremamente importante.
d) Integridade: dados tem como função produzir informação, para que a mesma seja correta, o sistema tem que garantir que a informação ali inserida está correta e nos parâmetros certos.

Muitas empresas dependem diretamente das informações contidas em seu sistema ou possuem informações sigilosas que assim devem ser mantidas para o bom funcionamento da empresa.

1.2 SISTEMA DE INFORMAÇÕES

Segundo Polloni; Enrico; Fedeli (2010), sistema de informação é um conjunto de programas que atuam em determinado computador, operado por usuários, voltados a resolução de necessidades de uma determinada empresa, contando com um serviço de suporte a sistemas. A figura 1 ilustra esse conceito.

Figura 1 – Sistema de informação



Fonte: Polloni; Enrico; Fedeli (2010).

Um sistema de informação, segundo STAIR (1998), é formado por vários componentes que tem como intuito receber dados, manipular, armazenar, mostrar e emitir um relatório daquilo que foi inserido no mesmo.

Segundo Pereira e Fonseca (1997), um sistema de informação é uma ferramenta indispensável para o processo de uma empresa, uma vez que é o meio que permite a agilização de processos e as tomadas de decisão dentro da empresa,

o que por muitas vezes, é um fator crucial para o sucesso de uma empresa. O processo de desenvolvimento de um sistema de informação passa por várias etapas, dentre elas:

a) Identificação de uso e levantamento de requisitos: Nesta parte do desenvolvimento é analisado o propósito do software e quais funções ele deve realizar para atender a necessidade do caso.
b) Mecanismo de obtenção de dados: Aqui é pensado em como as informações serão inseridas, seja interna ou externa, sempre se preocupando em como manter a integridade dela.
c) Tratamento: Nem toda informação virá no formato mais adequado ou padrão do sistema, aqui é onde o processo de padronização dos dados garantirá que os dados fiquem adequados ao sistema, se preocupando sempre com a confidencialidade e integridade da informação.
d) Disseminação da informação: Elaboração do plano que garantirá que a informação esteja disponível aqueles que necessitam dela.
e) Uso: Aqui a informação atinge seu ápice sendo o momento no qual a informação de fato terá valor para empresa, se tornando um ativo dela, sempre respeitando os aspectos estabelecidos, disponibilidade, integridade e confidencialidade.
f) Armazenamento: Uma empresa sempre tem de se preocupar em como seus dados são guardados, seja para proteção de ataques ou por acidentes como incêndios. Um bom sistema de defesa e backup são itens essenciais para esta etapa.

1.3 TRABALHOS RELACIONADOS

Durante o processo de revisão da literatura, observou-se que esta temática é um tópico relevante, sendo encontrado vários estudos na área, como a pesquisa desenvolvida por Assunção (2009), que mostram o uso de *HoneyPots* com o intuito de estudo de vulnerabilidades dentro de um sistema e projeto de TCC de (Ranieri Marinho de Souza), que faz um abordagem das boas práticas na segurança da informação e apresenta ferramentas e técnicas na proteção de um sistema, o artigo

escrito por Angrishi (2017) expõe o risco da expansão do IoT e como a mesma pode se tornar uma ferramenta poderosa para a criação de *botnets*.

A segurança da informação, é um tópico que apenas ascende, quanto mais a tecnologia mais a informação se torna o poder e um ativo de valor inestimável para as empresas, desta forma o investimento em pesquisa e em técnicas de proteção é tão procurado.

Muitos governos preocupados com a segurança das informações dos usuários, no Brasil isto foi marcado em dois momentos “O marco civil da internet” e mais recentemente com a “Lei geral de proteção de dados” de 2018 que passará a ter vigor agora em 2020.

2 OBJETIVOS

A seguir são descritos os objetos gerais e específicos que norteiam essa pesquisa.

2.1 OBJETIVO GERAL

Avaliar e comparar resultados obtidos através de testes com ferramentas para invasão e segurança de sistemas de informação.

2.2 OBJETIVOS ESPECÍFICOS

Para alcançar o objetivo geral, trabalhou-se etapas as quais estão listadas a seguir.

a) levantamento bibliográfico e dos trabalhos da literatura correlata pertinentes ao tema para caracterização do problema;
b) estudo, avaliação e análise de trabalhos relacionados a este projeto;
c) estudo, avaliação e análise de técnicas para invasão de sistemas de informação;
d) estudo, avaliação e análise de ferramentas para defesa contra invasões de sistema de informações;
e) relatório apontando as técnicas mais comuns de invasão, assim como as ferramentas e contramedidas para combater cada caso;

3 JUSTIFICATIVA

Nos últimos anos vivenciamos uma evolução muito rápida da tecnologia e dos meios de comunicação. Novos celulares, computadores e ferramentas cada dia mais tecnológicas estão disponíveis no mercado e são de fácil acesso para toda população. É cada vez mais comum aplicativos de redes sociais, compras online, armazenamento de dados em nuvens etc. desta forma, uma vez que a tecnologia faz parte do nosso cotidiano, não poderíamos destacar a importância de que esses sistemas usado por nós, sejam seguros e transparentes para com o usuário, para assim, evitar problemas como exposição de dados particular em sites de consulta pública que, podem acabar sendo utilizado por pessoas má intencionadas e muitas vezes, causar danos irreversíveis para usuários e empresas.

As tentativas de invasões são constantes e podem acarretar diversos problemas, uma vez que boa parte do valor de uma empresa está nas informações guardadas em seu sistema. Em tempos de crise como a qual estamos passando com a corona vírus e a migração da boa parte do setor de trabalho para o regime de *home office,* a preocupação de que as informações permaneçam seguras em um ambiente externo da empresa, gera uma necessidade de treinamento e preparação por parte dos profissionais que adotaram o sistema, por exemplo.

Dessa forma, esse projeto, propõe uma contribuição com a área de pesquisa de segurança da informação, apontando como se prevenir das principais formas de invasão de sistema de informações, assim como quais ferramentas podem ser utilizadas para cada situação.

4 METODOLOGIA

Este trabalho foi dividido em duas etapas: fundamentação teórica e estudo de técnicas de invasão de sistemas, assim como formas de preveni-los.

Na fundamentação teórica, foram abordadas teorias e ferramentas computacionais necessárias ao desenvolvimento deste projeto. Este levantamento bibliográfico será baseado em consultas à literatura especializada e de alta relevância científica, incluindo: monografias, dissertações, teses, livros, sites de documentação e artigos científicos.

Após concluir a pesquisa bibliográfica, será realizada uma seleção de conteúdos relacionados ao tema do projeto, com o intuito de auxiliar o desenvolvimento da proposta.

Sequencialmente, definiu-se quais técnicas de invasão serão estudadas neste trabalho e suas respectivas formas de prevenção.

Por fim, será elaborou-se um documento apontando as características de cada técnica de invasão, quais são os pontos explorados em sistema de informações para efetuar as invasões. Após, será apontado os métodos de prevenção que devem ser utilizados para cada caso, assim como suas características gerais.

Os resultados serão apresentados no Fórum de Iniciação Científica do UNISAGRADO, bem como, submetido a eventos/revistas científicas da área.

