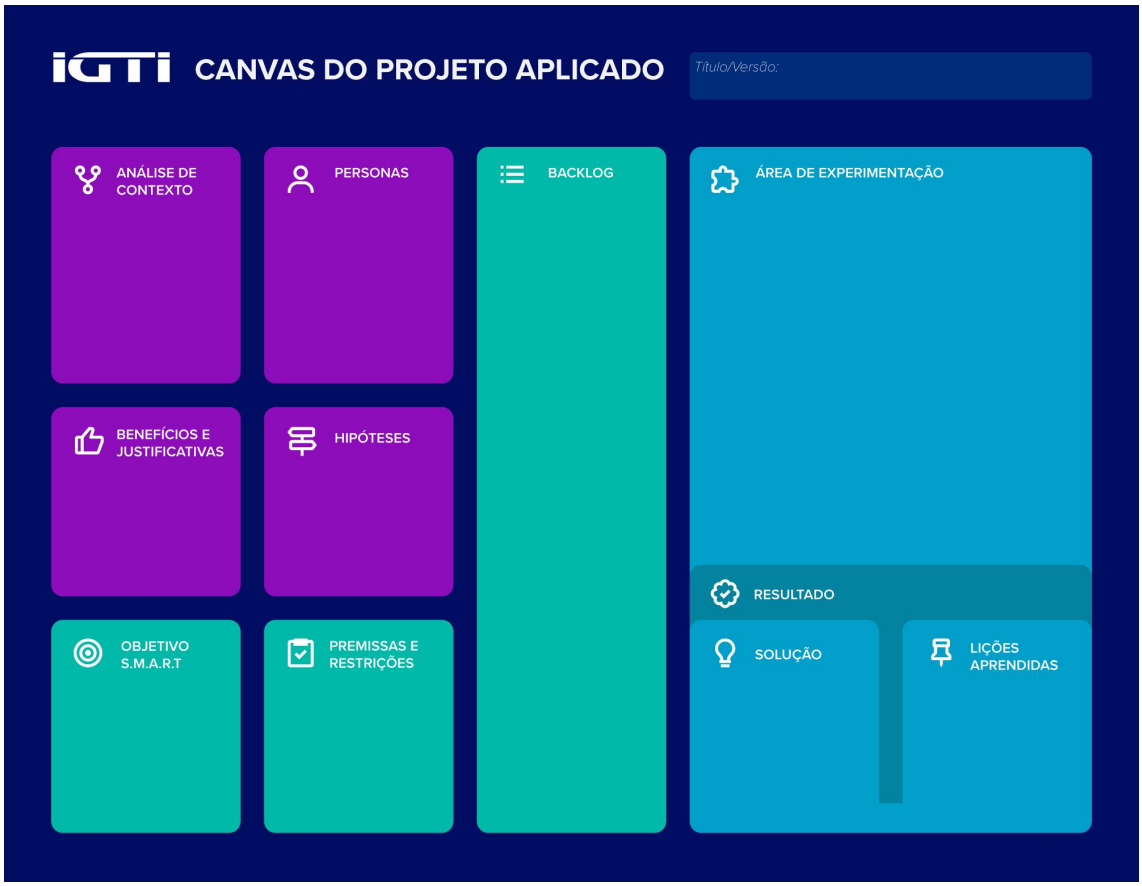
Relatório do Projeto Aplicado

Nome	Júnior André Marostega
Título	MSIS – Um *software* aplicável a verificação de informações de portas e versões sistêmicas de dispositivos de Internet das Coisas (IoT) utilizando redes *Ethernet* e Wi-Fi.
Curso	Pós Graduação em Segurança Cibernética
Orientador(a)	Tiago Rubia Martins
Data	07/11/2021

Sumário

1. CANVAS do Projeto Aplicado

Figura conceitual, que representa todas as etapas do Projeto Aplicado.



1.1 Desafio

1.1.1 Análise de Contexto

Introdução

Com a evolução e praticidade das tecnologias de *Internet of Thing* (IoT) é visível a aderência de várias áreas na adoção deste modelo de tecnologia. IoT é uma nomenclatura que foi usada pela primeira vez por Kevin Ashton no ano de 1998 (WEBER, 2009). Este é um conceito que visualiza um paradigma de conexão, definido por (GUBBI et al., 2013) é a comunicação entre um ou mais dispositivos que se localizam em sua volta. IoT é uma rede que possui diversos dispositivos conectados entre si para realizar a comunicação e transferência de dados. Os complementos desta rede podem ser definidos com *hardwares* e *softwares*, ambos realizando a troca de informações de forma automática (MIORANDI et., 2012).

Com esta enorme capacidade que este conceito aborda, fica cada vez mais perceptível que o mercado está correndo para fins de usufruir e tornar seus ambientes cada vez mais controlados. Um dos pilares que por muitos não são observados e muito menos ponderados é na segurança seja na comunicação ou nas configurações que cada dispositivo possuí. Assim fazendo com que pontos de vulnerabilidades nascem em projetos quais estes aspectos não são tratados.

Assim o desafio deste projeto aplicado é no desenvolvimento do MSIS. O MSIS é um *software* desenvolvido para auxiliar na gestão de uma rede de comunicação onde existem dispositivos de IoT atuando. O objetivo é fazer a captura de informações básicas, como versão de *firmware* do sistema e também identificar se existem portas abertas nos devices. O MSIS poderá ser usado em redes corporativas e também em redes *home office,* ou seja, redes residências. Sua arquitetura pode ser executada em dispositivos como computadores, mas também em dispositivos de IoT, como exemplo *Raspberry* PI.

As métricas para a conclusão deste projeto aplicado se orientam na execução e testes em uma rede *home office* onde existem vários dispositivos atuando nesta rede. Como este objetivo será específico na execução de uma rede doméstica, o escopo do planejamento será dado através das seguintes colocações:

a) Existem portas abertas nos dispositivos conectados a rede? Quais são elas?

b) Quais são as informações de versionamento e modelos dos *firmwares* em execução nestes dispositivos de IoT em execução na rede?

Na busca de facilitar a compreensão e também o problema será apresentado abaixo a matriz CSD, apresentado pela Figura 1. Esta matriz trabalha com Certezas, Suposições e Dúvidas, o objetivo de aplicar esta matriz é compreensão de pontos chaves que esclarecem mais a ideia do projeto.

Matriz CSD:

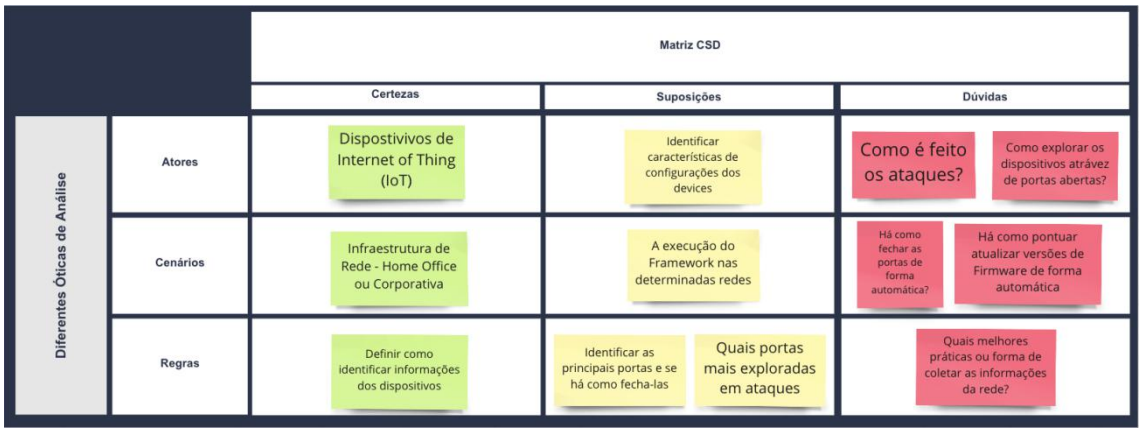


Figura 1 - Matriz CSD.
Elaborado pelo autor.

Também vamos aplicar o *framework* POEMS (Pessoas, Objetos, Ambientes, Mensagem e Serviços). O objetivo de aplicar este *framework* é orientar e estruturar a pesquisa deste trabalho. Uma vez que este facilita e visualmente exibe pontos muito importantes e facilitadores para a evolução deste.

Análise do contexto do problema – POEMS:

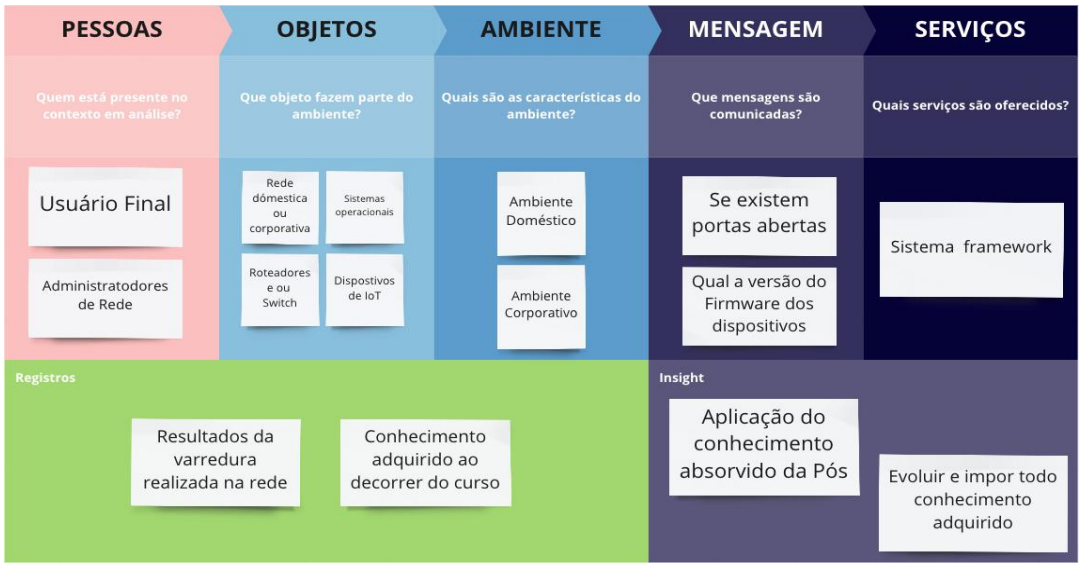


Figura 2- Framework - Análise de contexto POEMS.
Elaborado pelo autor.

1.1.2 Personas

Nesta etapa será apresentado as pessoas envolvidas diretamente neste cenário apresentado, assim como suas características, para exibição deste será abordado de acordo com o mapa da empatia e suas seções.

Mapa da Empatia:

Mapa de Empatia: Usuário de rede					
Quem	Fazer	Vê	Diz	Faz	Ouve
Usuário final de rede	Executar o framework em sua rede (doméstica ou corporativa)	Quais as portas que se encontram abertas dos dispositivos de ioT conectados em sua rede e também informações de Firmware destes.	Eu quero saber se existem e quais são as portas estão abertas dos dispositivos conectados na rede e também a versão do(s) firmware(s) destes.	Executa o Framework em sua rede.	Se há portas abertas e as versões dos firmwares.
Pensa / Sente					
Dores			Ganhos		
Ter conhecimento das portas abertas dos dispostivos de IoT conectados na rede.			A segurança em saber que os dispositivos estão com o escencial de configuração aplicada, assim garantindo uma segurança básica em seu ambiente.		

Tabela 1 - Mapa da Empatia – Usuário de rede.
Elaborado pelo autor.

Mapa de Empatia: Ameaça					
Quem	Fazer	Vê	Diz	Faz	Ouve
Humano ou sistêmico	Explorar as vulnerabilidades de configurações básicas dos dispositivos de IoT (portas abertas / firmware desatualizado)	A oportunidade de explorar as falhas e também manipular os dispositivos dado acesso concedido	Quero encontrar vulnerabilidades em dispositivos de IoT conectados na rede (internet)	Acessa os dispositivos, manipula os dados, altera configurações, monitora e tem conhecimento de contextos privados	Que existe adoção em larga escala de dispostivos de IoT e muitos dispositivos sem o minimo de configurações visando a privacidade assim concluído
Pensa / Sente					
Dores			Ganhos		
Perder o acesso aos dispositivos ou se quer conseguir acesso aos dispostivos de IoT			Financeiro uma vez que existe o roubo das informações. Também a boa visibilidade diante de outras comunidades.		

Tabela 2 - Mapa da Empatia - Ameaça.
Elaborado pelo autor.

1.1.3 Benefícios e Justificativas

A justificativa para o desenvolvimento deste projeto segue descrito abaixo:

- Como existem vários dispositivos de diferentes marcas e modelos disponível no mercado, a gestão sobre estes é sempre um problema, uma vez que se muda vários parâmetros e também características. Ou seja, dificuldade de validar todos os pontos de atuação dos mesmos.
- Estes dispositivos configuram-se como objetos de ponta e quando estes não se enquadrarem em pontos mínimos de segurança as diversas possibilidades de explorar estes acabam sendo de extrema importância aos atacantes.
- Conforme o projeto os dispositivos acabam sendo pontos de coletas das informações, uma vez que estes estejam vulneráveis nas redes a flexibilidade de manipular os mesmos e causar danos como alterações dos dados podem gerar reflexos a uma cadeia global de atuação dos mesmos.
- A falta de padrões nos diversos modelos de dispositivos que estão no mercado só aumenta a dificuldade de acompanhar e parametrizar uma rede ponto a ponto.
- Nos últimos tempos o número de residências automatizadas vem ganhando força, nestes cenários muitas vezes existem usuários limitados em conhecimentos técnicos para visualizar algumas informações básicas dos dispositivos, o que aumenta ainda mais a possibilidade de ataques e manipulações a redes domésticas.

Abaixo é pontuado os benefícios obtidos com este projeto e também com a resolução deste problema:

- A garantia de possuir um ambiente quais os dispositivos de IoT estão com as configurações mínimas setadas.
- A centralização da informação sobre estes dispositivos a fins de otimizar a gestão de quem controla a rede, neste aspecto podemos descrever usuários básicos aos mais avançados.
- A fácil manipulação do *software* a fins e aplicável em qualquer cenário de rede, seja doméstica ou corporativa.
- Ao acréscimo de entender mais a fundo sobre pontos que seus dispositivos já vêm pré- configurados e assim poder tomar medidas básicas de segurança.

Para facilitar a compreensão deste cenário, abaixo será apresentado as interações existentes através da metodologia *Blueprint*. Aplicando esta metodologia é possível visualizar e também compreender a fins de corrigir pontos e melhorar de fato o projeto como um todo. Na sequência também será apresentado o *framework* de Canvas, que visa apresentar como objetivo a proposta de valor deste projeto.

Blueprint

Tarefas e processos para alcançar os objetivos esperados.					
Blueprint	Desenvolver um framework	Identificar como realizar a coleta básicas dos dispositivos de IoT	Montar ambiente doméstico para execução dos testes	Desenvolver técnicas para realizar as varreduras nos dispositivos de IoT	Garantir o retorno e formas de visualizar os dados após as varreduras
Ações do Cliente	Identificar, varrer e analisar as configurações de porta e também versão de Firmware dos dispositivos de IoT				
Objetivos	Identificar os dispositivos de IoT conectados a rede e trazer infomações de portas abertas e também versões em execuções dos firmwares destes				
Atividades	Realizar varredura na rede geral para obter as informações desejadas				
Questões	O framework sera desenvolvido em qual linguagem? Quais técnicas serão adotadas? O framework terá que ser executado em dispositivos de baixo processamento como raspberry PI.				
Barreiras	Trabalhar com 2 ou 3 modelos de dispositivos IoT, Custos para atribuílção destes dispostivos e ambiente para executar estes dispositivos.				
Saída desejavel	Garantir a verredura e coleta das portas abertas mais versões dos Firmwares dos dispositivos				
Funcionalidades	Captura de informações dos dispostivos				
Interação	Operar o framework (software) e visualizar o relatório final				
Mensagem	Identificação das características mínimas para realizar uma configuração adequada e mais segura em seu ambiente				
Onde Ocorre	Rede doméstica e ou Rede corporativa				
Tarefas aparentes	Definir linguagem para desenvolver o framework, adotar técnicas x bibliotecas x tecnologias para adicionar ao mesmo				
Tarefas Escondidas	Execução das varreduras e tratamento dos resultados				
Processos de Suporte	Validação dos resultados de forma individual após a aplicação do framework e seus resultados				

Tabela 5 - Blueprint.
Elaborado pelo autor.

Canvas – Proposta de Valor:

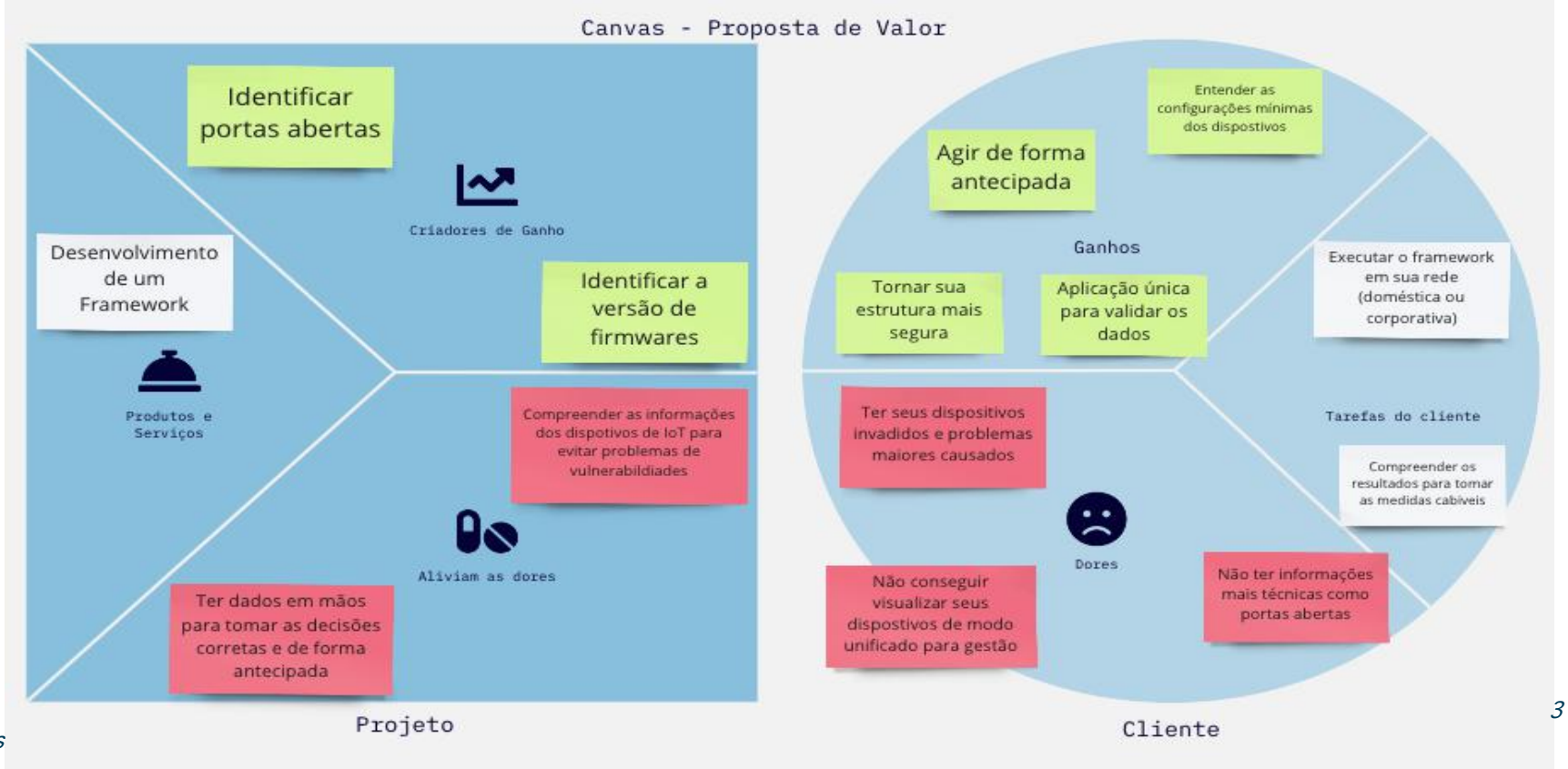


Figura 3 - Canvas

Elaborado pelo autor.