4.1 TESTES E ANÁLISE DE RESULTADOS

Ao final da pesquisa, será possível compreender o funcionamento de técnicas utilizadas para invasão de sistemas de informação assim como a forma de combatê-las durante o estudo, serão avaliados os seguintes aspectos:

a) Vulnerabilidades exploradas por cada técnicas;
b) Formas de prevenção de acordo com cada técnica de invasão
c) Características gerais das técnicas de invasão. Por exemplo: compatível somente com sistemas operacionais baseados em Unix.
d) Características gerais das técnicas usadas para segurança da informação. Por exemplo: técnica voltada para segurança de rede padrão IPV6, compatível com sistema operacional baseado em Unix e Switch CISCO.

Além desses, outros resultados pertinentes poderão ser avaliados conforme o desenvolvimento do projeto.

4.2 REDAÇÃO FINAL E APRESENTAÇÃO DA PESQUISA

Constituinte da documentação relacionada às técnicas para invasão de sistema de informação, o tipo de vulnerabilidade que cada uma explora e as ferramentas para combatê-las. Além de todo o levantamento bibliográfico utilizado, os materiais e métodos empregados para a elaboração do relatório final, os

resultados alcançados, as discussões e considerações sobre as técnicas estudadas e testadas, as referências utilizadas e todos os demais anexos indispensáveis para a reprodução e continuação desta pesquisa.

Por fim, após o término desse projeto de pesquisa, a proposta, os resultados obtidos e o próprio aplicativo serão apresentados no Fórum de Iniciação Científica da UNISAGRADO.

5 RESULTADOS

Esta seção apresenta o estudo feito em ferramentas de segurança e invasão de sistemas online. Até o momento foram analisadas nove ferramentas, sendo utilizadas para ataque ou defesa de invasão em sistemas online.

Após a análise das ferramentas restantes, será gerado uma tabela resumo para ferramentas de invasão e outra para ferramentas de defesa.

5.1 FERRAMENTAS DE INVASÃO DE SISTEMA DE INFORMAÇÕES

Proteger as informações privadas é uma preocupação de qualquer boa empresa e, com a modernização da tecnologia, é cada vez mais comum o uso de sistemas de informações para armazenamento de dados.

Junto do desenvolvimento tecnológico e das possibilidades que ele trouxe, a área de segurança da informação ganhou força. Em um cenário em que, antes, um espião industrial, por exemplo, necessitava entrar fisicamente na empresa - um trabalho muitas vezes complicado - hoje, com a disponibilização dessas informações em sistemas online, permite que invasores munidos de computadores, consigam atacar este sistema e ali coletar estes dados.

Com este panorama definido, é de se esperar que meios para proteger estas informações fossem desenvolvidas. Nesse contexto, aqui encontram-se detalhes referente as ferramentas que serão utilizadas neste projeto.

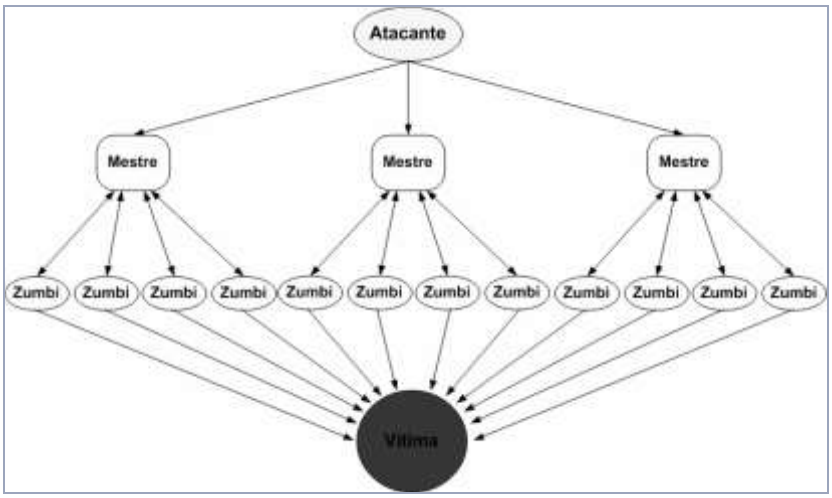
5.1.1 DDoS e Botnet

A princípio, os temas supracitados seriam tratados de maneira separada, porém, após estudo foi notado uma forte relação entre ambos os tópicos, dessa forma a abordagem dos respectivos foi feita de maneira conjunta.

Assim como a Negação de serviço, esta tem como objetivo tornar indisponível um sistema, porém, a mesma ao invés de um ataque com apenas um computador,

utiliza de computadores e outros dispositivos para estar realizando o ataque (MOORE *et al*., 2010). A Figura 2 exemplifica a estrutura de um ataque DDos.

Figura 2 - Estrutura de um ataque DDoS.



Fonte: Research Gate(2015).

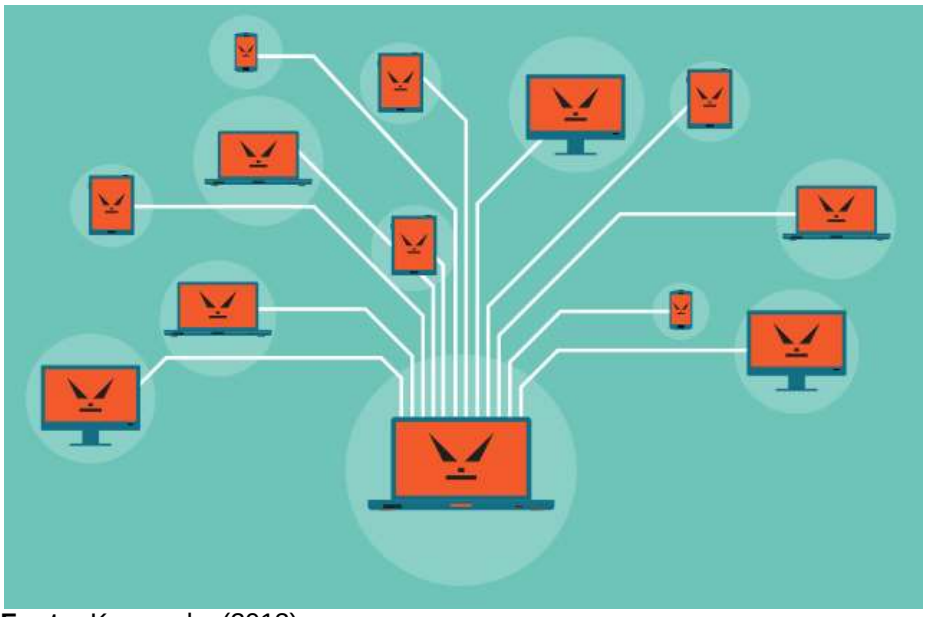
Um dos principais pilares da segurança da informação, supracitado, é a “Disponibilidade”, todo sistema web, tem de responder às solicitações feitas e estar acessível para os usuários legítimos os quais precisam do mesmo.

A negação de serviço depende de o atacante conseguir enviar mais solicitações do que o sistema em si consegue atender, e cada vez mais os sistemas estão mais parrudos e resistentes a este tipo de ataque. Para compensar isto, surge o “Ataque de negação de serviço distribuído”, o invasor utiliza de uma rede de computadores para estarem realizando o ataque, a “BotNet” (MOORE et al., 2010).