1.1.4 Hipóteses

Em relação as hipóteses mapeadas para elaboração deste projeto observamos alguns pontos que seguem listados abaixo.

Observação	Hipóteses
Devido ao crescimento na adoção de dispostivos de IoT em residencias ou companhias, a preocupação em relação a como estes dispostivos estão sendo configurados acaba muitas vezes sendo esquecidas.	Imagina-se por muitos que estes dispositivos não tem potencial de exploração em relação a vulnerabilidades, o que não é verdade.
Uma vez que estes dispostivos ficam expostos sem uma segurança básica, as chances de ataques e também manipulação de informações são sempre equivalentes.	Como identificar se os dispositivos estão expostos de fato? Quem da a devida importancia a isso? Será que é ou são assuntos de relevancia para pessoas e empresas?
A falta de padronização destes dispostivos os torna ainda mais comprometedores.	Os fabricantes não seguem padrões e também não se preocupam com pontos de tornar os mesmos mais seguros e de fácil comunicação com outros dispostivos.
A mesclagem de dispositivos torna a gestão muito mais complexas, isso em qualquer cenário, assim torna o processo de acompanhamento muito complexos e demorado.	Usuários domésticos e ou usuários mais avançados acabam tendo sempre uma ação de gestão mais massante para identificar certas informações que são basicas em redes onde se aplicam estes dispositivos.

Tabela 6 - Matriz de observação para hipóteses.
Elaborado pelo autor.

Após levantamento das hipóteses apresentadas acima, na sequência foi feito um alinhamento com o objetivo de priorizar as ideias em relação aos vários processos e pontos a serem implementados. Para pontuar uma forma entre as várias levantadas para o desenvolvimento deste, são:

- Opção 01 = Desenvolvimento de um *software* para varrer uma determinada rede e informar de modo intuitivo os resultados ao usuário final;
- Opção 02 = Atuar em fazer varreduras visando capturar o máximo de informações possíveis e informar aos usuários – modo monitoramento em tempo real;
- Opção 03 = Visualizar soluções abertas no mercado e adaptá-las a um determinado cenário;
- Opção 04 = Tentar integrar os dispositivos com *softwares* que fazem gestão de redes mais avançados, como *firewalls*.

Cenários:

C1	Complexidade na execução do projeto
C2	Urgência na execução do projeto
C3	Investimento necessário para explorar o projeto
C4	Benefícios do projeto
C5	Satisfação da direção

Tabela 7 - Tabela de Cenários.
Elaborado pelo autor.

Escala:

Escala	B - Benefícios	A - Abrangência	S - Satisfação	I - Investimentos	C - Cliente	O - Operacionalidade
5	De vital importância	Total (de 70 a 100%)	Muito grande	Pouquíssimo investimento	Nenhum impacto	Muito fácil
4	Significativo	Muito grande (de 40 a 70%)	Grande	Algum investimento	Impacto pequeno	Fácil
3	Razoável	Razoável (de 20 a 40%)	Média	Médio investimento	Médio impacto	Média facilidade
2	Poucos benefícios	Pequena (de 5 a 20%)	Pequena	Alto investimento	Impacto grande	Difícil
1	Algum benefício	Muito pequena	Quase não é notada	Altíssimo investimento	Impacto muito grande no cliente	Muito difícil

Tabela 8 - Tabela de Escala.
Elaborado pelo autor.

Comparação dos cenários:

Soluções	B - Benefícios	A - Abrangência	S - Satisfação	I - Investimentos	C - Clientes	O - Operacionalidade	Total
Opção 01	5	5	4	4	3	3	24
Opção 02	5	5	4	3	3	2	22
Opção 03	5	5	4	2	3	2	21
Opção 04	5	5	4	2	2	2	20

Figura 4 - Comparação dos Cenários.
Elaborado pelo autor.

1.2 Solução

Para alcançar os objetivos deste projeto aplicado, será desenvolvido um *software* chamado MSIS, este vai atuar na varredura de rede para realizar a coleta de informações como portas abertas e informações adicionais como, *firmware* dos dispositivos de *Internet of Thing* (IoT). O MSIS poderá ser executado em dispositivos computacionais mais robustos como também em dispositivos como *Raspberry* PI, a base do mesmo é a conexão na mesma rede de varredura.

Os resultados obtidos nestas varreduras devem ser objetivos para facilitar a interpretação dos usuários e assim poder auxiliar os mesmos a tomar as devidas atitudes para fins de aumentar o nível de segurança destes dispositivos. O MSIS poderá ser executado em redes domésticas e também redes corporativas.

1.2.1 Objetivo SMART

Com relação aos objetivos deste projeto, é esperado alcançá-los seguindo a exposição na tabela SMART, listados abaixo:

S (Específico)	Fazer varredura em redes para obter informações dos dispositivos de IoT
M (Mensurável)	Exibir informações importantes para o usuário tomar as devidas ações a fins de melhorar a segurança
A (Atíngivel)	Realizar a varredura em redes e pontuar portas abertas + informações extras como Firmware dos dispostivos
R (Relevante)	Poder centralizar essa função atráves de um framework
T (Temporal)	Identificação de informações importantes que visam auxiliar na gestão da rede

Tabela 9 - Tabela SMART.
Elaborado pelo autor.

1.2.2 Premissas e Restrições

Este projeto apresenta as seguintes premissas:

- Desenvolvimento de um *software*;
- Este *software* deve fazer varredura em redes para fins de coletar dados (porta e informações extras) dos dispositivos conectados e exibir em forma objetiva;
- Os resultados deverão ser positivos para que os usuários possam validar e tomar ações sobre;
- O *software* pode ser executado em dispositivos como *Raspberry* PI.

Quanto a relação das restrições segue abaixo:

- O *software* tem que ser aplicado a redes domésticas;
- O processo deve ser satisfatório para os usuários;
- A coleta de dados deve ser o mais objetiva;
- O *software* deve ser desenvolvido sobre uma arquitetura (linguagem) aberta (*Open Source*) .

Em reação aos riscos do projeto, levando em consideração as premissas e restrições foram identificados alguns pontos onde foram elaborados os riscos:

Risco	Probabilidade	Impacto	Ação a ser realizada
Usar um formato para elaboração do framework objetivo	ALTO	ALTO	Cuidadosamente escolher uma linguagem que pode ser utilizada sem restrições de custos
Elaboração de técnicas visando ser objetivo na pratica de varredura	MEDIO	ALTO	Observar as técnicas e formas quais serão aplicado as técnicas
Não dificultar a operação no framework a fins de tornar o mesmo inviavel de uso	MEDIO	ALTO	Desenvolver um mecanismos agil e simples para operação do framework
Tornar os resultados muito complexos a fins de dificultar a interpretação	ALTO	ALTO	Observar e filtar o máximo para tornar as respostar objetivas
Validar os resultados finais com cada dispostivo individual	BAIXA	BAIXA	Depois de opter os resultados, validar os mesmos com cada dispositivo

Tabela 10 - Tabela de Risco.
Elaborado pelo autor.

1.2.3 Backlog de Produto

Neste momento segue abaixo a exibição do projeto:

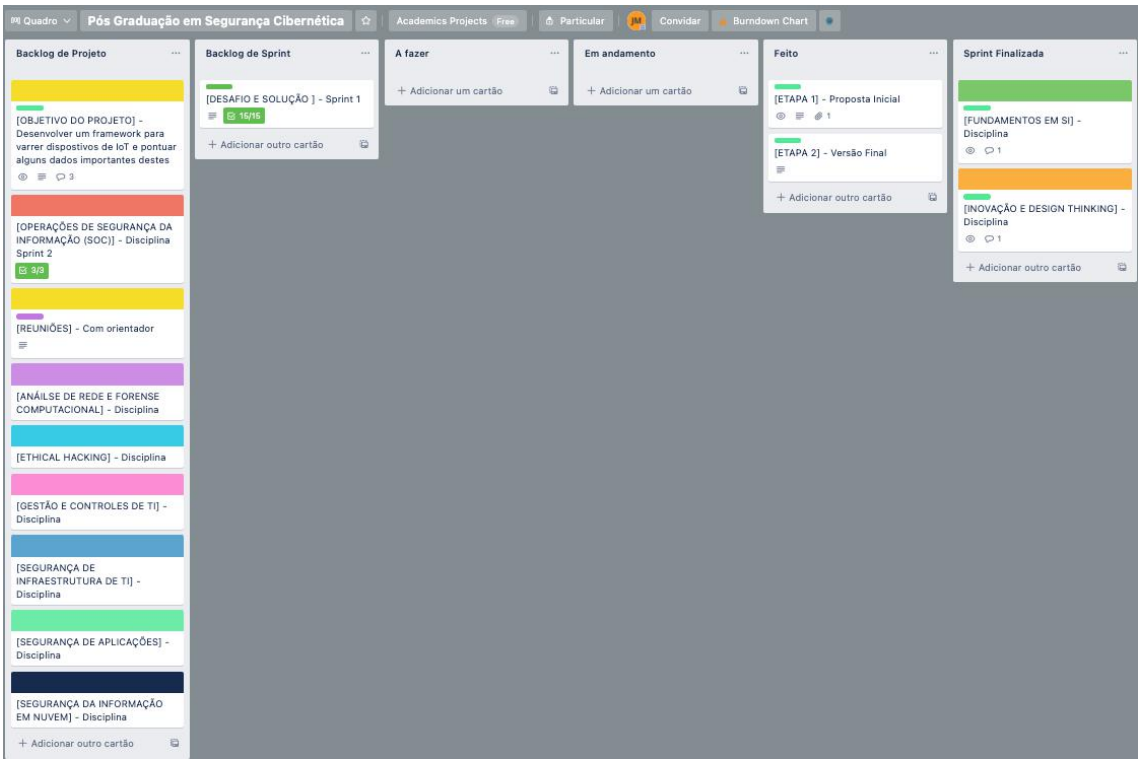


Figura 5 - Projeto Backlog.
Elaborado pelo autor.