A BotNet é uma rede composta por “*bots*”, que são computadores ou qualquer outro dispositivo eletrônico com internet, infectados, os quais provavelmente se infectaram sem nem o usuário saber como, através de um download de um crack possivelmente (MOORE et al., 2010).

Uma vez infectado o computador se torna escravo de um computador mestre, este que é controlado pelo invasor. O principal objetivo do invasor ao capturar este computador, é roubar o poder de processamento da máquina, seja para ataques, ou até mesmo para mineração, por exemplo (MOORE et al., 2010). A Figura 3 exemplifica esse tipo de ataque.

Figura 3 - Exemplo de BotNet



Fonte: Kaspersky (2013).

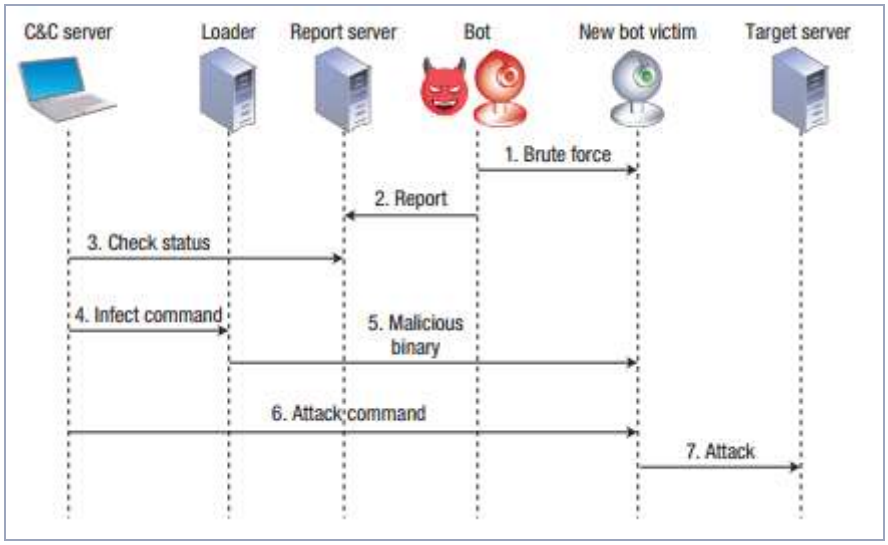
5.1.1.1 Mirai

Nos últimos anos nota-se um aumento da popularidade da IoT, o que gerou uma necessidade de ampliação do seu poder computacional, tornando-a uma plataforma suscetível a invasões sendo o elo mais frágil em questão de segurança nas redes modernas de computadores. São dispositivos que estão constantemente conectados à internet, e seu baixo nível de segurança os tornam alvos fáceis para integrar uma BotNet para DDoS (KOLIAS *et al*., 2017).

O método de infecção que a Mirai utiliza é primeiro buscar por aparelhos que utilizam o BusyBox, geralmente roteadores, em seguida, por meio de força bruta ele tenta acessar como administrador de outros dispositivos na rede e se espalhar.

Segundo Kolias *et al.* (2017) Desde a publicação do seu código fonte, diversas versões deste mesmo Malware surgiram, explorando da mesma forma e ela falha. A Figura 4 traz um esquema de como funciona um ataque utilizando mirai BotNet.

Figura 4 - Esquema do funcionamento da Mirai botnet



Fonte: Kolias *et al*. (2017).

O *bot* é o primeiro dispositivo que irá começar a proliferar o vírus, buscando por dispositivos vulneráveis. O servidor C&C (*command and control*) é aquele que irá prover uma interface de gerenciamento dos dispositivos que foram infectados permitindo um atacante fazer uma gestão dos recursos que possui. O Loader tem como intuito facilitar a infecção de dispositivos de diversas arquiteturas diferentes, e, por último, o servidor de relatórios mantém um banco de dados com as informações dos dispositivos conectados (KOLIAS *et al*., 2017).

Kolias *et al.* (2017), menciona alguns passos relacionados ao processo de ataque da BotNet, são eles:

a) ataque de força bruta: O meio de proliferação da BotNet é através de ataques de força bruta, buscando sempre por dispositivos mal configurados, a busca é feita através de IP públicos e tentando acesso através de portas TCP, geralmente sendo à porta 23.
b) adquirir informações e enviá-las para o servidor de relatórios
c) servidor C&C verifica o *bot* master e possíveis novas vítimas se comunicando com o servidor de relatórios (geralmente via rede TOR).
d) decidir quais dispositivos atacar, e enviar instruções pelo *Loader*, como IP e arquitetura do hardware das vítimas.
e) após infectado a vítima começa a receber instruções para instalar o arquivo do malware, em seguida, o mesmo pode ser controlado pelo C&C e passa a receber instruções para o ataque.

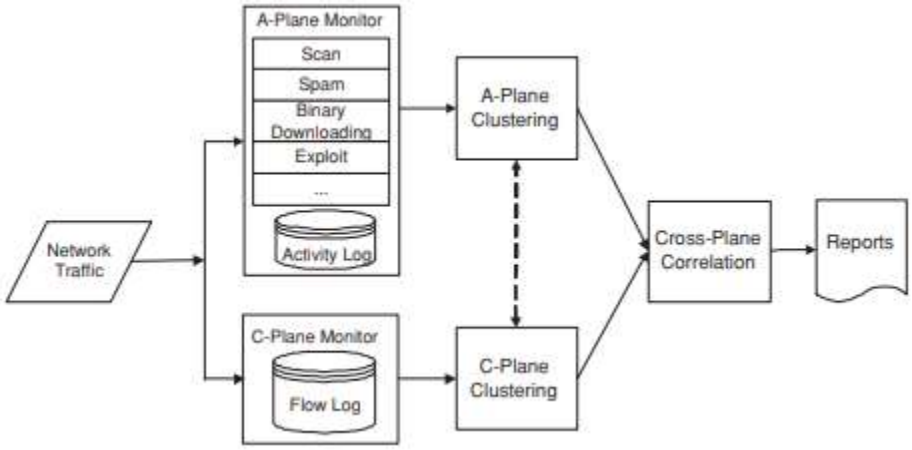
Dispositivos de IoT tendem a se tornar obsoletos muito rapidamente, deixando de receber atualizações das fabricantes, porém, ainda possuem uma capacidade computacional relevante, desta forma se tornam vítimas fáceis para hackers. Este tipo de problema já havia sido

5.1.1.2 BotMiner

GuofeiI *et al*., (2008) A utilização de *botnets* para ataques está cada vez mais comum e mais perigosa. Apesar da variedade a estrutura na maior parte das *botnets* é a mesma, cada *bot* membro de uma *botnet* depende de um servidor central para enviar comandos para ele.