Este projeto está classificado da seguinte maneira:

1. Sprint: Disciplina = Definição do Desafio / Problema, criação análise do contexto;
2. Sprint: Disciplina = Operações de segurança da informação;
3. Sprint: Disciplina = Segurança da informação em nuvem;
4. Sprint: Disciplina = Segurança de infraestrutura de TI;
5. Sprint: Disciplina = *Ethical Hacking*;
6. Sprint: Disciplina = Análise de rede e forense computacional;
7. Sprint: Disciplina = Segurança de aplicações;
8. Sprint: Disciplina = Gestão e controles de TI.

Onde estamos concluindo a etapa de definição do desafio / problema juntamente com a reunião *Kick-Off* e definição do escopo.

2. Área de Experimentação

2.1 Sprint 1

Primeira etapa do projeto aplicado, no qual serão definidos os seguintes elementos: Desafios, Análise do Contexto, Matriz CSD, *Framework* POEMS, *Personas*, Mapa de Empatia, Benefícios e Justificativas, *Blueprint*, Canvas - Proposta de Valor, Hipóteses. Matriz de observação para Hipóteses, Priorização de Ideias, Solução, Objetivos SMART, premissas e restrições, matriz de risco e *backlog* do produto.

2.1.1 Solução

- Evidência do planejamento:

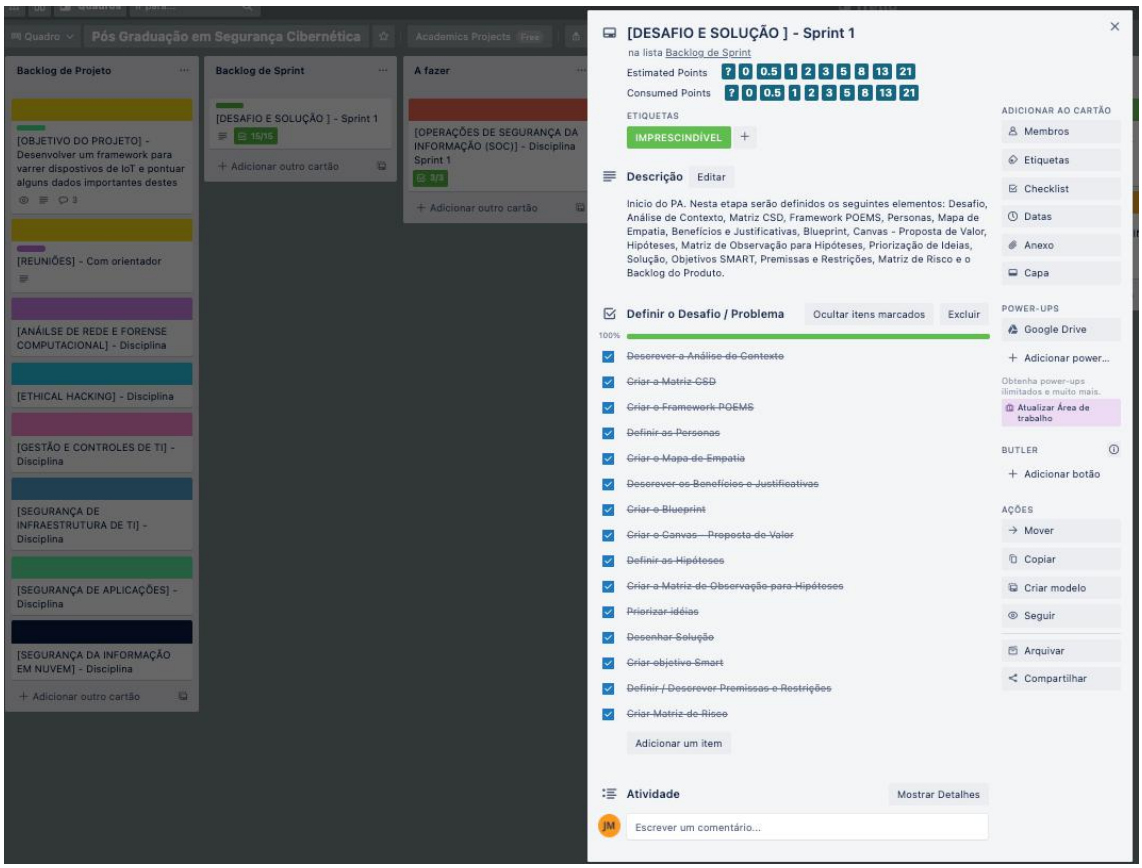


Figura 6 - Planejamento.
Elaborado pelo autor.

- Evidência da execução de cada requisito:

A execução desta primeira Sprint, pode ser observada através dos capítulos anteriores deste documento, qual todas as atividades descritas na ferramenta Trello, ambas devidamente executadas e finalizadas.

- Evidência da solução:

Destacam-se como evidências da solução os seguintes itens: (a) descrição da análise do contexto; (b) criação da matriz CSD; (c) criação do *framework*; (d) descrição das personas; (e) criação do mapa de empatia; (f) descrição dos benefícios / justificativas acerca da realização do projeto; (g) criação do *Blueprint*; (h) criação do CANVAS - Proposta de Valor; (i) definição das hipóteses atreladas ao projeto; (j) criação da matriz de observação para hipóteses; (l) priorização de ideias em conjunto com a equipe de TI e alta administração; (m) definição da solução a ser aplicada para o desafio / problema; (n) descrião das premissas e restrições e por fim; (o) criação da matriz de riscos.

2.1.2 Lições aprendidas

Ficou claro a importância da criação dos elementos descritos acima para que a compreensão da construção do projeto se torne mais clara. A disciplina de *Design Thinking* foi um auxílio no entendimento dos conceitos relacionados para a criação da abordagem inovadora e bem estruturada que tem por fim apresentar soluções que estejam alinhadas ao desejo e as necessidades deste projeto.

2.2 Sprint 2

Abaixo segue a imagem do planejamento pontuado para esta disciplina. Esta disciplina foi mais voltada ao aspecto de organização e funcionamento de um departamento de segurança da informação em companhias que optam por adotar esta metodologia. Para o projeto alguns pontos foram levados em conta assim com o objetivo de mostrar como é possível aplicar o MSIS em um ambiente corporativo.

2.2.1 Solução

- Evidência do planejamento:

Abaixo segue a imagem do planejamento pontuado para esta disciplina. Uma vez que a mesma retrata assuntos bem voltados a organização e também a conceitos mais aprimorados de uma organização, vamos absorver algumas situações para indicar como melhor aplicar o nosso projeto aplicado a ambientes corporativos.

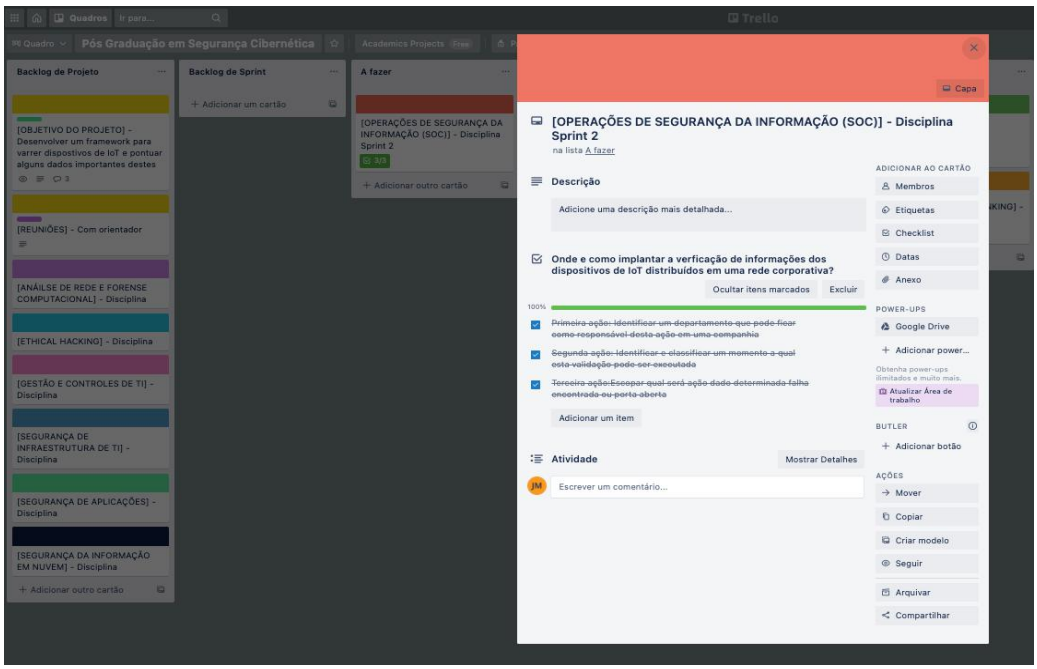


Figura 7 - Projeto de Operações de Segurança da Informação SOC.
Elaborado pelo autor.