Apesar da simplicidade da estrutura o problema está em como detectar, uma *botnet* pode facilmente detectar a forma que utiliza para um ataque, se adaptando a defesa do sistema. A utilização de um BotMiner para detecção de *botnets* consiste na monitoria do tráfego de internet à procura de pacotes suspeitos enviados de possíveis nós de uma *botnet*. Esta abordagem tem uma vantagem, não depende de o atacante utilizar um tipo de ataque ou estrutura para organização da *botnet*, tem uma baixa taxa de falhas e graças a quantia de dados disponíveis permitindo um algoritmo acurado. A monitoria da rede tem o objetivo de gerar padrões a serem utilizados, existem duas estruturas para monitoramento, a primeira procura pelas atividades maliciosas e a outra registra os logs de conexão com as informações dos computadores que são suspeitos de estarem comprometidos (GUOFEI *et al*., 2008). A Figura 5 exemplifica a arquitetura do BotMiner

Figura 5 - Arquitetura do BotMiner



Fonte: Guoefei *et al*., (2008)

5.1.1.3 BotGraph

Ataques de DDoS sempre foram problemas significativos, devido à complexidade para se detectar e se defender. Apesar da enorme pesquisa na área este tipo de ataque ainda é capaz de ser um empecilho muito grande para aplicações (YAO *et al*., 2009).

Segundo Yao *et al*. (2009) a proposta do BotGraph, é analisar padrões de ataques utilizando grandes provedores de e-mail como meio de spam. Em muitos casos, analisar um elemento em particular não traz informações relevantes e o suficiente para detectar um padrão, e muitas vezes passa despercebido, porém, ao analisar em massa os e-mails, é possível detectar padrões que separados seriam irrelevantes. O primeiro passo é diferenciar comportamentos de *bots* a comportamento de usuários comuns e depois, entender como lidar com um grande volume de informação.

Meios de detecção de *bots* de e-mail:

a) Detecção através de cadastros: Múltiplos cadastros de um mesmo IP em um curto período, mesmo utilizando um proxy, mantém um padrão, um aumento incomum pode ser motivo de suspeita
b) Utilização de credenciais parecidas: Por facilidade, uma *botnet* pode utilizar de endereços de e-mail parecidos ou ter o mesmo IP
    I. Utilização do mesmo IP
    II. *Bots* de uma *botnet* alternando o IP entre si

O meio de detecção utiliza de artifícios como os supracitados para identificar, são comportamentos dificilmente adotados por usuários comuns. O Quadro 1 sintetiza os resultados dos autores Yao *et al*. (2009).

Quadro 1 - Tabela com resultados do experimento

Resultado dos Testes			
Tentativas de Login	Com Sucesso	Sem sucesso	Total
	9866	413362	423228
Fontes de IP	NÃO SE APLICA	NÃO SE APLICA	5297
Comandos executados	NÃO SE APLICA	NÃO SE APLICA	31328
Downloads	NÃO SE APLICA	NÃO SE APLICA	5368

Fonte: Elaborado pelo autor

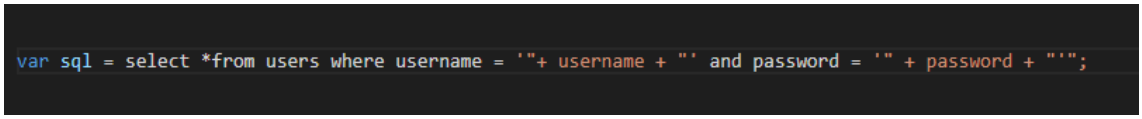
O Quadro foi elaborada com o intuito de representar visualmente dados contidos no artigo supracitado.

5.1.2 SQL Injection

O objetivo deste método de ataque é manipular os dados de uma consulta dentro do banco de dados do sistema.

Este problema é decorrente principalmente de problemas de projeto, muitos códigos ficam de fácil acesso para o usuário, permitindo ao mesmo manipular o que será enviado dentro da instrução de Linguagem de Consulta Estruturada (SQL) para fazer uma consulta no banco por exemplo. Um usuário que consiga realizar este tipo de manipulação nas instruções que serão enviadas para o banco de dados pode, por exemplo se autenticar como outro usuário. A Figura 6 exemplifica uma consulta SQL utilizando Node.js

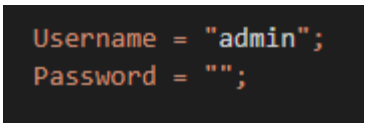
Figura 6 - Frase SQL utilizada para realizar consultas



Fonte: Elaborada pelo autor.

No exemplo acima, o código não realiza nenhum tipo de validação, permitindo ao usuário burlar o sistema e por exemplo, realizar o login sem que seja necessário digitar a senha correta para o usuário em questão, alterando os campos por exemplo, conforme a figura a seguir, passando para o sistema o nome de usuário, como pode-se observar na Figura 7.

Figura 7 - Exemplo de forma de burlar o sistema



Fonte: Elaborada pelo autor.

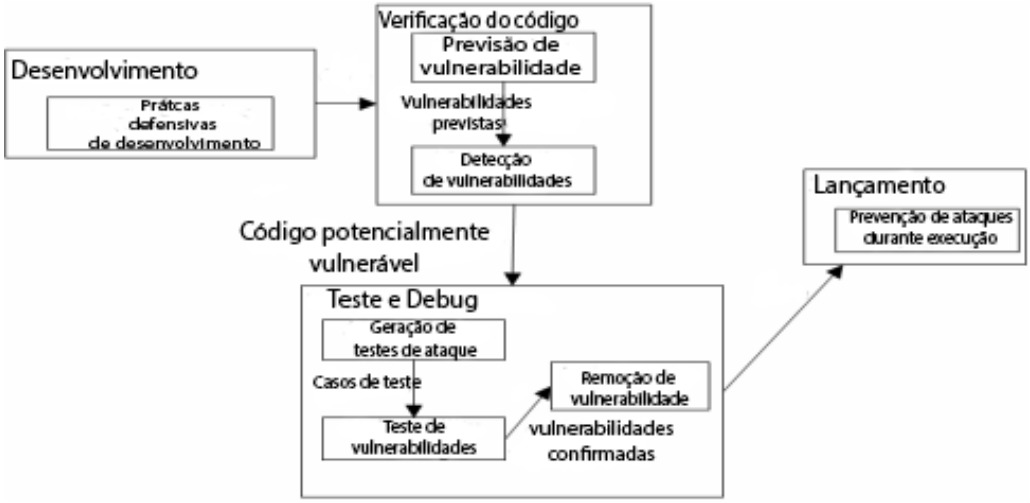
Previsto, e com o avanço das tecnologias e dos hardwares só tende a piorar, uma vez que ainda é difícil de perceber as invasões e não há tanto investimento em questão de segurança para os mesmos dispositivos (Kolias *et al*., 2017).

5.1.2.1 Derrotando o SQL INJECTION

Os ataques utilizando de SQL *Injection* em grande maioria são resultados resultantes da falta de experiência e malícia do desenvolvedor. Quando construído uma aplicação que utiliza de SQL é comum a construção de *queries* para consultas e afins. SHAR (2012).

Um banco de dados não devidamente protegido pode facilmente entregar dados como, nome de tabelas, campos e valores da mesma o que são informações valiosas quando se tenta invadir um sistema através dessa metodologia. Os ataques de *SQL Injection* utilizando de tautologia por exemplo, consistem em fazer consultas onde é quase que garantido que o resultado dela é positivo, por exemplo a consulta com uma cláusula" WHERE 1 = 1" retornará como positivo. SHAR (2012). A Figura 9 ilustra um passo a passo das verificações que devem ser feitas para garantir que um software seja o mais integro o possível.

Figura 8 - Planejamento de um código com boas práticas



Fonte: SHAR (2012).

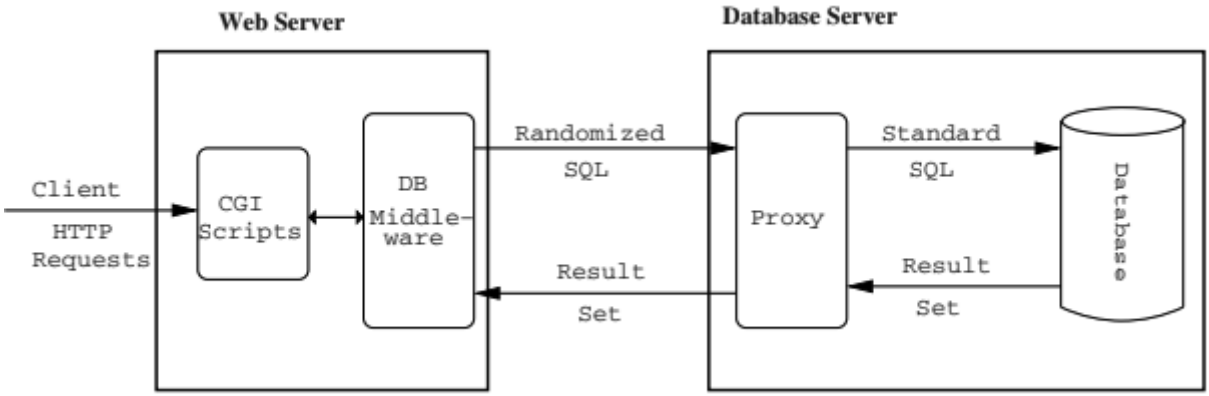
Segundo SHAR (2012) o melhor caminho para se prevenir dos ataques de SQL *Injection* realmente é uma programação defensiva buscando prever possíveis casos de vulnerabilidades e reforçar os testes e verificações.

5.1.2.2 SQLRand

Esta implementação de defesa se baseia em automatizar testes com as consultas recebidas via *Front-End* e verificar se são válidas antes de enviá-las ao servidor. A estratégia baseia-se na criação de uma coleção de casos de consulta

que são imprevisíveis pelo atacante (BOYD, 2004). A Figura 9 ilustra a estrutura do método prático implementado como defesa.

Figura 9 - Estrutura do SQLRand



Fonte: BOYD (2004).

O primeiro passo é receber uma *request* via *front* e ela será randomizada pelo proxy, caso seja válido será traduzida e enviada para o banco de dados que retornará corretamente a consulta, caso inválida, a consulta não chega ao banco de dados principal ficando presa no intermediário que retorna um erro genérico para evitar ceder informações ao atacante (BOYD, 2004).

5.1.2.3 Formas de SQL Injection e suas contramedidas

Ataques de *SQL Injection* consistem em buscar falhas nas consultas realizadas e injetar código malicioso para extrair informações, porém, aprofundando mais é possível notar que existem várias brechas que podem ser utilizadas que retornam informações diferentes que são vantajosas em determinadas situações (HALFOND, 2006). Os métodos de invasão mais comuns:

a) tautologia: método que tenta transformar a consulta em uma tautologia para permitir a autenticação do invasor.
b) consultas ilegais: é utilizado para extrair informações de erro do sistema e permitir o invasor a mapear e traçar um plano para continuar a invasão
c) *union* Query: utilização do método Union para mostrar dados de uma tabela diferente da originalmente planejada pelo desenvolvedor

Contramedidas adotadas:

a) SQLRand: criação de instruções SQL randômicas para permitir conferir se são válidas com os modelos pré-moldados pelo desenvolvedor.
b) verificadores de código estático: esse tipo de prevenção baseia-se na verificação dos tipos inseridos na consulta e se deveriam ser válidos ao sistema, não foram projetados para serem usados contra-ataques, porém por serem uma boa prática de desenvolvimento é uma eficaz maneira de prevenir ataques.
c) teste de Caixa preta: assim como uma caixa preta de um avião o objetivo dessa metodologia é realizar todos os testes e verificações no sistema e armazenar elas, permitindo uma auditoria do sistema, porém essa metodologia só é eficaz junto a outras formas de defesa.

5.2 FERRAMENTAS DE DEFESA CONTRA INVASÕES DE SISTEMA DE INFORMAÇÕES

Existem vários procedimentos que tornam um sistema mais seguro, aqui serão apresentados algumas táticas e ferramentas para a proteção de um sistema.

5.2.1 Honeypot

Segundo Assunção (2009), o *Honeypot* (pote de mel, em tradução literal), trata-se de uma armadilha. Esse tem como objetivo conter o ataque e manter o atacante restringido àquela parte, para coletar seus dados e evitar que o sistema seja violado. Esta ferramenta funciona simulando um ambiente no qual o objetivo é monitorar os passos do invasor, e gerar um relatório com tudo que foi feito pelo mesmo, dessa forma podendo aprimorar o sistema em questão.

Para Assunção (2009) o intuito do *honeypot* é manter o invasor fora, porém, ainda sim a implantação de um é perigosa, dependendo do nível de interatividade de um *honeypot*, o mesmo pode ser utilizado contra a própria rede da qual faz parte. Um *honeyPot* de baixa interação, conta apenas com alguns serviços simples, simulados e não reais não dando acesso ao invasor ao sistema de verdade, porém estes são facilmente detectáveis e o invasor simplesmente perde o interesse. Em casos de alta interação, são expostos todos os serviços que são encontrados dentro do sistema real, e caso o invasor consiga passar pelo *honeypot* ele terá acesso a

uma máquina da qual faz parte da rede. Geralmente dado o alto nível de realismo da armadilha, poucos são os casos que o invasor vai perceber, porém, ainda sim, há seus riscos.

5.2.1.1 Prevenção de detecção de HoneyPots

Nas últimas duas décadas os *honeyPots* têm sido utilizados em massa como mecanismo de proteção de sistemas. A ideia é atrair o atacante e por meio das interações do mesmo com o *honeypot*, enquanto ele interage, o sistema estará armazenando as informações do atacante que podem ser utilizadas para rastrear o mesmo e ao mesmo tempo levantar uma lista de possíveis vulnerabilidades (TSIKERDEKIS et al, 2018).

a) *honeypot* de pesquisa: Utilizado para adquirir informações do atacante, geralmente posicionados dentro de uma DMZ, perto de recursos de segurança, assim alertando o usuário de uma possível invasão, coletando-as e a armazenando.
b) *honeypot* de Produção: Utilizado apenas como meio de distração para o atacante.

Em relação ao seu modelo de detecção, a técnica utiliza utilização de *honeypot* depende totalmente da capacidade dele se passar por um real sistema, copiando e replicando características originais do mesmo e passar uma maior credibilidade distraindo e mantendo o atacante no *honeypot*. A maior vulnerabilidade de um h*oneypot* em quesito de detecção é a sua limitação de recursos. Geralmente o atacante vai testar o servidor e medir a quantidade de interações que ocorreram dentro do mesmo, e na maior parte dos casos o nível de interação é baixo comparado a de sistemas reais. Além das limitações de recursos, tanto financeiros, quanto de desenvolvimento, há também implicações legais que impedem que alguns recursos sejam adicionados ao *honeypot* para dar maior credibilidade ao mesmo. Geralmente dada essas limitações legais, os h*oneypots* são descobertos por meio de testes que não são possíveis uma vez que dada a legislação os recursos são bloqueados. As técnicas de invasão têm progredido rapidamente, porém, o nível de evolução dos h*oneypots* não está sendo o suficiente para acompanhar, já há meios e hardware o suficiente para isto, porém, falta um maior planejamento para projetar um ecossistema mais realista, o suficiente para prender um atacante mais experiente (TSIKERDEKIS et al, 2018).