- Evidência da execução de cada requisito:

Primeira ação: A identificação do departamento responsável dentro de uma organização corporativa para atuar na execução de um processo de varredura em dispositivos de IoT. Compreendendo a abordagem do cenário de uma empresa corporativa estruturada onde conta com uma estrutura organizada e definida com SOC e CSIRT. Podemos definir que a varredura por se tratar de uma ação de forma manual e também de uma análise humana na avaliação de seus resultados pode ser adicionada na primeira camada, neste caso seria na fase do SOC.

Este evidência pode ser dada na refinação de uma estrutura onde por exemplo podemos obter a seguinte definição: Primeiro nível de atendimento SOC e um segundo nível de atendimento CSIRT.

Segunda ação: Por classificar e identificar o papel inicial de execução do MSIS, uma validação construtiva seria em um escalonamento de evidências. Ou seja, uma vez que temos a SOC atuando de modo diário, semana ou até mensal – buscando identificar falhas nos dispositivos de IoT de sua rede, em um segundo nível pode vir a ser o CSIRT que pode atuar na correção seja de portas abertas ou de atualizações desejadas por cada dispositivo.

Terceira ação: A triagem de escalonar qualquer problema encontrado pela SOC na rede dos dispositivos de IoT pode ser efetivo após a identificação dos logs qual o MSIS deve gerar. Ação esta que visa otimizar a função e agir o mais rápido possível dado qualquer inconsistência observada pelo time SOC.

- Evidência da solução:

Para facilitar as evidências e deixar mais objetivo o processo que pode ser seguido segue a imagem abaixo.

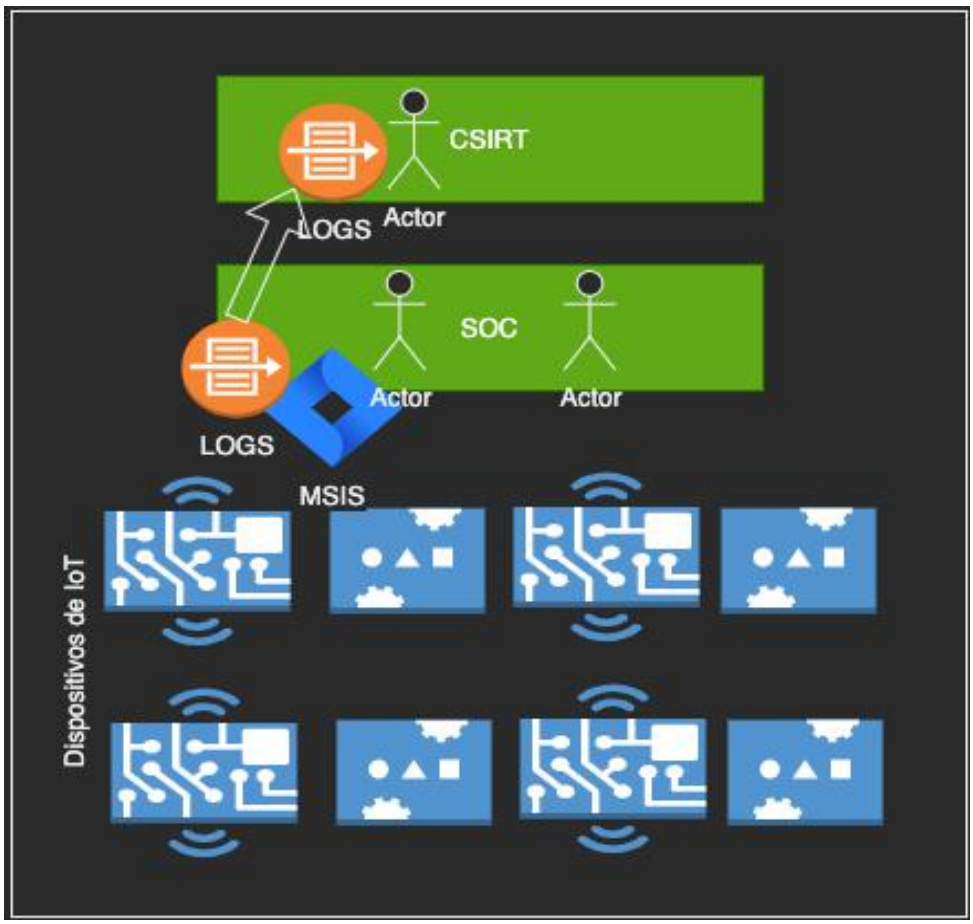


Figura 8 - Exemplo de fluxo que pode ser implantado em uma companhia.
Elaborado pelo autor.

2.2.2 Lições aprendidas

A importância de uma organização trabalhar com um departamento exclusivo na atuação de proteção de seus ativos. Uma grande premissa que se leva em consideração é também a realidade de cada negócio. Porém conforme os negócios vão sendo cada vez mais digitais a dependência por ter uma estrutura seja *on-premisses*, *cloud* ou terceirizada deve ser tratada de modo muito sério pela alta direção.

Cada necessidade deve levar em conta aspectos pontuais de investimentos, muitos gestores não validam e pontuam investimentos para as áreas de segurança, o que torna sempre mais complexo o investimento e proteção destas áreas dentro das companhias.

Atualmente as empresas buscam produzir mais investindo sempre menos possível, esta é uma discussão que cabe muitas colocações, uma vez que em muitos segmentos a estratégia vem sendo adotada com a substituição da mão de obra humana por automações e pela robótica. Ponto este que vem de encontro com o objetivo deste projeto aplicado. Uma vez que *Internet of Thing*

(IoT) vem ganhando muito espaço dentro de várias indústrias, independente de seus segmentos.

A grande falha dessas viradas é a não visualização aos *gaps* que surgem dentro dos ambientes corporativos quando olhamos para o cenário de segurança da informação. Ambientes onde são adotados sensores, computadores, coletores, câmeras e muitos outros dispositivos acabam sendo implantados e esquecidos de monitorar o contexto destes dispositivos, como consequência acabam sendo explorados por vulnerabilidades de configurações mal realizadas, *bugs* de *firmwares*, times de profissionais de segurança da informação não constituídos dentre outros fatores.

Ao concluir esta disciplina ficou claro o quão importante é uma estrutura bem modela com objetivos bem definidos tanto pela alta direção de qualquer empresa quanto pelo time de segurança da informação e também os processos muito bem definidos para que o dia a dia seja dado por uma organização a fins de atuar de modo conjunto e com o mesmo propósito, efetuar a segurança máxima de seus colaboradores e também seus ativos.

2.3 Sprint 3

Abaixo segue a imagem do planejamento para esta disciplina. A abordagem da mesma foi direcionada a infraestrutura alocada em nuvem (*cloud*). Alguns exemplos de ferramentas foram visualizados na disciplina, também exemplos de arquiteturas direcionadas como alternativas para determinados projetos. Depois de avaliado os pontos abordados na disciplina e realizado uma análise para aplicação ao projeto do MSIS, chego a conclusão que não há pontos que possam ser aplicáveis até determinado momento. Desta forma não haverá uma abordagem nesta *sprint* sobre a disciplina.

2.3.1 Solução

- Evidência do planejamento:

Abaixo segue a imagem do planejamento pontuado para esta disciplina, disciplina que aborda referências sobre segurança aplicadas em arquitetura de nuvem. Realizado várias análises buscando referenciar como pré requisito o projeto deste PA.

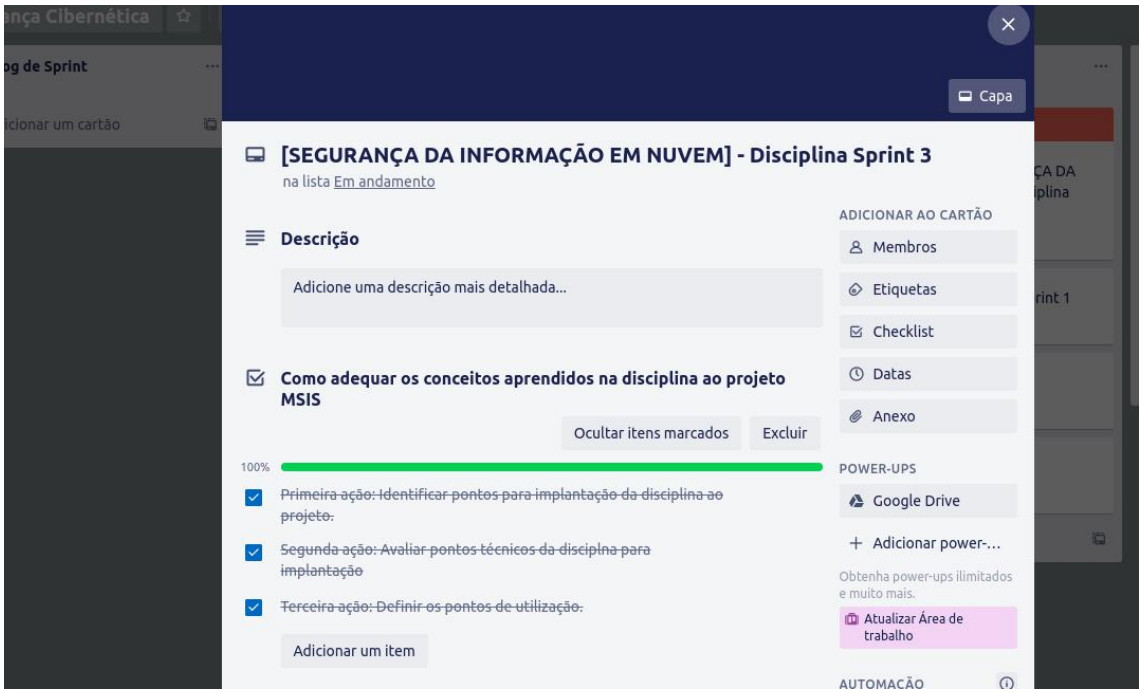


Figura 9 – Projeto de Segurança da Informação em nuvem.

Elaborado pelo autor.

- Evidência da execução de cada requisito:

Primeira ação: Durante a disciplina foi observado pontos mais abstratos quais na segunda opção seriam observados e avaliados para fins de estressar um nível de aprendizado e aplicação mais técnico.