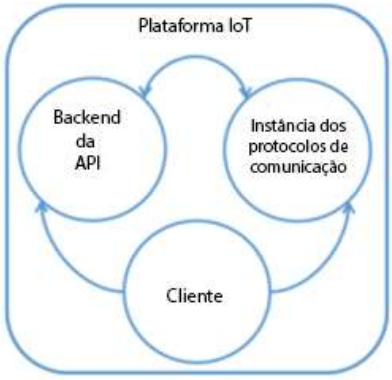
5.2.1.2 ThingPot

A *Internet of Things* (IoT) está se popularizando ainda mais, porém este tipo de tecnologia sempre teve uma limitação de hardware, porém, com o passar do tempo o hardware vem ficando mais forte, desta forma os riscos de um ataque utilizando os mesmos, se tornaram grandes, tendo como exemplo a Mirai, uma poderosa *botnet* que utiliza destes dispositivos para fazer ataques (WANG, 2018).

Dispositivos IoT utilizam vários tipos de protocolos, XMPP, CoAP e Http, por exemplo. Com estes protocolos é possível atender várias finalidades, mas principalmente para comunicação, seja ela em tempo real ou assíncrona (Wang, 2018).

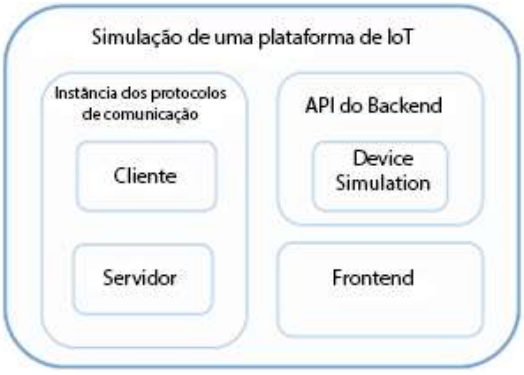
Segundo Wang (2018) A ideia do ThingPot é criar um sistema que simule um dispositivo IoT, mas não se limitando a camada de comunicação. A Figura 10 traz uma representação visual de uma estrutura básica de IoT. Já a Figura 11 exibe um esquema visual de uma simulação de *honeypot* para IoT.

Figura 10 - Representação da estrutura de um dispositivo IoT



Fonte: Wang (2018).

Figura 11 - Representação da simulação de um Honeypot



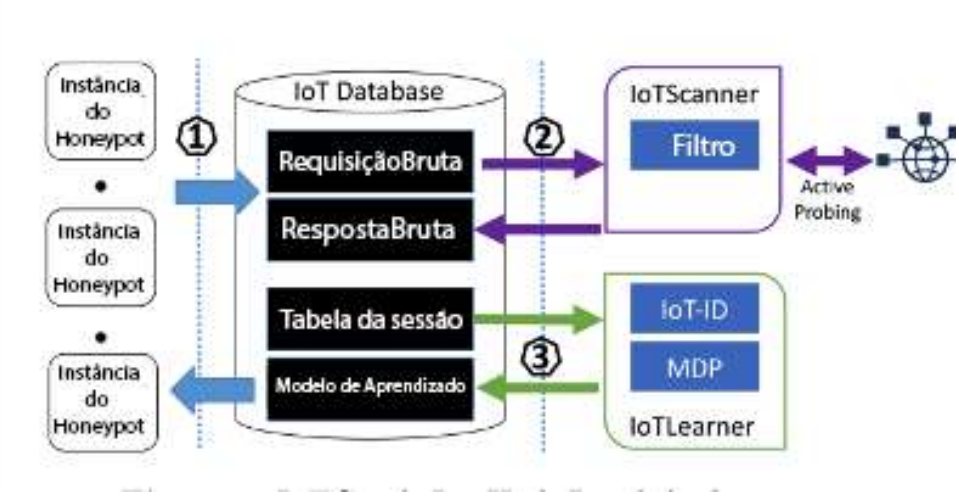
Fonte: Wang (2018).

5.2.1.3 CandyJar

Nos últimos anos é possível observar uma crescente na popularização do interesse em IoT, o que nos leva a uma necessidade de discussão quanto a segurança desses dispositivos com sistemas embarcados. Há uma alta gama de dispositivos de IoT, desta forma criar *honeypots* de baixa interação, além de não garantir tantas informações dada a sua natureza, é inviável devido ao fato de ser um trabalho manual de construção do mesmo, e criar *honeypots* de alta interação seria altamente custoso e financeiramente inviável (LUO, 2018).

A proposta do *CandyJar Honeypot* é abranger tanto os dados adquiridos por h*oneypots* de baixa interação quanto os de alta interação, porém sem correr risco do mesmo ser comprometido. Ele faz uso de *honeypot* de interação inteligente. Este tipo de *honeypot* tem como objetivo garantir uma fiel interação, aprendendo exatamente qual a resposta que uma interação com o dispositivo resultará utilizando os dados de usuários sem conhecimento acerca de dispositivos de IoT. A coleta desses dados de interações verídicas de usuários ajudará o algoritmo a identificar as conexões que são de fato válidas e quais estão tentando se aproveitar de alguma falha do dispositivo (LUO, 2018). A Figura 12 traz a ilustração desse esquema.

Figura 12 - IoT Candy Jar de alta interação



Fonte: Black Hat (2017).

Explicação da estrutura:

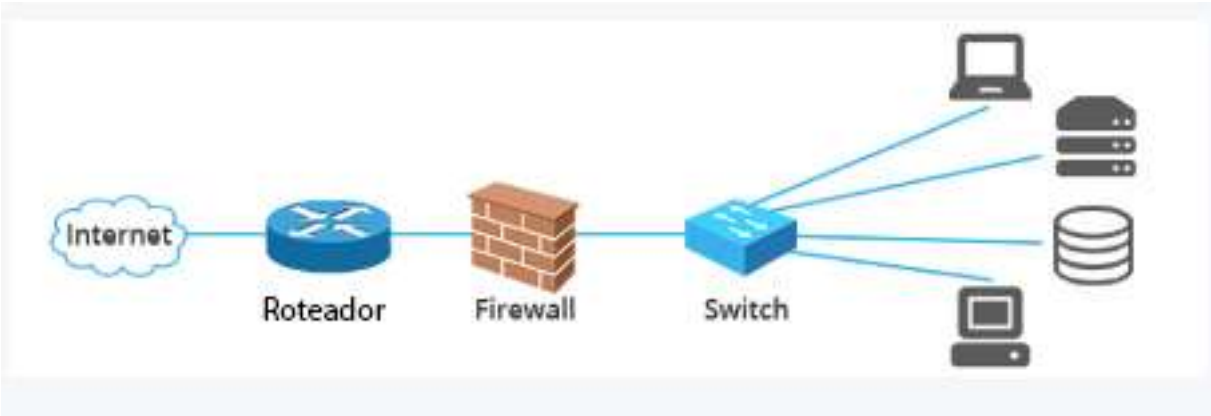
a) IoT Oracle: é um banco de dados nos quais ficaram salvo as informações adquiridas pelo dispositivo.

b) *Honeypot* Module: são as instâncias que estão rodando em servidores para atrair atacantes através de interações e instigá-los a explorar o sistema, e eventualmente sincronizar os dados com a base de dados.
c) IoTScanner: analisa as requisições feitas e procura por dispositivos que possam responder de acordo com o que foi recebido.
d) IoTLearner: o CandyJar Honeypot começa como um sistema de baixa interatividade, o que permite ele a evoluir é este módulo, uma vez que ele utiliza de um algoritmo de aprendizagem para se aprimorar e otimizar o modelo de respostas utilizado com os invasores, junto dos conhecimentos pré-adquiridos pelas interações com usuários comuns.