Segunda ação: Depois de avaliar tópicos mais abstratos foi avaliado um nível mais técnico para fins de buscar aproveitar ferramentas e também conceitos afins de ser compatível com o projeto atual.

Terceira ação: Depois de absorver conceitos e visualizar ferramentas, não tive uma métrica técnica para usufruir de detalhes a fins de utilização ao projeto atual. Uma vez que a arquitetura do projeto PA é algo voltado a *on-premisses* e não *cloud*.

- Evidência da solução:

Como a abordagem da disciplina foi direcionada a conceitos e ferramentas usadas para proteção de arquitetura em cloud, o conhecimento adquirido foi direcionado a visualizar e conhecer algumas ferramentas dessas, exemplo da AWS. Uma vez que se projeta usar este modelo de arquitetura, principalmente em ambientes corporativos, vários cuidados devem ser levados em consideração para manter um nível elevado de proteção. A disciplina teve muitos momentos de troca de experiências abordados pelo professor nas aulas ao vivo, o qual agregou uma visualização e compreensão de forma mais abrangente a estes cenários.

2.3.2 Lições aprendidas

Ao concluir esta disciplina foram realizadas várias análises para identificar contextos como metodologias e técnicas abordadas na disciplina para fins de utilizar no projeto aplicado. Infelizmente no momento não foi aplicável ao mesmo uma vez que o objetivo do projeto é voltado a estruturas *on-premisses*, levando em conta aplicações pontuadas a dispositivos de internet das coisas (IoT).

2.4 Sprint 4

2.4.1 Solução

- Evidência do planejamento:

Nesta diciplina esperava-se um conteúdo de extrema importância para o projeto aplicado. Assim esta disciplina ganhou uma atenção totalmente redobrada para fins de refinar o conhecimento teórico, entender algumas metodologias e aplicar o conhecimento na prática.

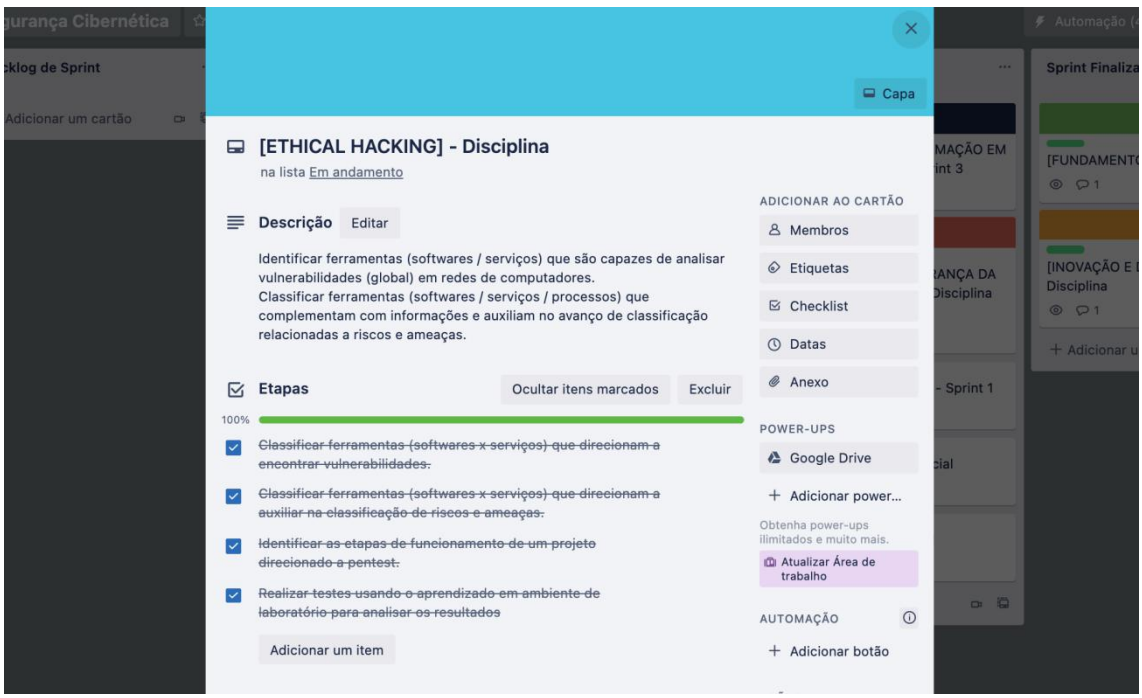


Figura 10 – Projeto de Ethical Hacking.

Elaborado pelo autor.

- Evidência da execução de cada requisito:

Após alguns estudos realizados na disciplina, abaixo seguem a exibição de algumas ferramentas que podem ser utilizadas na fase de análise de vulnerabilidades.

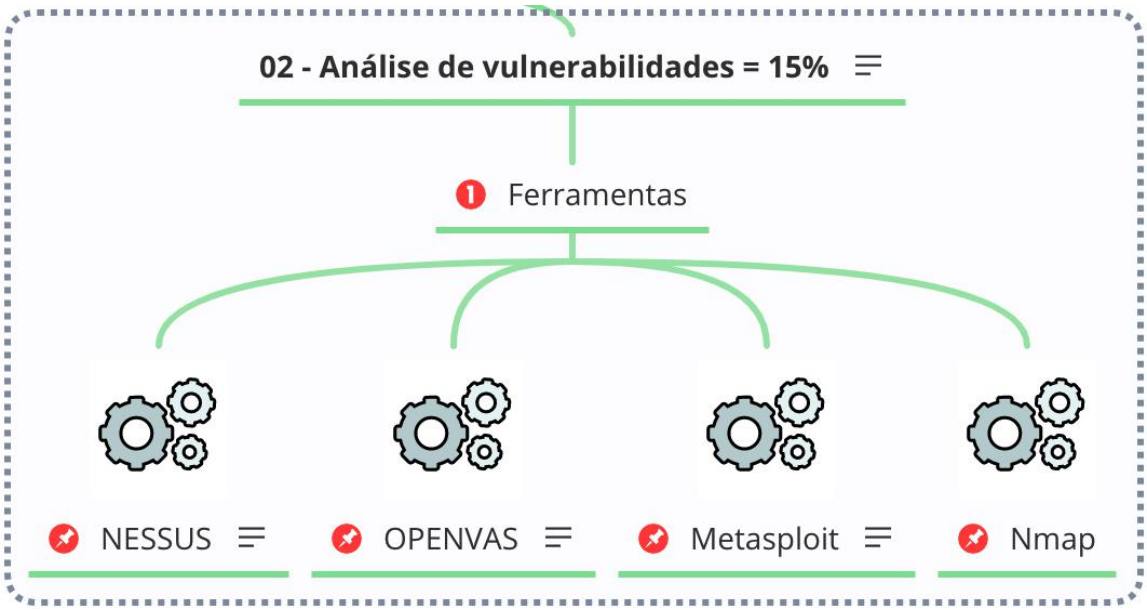


Figura 11 – Análise de vulnerabilidades.

Elaborado pelo autor.

Em um segundo momento, foi realizado uma classificação de algumas ferramentas para auxiliar em um momento de exploração de vulnerabilidades, abaixo a exibições destas ferramentas.

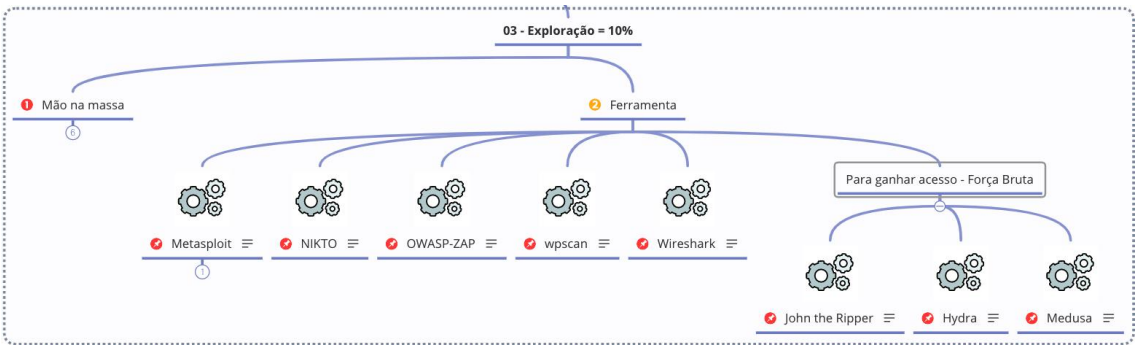


Figura 12 – Exploração.

Elaborado pelo autor.

Entendendo as fases de um projeto de Pentest, assim como o objetivo deste foi buscar a compreensão abaixo foi desenvolvido um modelo de mapa mental buscando abrir e dividir cada etapa deste.

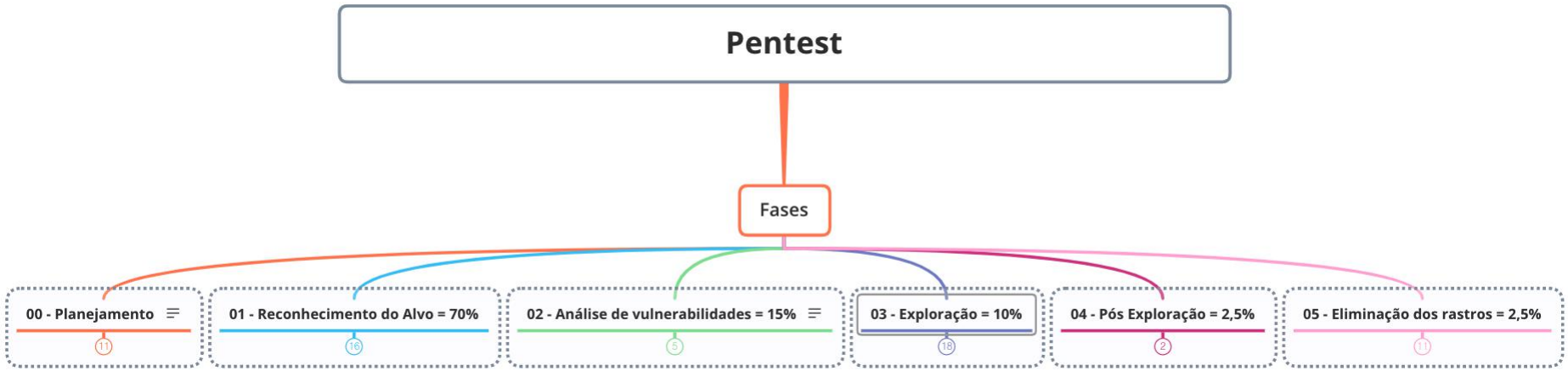


Figura 13 – Representação das fases de Pentest.