5.2.2.1 *Firewall*

Firewall é um equipamento dedicado ou software, cuja função é analisar todas as requisições que passam pelo mesmo antes de entrarem na rede, como pode-se observar na Figura 13.

Figura 13 - Exemplo de estrutura de ambiente com Firewall



Fonte: IPERIUS BACKUP (2020).

Existem dois tipos de *Firewall,* o filtro de pacotes e o proxy firewalls:

a) filtro de pacotes: Este é o mais simples, analisa apenas o cabeçalho dos pacotes que passam pela camada de rede e transporte, analisando o IP de destino, IP de origem, porta de origem, porta de destino e qual o tipo de protocolo da requisição (TCP ou UDP), ignorando completamente o conteúdo da aplicação.

b) proxy: Este serve como um intermediador para qualquer conexão que um usuário possa estar realizando para um servidor web. O usuário manda uma solicitação para o proxy, este que encaminhará a solicitação para o servidor, podendo retornar tanto uma resposta permissiva ou uma negativa para o usuário, determinando se o usuário conseguirá acessar o serviço. Este pode ser tanto utilizado para barrar conexões externas quanto impedir que usuários internos de uma rede acessem conteúdos os quais não estejam disponíveis dentro das regras configuradas.

5.2.2.1 Segurança em dispositivos de IoT utilizando grafos

Segundo os autores George e Thampi (2018) a segurança na IoT é algo a se levar em conta uma vez que ela possui diversas limitações de implementação, principalmente na questão de segurança. Geralmente dispositivos de IoT ficam distribuídos na rede, e com o aumento do poder de processamento deles, e dessa forma é possível encarar como um sistema distribuído de computação.

Um dispositivo de IoT está constantemente analisando dados do seu usuário, inclusive dados importantes para a segurança do dispositivo, dessa forma aprimorando o algoritmo dele. Alguns dispositivos como *SmartLocks* representam riscos reais para a integridade do usuário caso uma falha seja explorada. As políticas de segurança destes dispositivos são baseadas em políticas PDP, porém essa política depende de os desenvolvedores implementarem-nas corretamente para garantir a integridade do dispositivo. Uma solução para estas falhas é implementar monitores de tráfego em cada nó de IoT. A rede será monitorada capturando e analisa todo pacote de comunicação realizando a DPI (Deep Packet Inspection), procurando por anomalias e conferindo os dados do pacote, emissor, metadados, tipo de requisição e tipo de serviço, por exemplo (GEORGE; THAMPI, 2018).

5.2.2.2 Firewall for Internet of Things

Os dispositivos de IoT em sua grande maioria estão diretamente conectados em uma rede doméstica, assim como os dispositivos aos quais estão relacionados.

Este tipo de aparelho possui pouca capacidade de processamento e não suporta grandes métodos de encriptação reduzindo a sua segurança (GUPTA, 2017).

Segundo Gupta (2017) Estudos na área mostraram que a utilização de soluções comerciais no geral é muito cara para o perfil de uso desta aplicação, como alternativa, foi utilizado um Raspberry Pi o qual estava com um firewall configurado, monitorando todos os tipos de dados que passam por ele.

Configurações do firewall do Raspberry Pi:

a) máscara de IPv4 – Permite o Raspberry atuar como um Gateway de origem- NAT para pacotes de saída
b) proteção contra Spoof – filtrar os endereços que são roteáveis, descartando os que não são.
c) bloqueio de redirecionamento ICMP – Previne ataques MITM.
d) cookies SYN TCP/IP – Previne ataques de DOS do tipo SYN.
e) rastreamento de conexões através de *iptables*.

Outra implementação discutida é a criação de uma *whitelist*, permitindo apenas que dispositivos previamente autorizados tenham acesso ao gateway.

5.2.3 Scanners de vulnerabilidade

Em um sistema é comum ter vários serviços rodando, nestes casos se ele não for bem configurado podem gerar vulnerabilidades permitindo que invasores se conectem de maneira clandestina a esta porta.

Existem ferramentas que fazem uma varredura procurando por vulnerabilidades, estes mecanismos podem tanto ser utilizados tanto para ataques quanto para defesa do próprio sistema. Por exemplo o Nmap, uma ferramenta para teste de portas de conexão, esta ferramenta manda requisições para todas as portas de 0 a 65535, lembrando que a porta 0 é reservada e, quando receber uma resposta, ele classifica a porta como aberta.

Uma vez descoberta a porta de serviço que se encontra aberta, podemos utilizar outras ferramentas como o OpenVas para rodar alguns scripts de ataque com base na vulnerabilidade achada ao encontrar o sistema.

6 DISCUSÃO DOS RESULTADOS

A fim de compilar as informações avaliadas em cada técnica, elaborou-se um quadro, de forma que facilite a visualização das ferramentas analisadas nesse trabalho juntamente ao método de invasão que elas tentam prevenir, conforme observa-se no Quadro 2.

Quadro 2 - Resumo das abordagens estudadas em cada artigo

ASSUNTO	DESCRIÇÃO	METODOLOGIA
DDOS	Crescimento constante da IoT e a criação de BotNets para ataques de DDOS	Criação de uma BotNet através de ataques de força bruta para invadir dispositivos frágeis e os demais aparelhos existentes na mesma rede ou que se comuniquem de alguma forma com eles.
DDOS	Análise comportamental de usuários na rede para criar filtros que previnam ataques de DDOS	Análise da origem dos pacotes em massa, buscando um padrão; diferenciação de comportamentos de *bots* a usuários e a compreensão de como lidar com um grande volume de informações.
SQL Injection	Explicação do funcionamento dos ataques a base de SQL	Explicação sobre a facilidade de se entregar dados pessoais importantes com o uso de SQL, quando não possuída a devida segurança de sistema.
SQL Injection	Defesa beseada na automatização dos testes e de sua validez antes de serem enviadas ao servidor	Criar uma coleção de casos de consulta que serão imprevisíveis ao atacante, uma avaliação do próprio sistema, e, posteriormente, se aprovado, sua tradução e envio ao banco de dados.
SQL Injection	Explicitação sobre o modo de ataque através de SQL Injection	Citação sobre os meios mais comumente usados de ataques de SQL e os modos de proteção do sistema e de suas informações.
HoneyPot	Sistema de proteção	Atua como uma armadilha, onde protege e