Elaborado pelo autor.

- Evidência da solução:

Para evidênciar de forma ampla o aprendizado, abaixo segue uma representação de uma modelagem desenvolvida para esboçar alguns parametros e como proceder em pontos quando se busca algumas váriavel de informação.

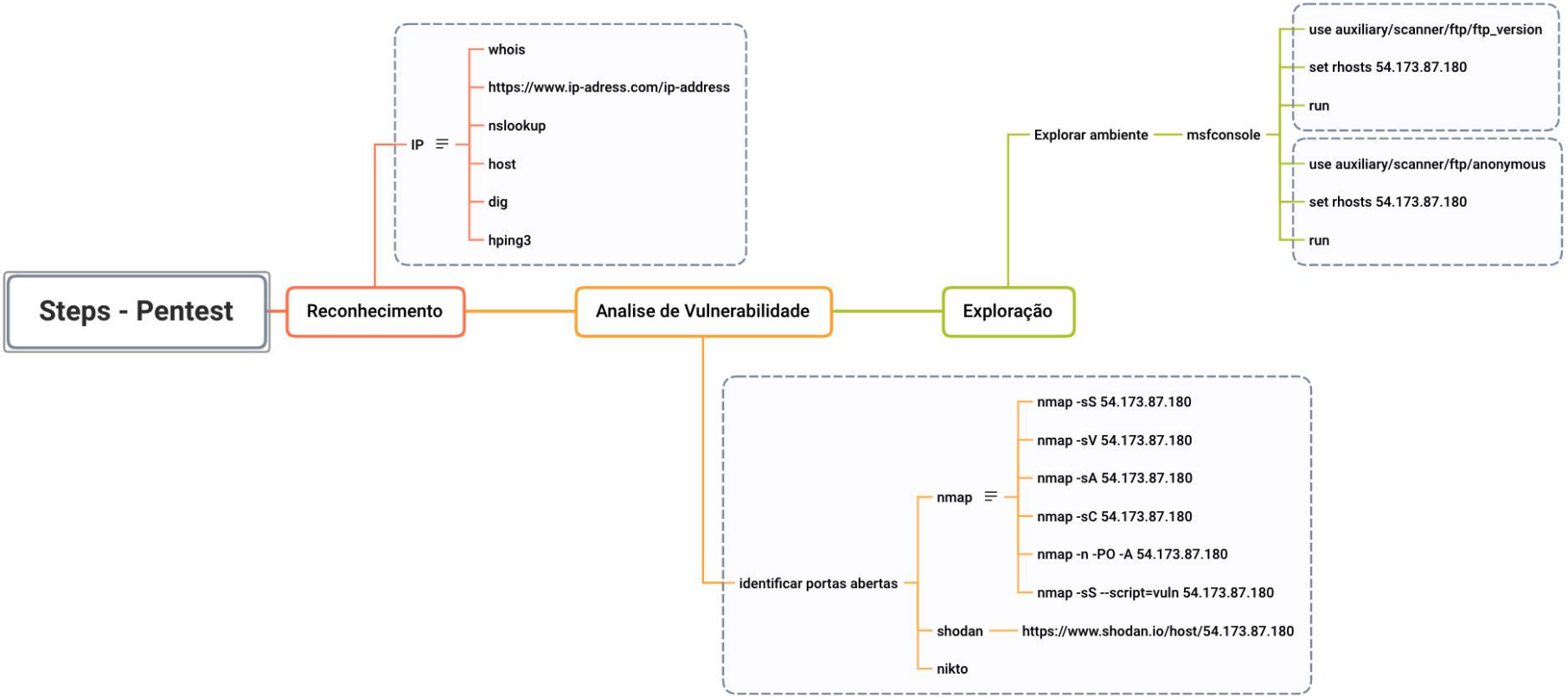


Figura 14 – Etapas de Pentest.

Elaborado pelo autor.

A figura acima demonstra de forma simples e objetiva pontos que podem ser levados em conta quando se busca avançar em reconhecimento de ambiente análise de vulnerabilidades e exploração.

Avançando nos testes abaixo segue mais exemplos. A busca por identificar portas abertas em determinados destinos usando ferramentas como Nmap.

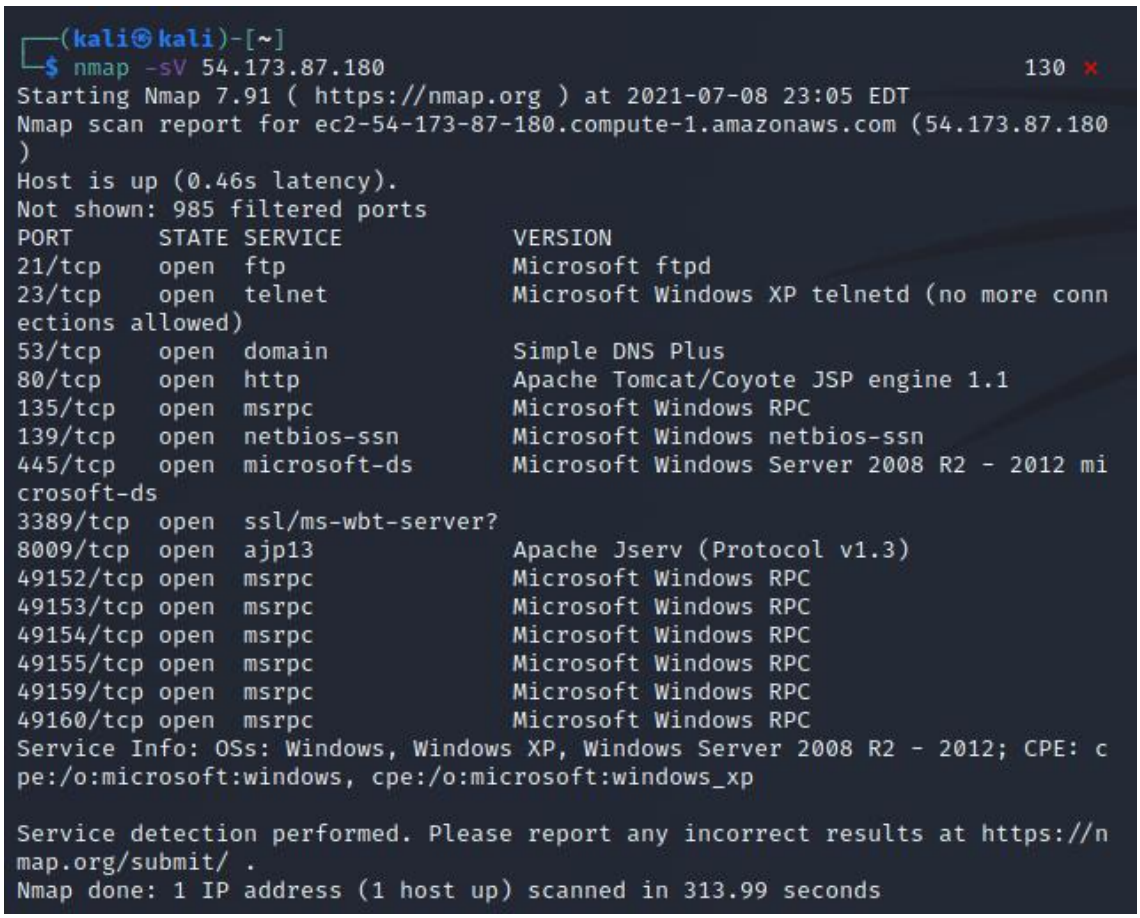


Figura 15 – Exemplos de testes.

Elaborado pelo autor.

Importante destacar que estas ferramentas são utilizadas sobre uma plataforma livre e também um sistema operacional que tem por objetivo atuar em foco a segurança da informação, este chamado de Kali Linux.

Avançando nos testes e explorando mais recursos da ferramenta Nmap podemos identificar que é possível obter informações de extrema importância de destinatarios diversos.