	que contém o ataque e o invasor.	restringe o atacante a uma simulação de ambiente, promovendo o pensamento de que o invasor concluiu o ataque com sucesso.
HoneyPot	Explicação sobre a fácil detecção de HoneyPots em relação a um invasor experiente	Há duas variações de HoneyPots, ambas atuando em conjunto para proteção, porém não são capazes de acompanhar a evolução constante de métodos utilizados para ataques, tornando-os vulneráveis invasores avançados.
HoneyPot	Com as limitações de hardware encontradas na IoT, idealiza-se um novo sistema	Devido à grande limitação sobre a *Internet of Things*, a ideia de criar-se um sistema veio à tona: ThingPot, o qual simularia um dispositivo IoT, porém, menos limitado.
HoneyPot	Apresentação de proposta para um novo meio de defesa melhor e mais barato	CandyJar é um projeto de sistema de proteção, conteria, então, algoritmos de aprendizagem e aprimoramento, trazendo consigo melhores defesas e chances baixas de uma invasão.
Firewall	Definição e explicação sobre o funcionamento e utilizações do Firewall	Descrição da execução do programa, ilustração de seus tipos, diferenças e funcionalidades.
Firewall	Solução a segurança debilitada da IoT em relação a sistemas com informações valiosas	Com a constante análise de diversos dados, a IoT torna-se não confiável, devido a facilidade de um ataque, portanto uma resolução para esse problema seria implementar monitoramento que capturasse e analisando qualquer pacote.
Firewall	Teste de sistema de proteção utilizando Firewall em aparelhos	A maior parte de aparelhos IoT possuem grandes déficits em relação ao processamento, acarretando fraca

	com segurança debilitada	segurança. Testes em Raspberry Pi demonstraram evolução positiva a sua segurança com as configurações Firewall.

Fonte: Elaborada pelo autor.

7 CONSIDERAÇÕES FINAIS

No contexto de Segurança da Informação, tem-se inúmeras abordagens, tecnologia, perspectivas e possibilidade em relação ao modelo de defesa ou invasão. O trabalho buscou contribuir avaliando propostas e descrevendo suas vantagens e desvantagens. No Quadro 1 resume-se as informações apresentadas ao longo do capítulo 5, a ideia é sintetizar para o leitor, e caso ele queira mais informações leia o tópico relacionado no trabalho. Para trabalhos futuros, seria interessante efetuar testes com cada uma das metodologias, e utilizar métricas para medir sua eficácia em distintos ambientes computacionais.

8 ORÇAMENTO

Descrição	DISPONÍVEL		
	Qtde	Valor Unit. (R$)	Valor Total (R$)
Computador pessoal (PC) do pesquisador, notebook com sistema operacional Windows 10 pro – 64 bits, com processador i5 8250u 8 GB de memória RAM	1	R$ 2.600,00	R$ 2.600,00
Sistema Operacional Linux e Windows		-	-
Ferramentas para ataque e defesa de sistemas de informação		-	-
TOTAL			R$ 2.600,00

REFERÊNCIAS

ANGRISHI, Kishore. Turning internet of things (iot) into internet of vulnerabilities (iov): Iot botnets. **arXiv preprint arXiv:1702.03681**, 2017.

Approaches for preventing Honeypot Detection and Compromise /Michail Tsikerdekis / Sherali Zeadally / Sherali Zeadally / Nicolas Sklavos

ASSUNÇÃO, Marcos Flávio Araújo. **Honeypots e Honeynets**. Marcos Flávio Araújo Assunção, 2009.

BOYD, Stephen W.; KEROMYTIS, Angelos D. SQLrand: Preventing SQL injection attacks. In: **International conference on applied cryptography and network security**. Springer, Berlin, Heidelberg, 2004. p. 292-302.

GU, Guofei et al. Botminer: Clustering analysis of network traffic for protocol-and structure-independent botnet detection. 2008.

GUPTA, Naman; NAIK, Vinayak; SENGUPTA, Srishti. A firewall for Internet of Things. In: **2017 9th International Conference on Communication Systems and Networks (COMSNETS)**. IEEE, 2017. p. 411-412.

HALFOND, William G. et al. A classification of SQL-injection attacks and countermeasures. In: **Proceedings of the IEEE international symposium on secure software engineering**. IEEE, 2006. p. 13-15.

IPERIUS BACKUP. Disponível em <https://www.iperiusbackup.net/pt-br/entendendo-a-diferenca-entre-switch-roteador-firewall/ >. Acessado em: 25 mar. 2020.

LYRA, Maurício Rocha. Segurança e auditoria em sistemas de informação. **Rio de Janeiro: Ciência Moderna**, 2008.

LUO, Tongbo et al. Iotcandyjar: Towards an intelligent-interaction honeypot for iot devices. **Black Hat**, p. 1-11, 2017.

MOORE, David et al. Inferring internet denial-of-service activity. **ACM Transactions on Computer Systems (TOCS)**, v. 24, n. 2, p. 115-139, 2006.

PEREIRA, Maria José Lara de Bretãs; FONSECA, João Gabriel Marques. Faces da Decisão: as mudanças de paradigmas e o poder da decisão. São Paulo: Makron Books, 1997.

POLLONI, ENRICO GIULIO FRANCO; FEDELI, RICARDO DANIEL. **Introdução à ciência da computação**. Cengage Learning Editores, 2010.

RESEARCH GATE. Disponível em < https://www.researchgate.net/>. Acessado em: 18 mar 2020.

SHAR, Lwin Khin; TAN, Hee Beng Kuan. Defeating SQL injection. **Computer**, v. 46, n. 3, p. 69-77, 2012

STAIR, Ralph M. Princípios de sistemas de informação. Rio de Janeiro: LTC, 1998.

TSIKERDEKIS, Michail et al. Approaches for preventing honeypot detection and compromise. In: **2018 Global Information Infrastructure and Networking Symposium (GIIS)**. IEEE, 2018. p. 1-6.

WANG, Meng; SANTILLAN, Javier; KUIPERS, Fernando. ThingPot: an interactive Internet-of-Things honeypot. **arXiv preprint arXiv:1807.04114**, 2018.

ZHAO, Yao et al. BotGraph: Large Scale Spamming Botnet Detection. In: **NSDI**. 2009. p. 321-334.

ANEXO I – CARTA DE DISPENSA DE APRESENTAÇÃO AO CEP OU CEUA

UNISAGRADO
Ensino Superior de Excelência

CARTA DE DISPENSA DE APRESENTAÇÃO AO CEP OU CEUA

À

COORDENADORIA DO PROGRAMA DE INICIAÇÃO CIENTÍFICA DA UNISAGRADO

Informo que não é necessária a submissão do projeto de pesquisa intitulado ESTUDO DE TÉCNICAS DE INVASÃO E SEGURANÇA DE SISTEMAS DE INFORMAÇÃO ONLINE, ao Comitê de Ética em Pesquisa com Seres Humanos (CEP) ou à Comissão de Ética no Uso de Animais (CEUA) devido à pesquisa não envolver seres humanos nem animais, pois só utilizará métodos de computação consagrados na literatura, programação/simulação e acesso a dados públicos da internet.

Atenciosamente,

Vinicius Santos Andrade

Vinicius Santos Andrade

Bauru, 25 de março de 2020.