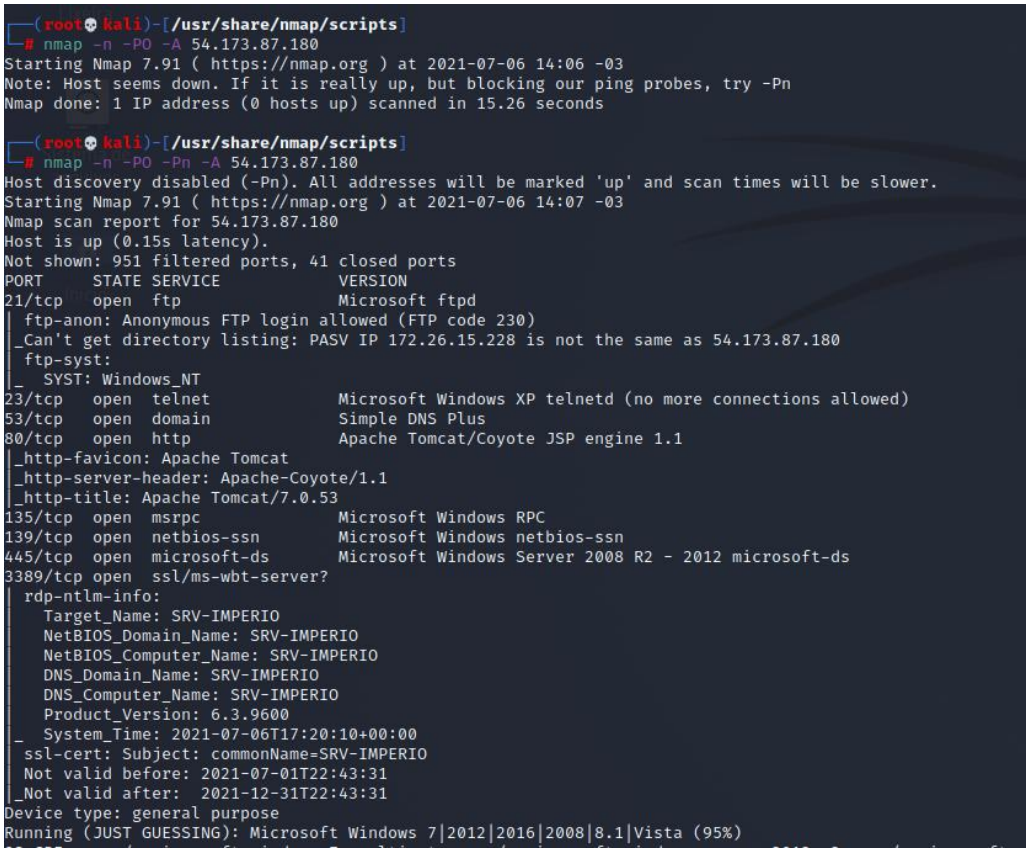


Figura 16 – Exemplos de testes.

Elaborado pelo autor.

Outra ferramenta utilizada nos testes e que chamou bastante atenção foi o Nikto. Mas um ponto ficou bem claro, é uma ferramenta voltada mais a soluções e ou destinos de arquiteturas WEB.

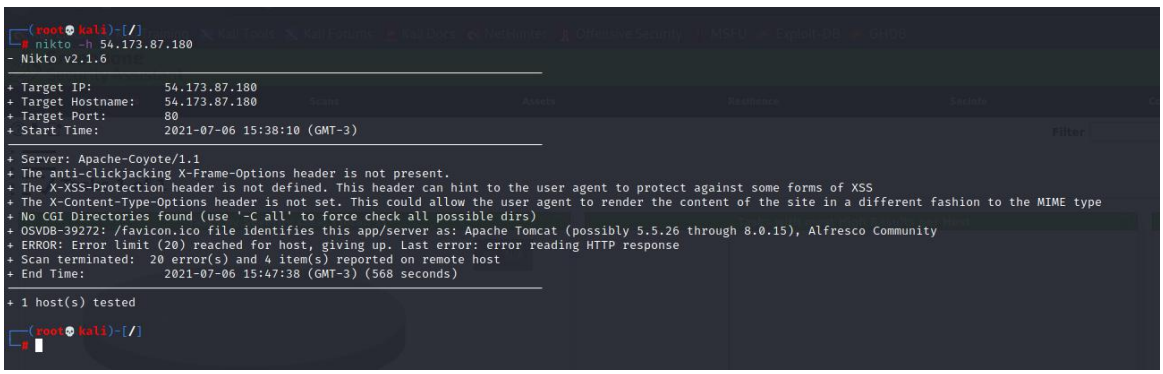


Figura 17 – Exemplos de testes.

Elaborado pelo autor.

Para obter informações sobre DNS e endereço de IP, foi utilizado a ferramenta Dig, que trouxe alguns dados que podem ser usados em caso de exploração a ambientes mais pontuais a explorações WEB.

Figura 18 – Exemplos de testes.

Elaborado pelo autor.

Nos testes também foi utilizado o hping3, ferramenta que é possível detectar *hosts*, regras de *firewall* e também varredurtas em portas.

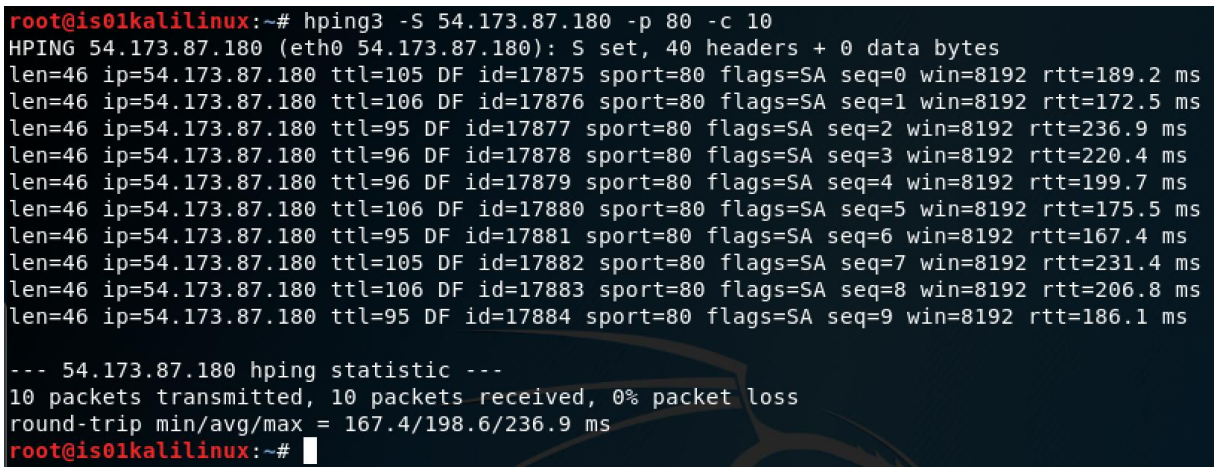


Figura 19 – Exemplos de testes.

Elaborado pelo autor.

Evoluíndo os testes, chegamos no Metasploit mais usados de fato para momentos em que deseja explorar pontos de vulnerabilidades encontrados no processo como um todo.

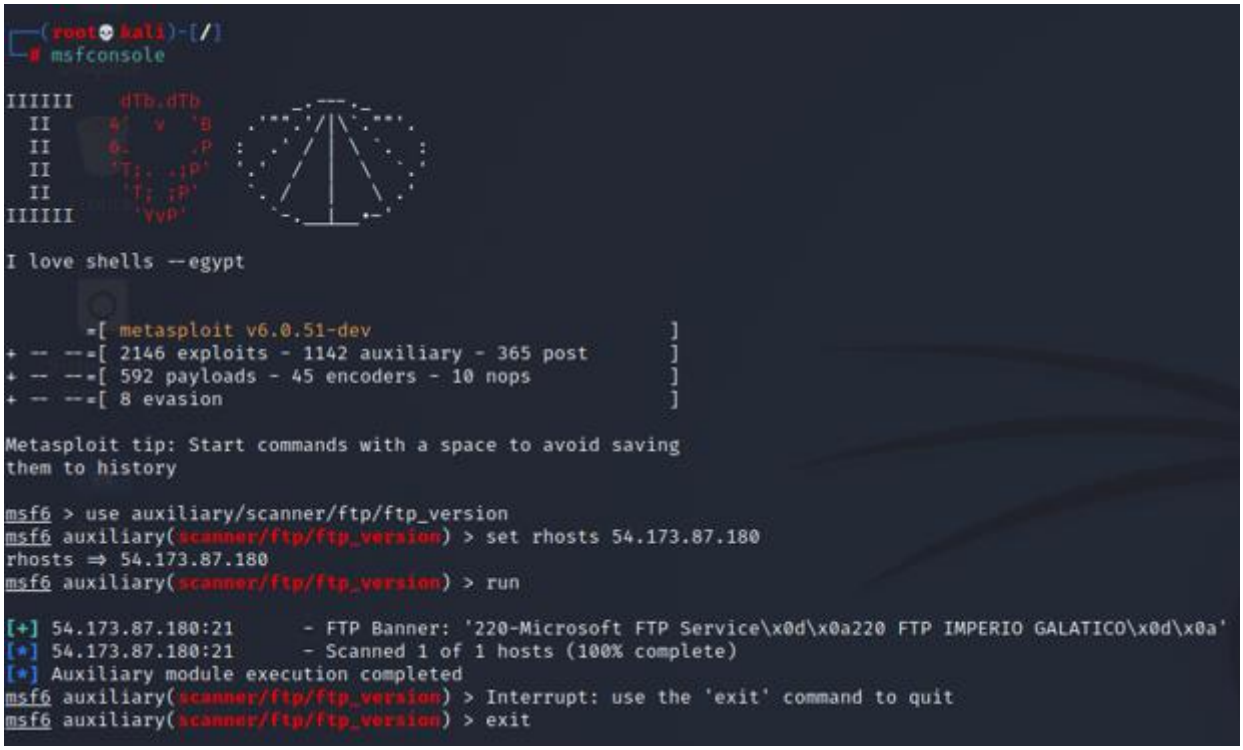


Figura 20 – Exemplos de testes.

Elaborado pelo autor.

2.4.2 Lições aprendidas

Durante a disciplina de *Ethical Hacker* foi possível adquirir um nível de conhecimento técnico de extrema relevancia para ações de execuções ofensivas relacionados a ataques em redes de computadores. Uma disclina que conseguiu indexar a teoria com a prática de forma muito produtiva para um aprendizado de qualidade.

Após o conhecimento adquirido na disciplina, se aproveita muitos embasamentos técnicos e práticas para anexar ao projeto aplicado. Vários testes já foram executados para visualizar e classificar algumas técnicas para implantação no MSIS.

2.5 Sprint 5

Abaixo segue a imagem do planejamento para esta disciplina. A abordagem da mesma foi direcionada a cenários forenses. Conceitos e exemplos de ferramentas foram visualizados na disciplina, também exemplos de cenários de aplicações da técnica forense para determinados casos. Depois de avaliado os pontos abordados na disciplina e realizado uma análise para aplicação ao projeto do MSIS, chego a conclusão que não há pontos que possam ser aplicáveis até determinado momento. Desta forma não haverá uma abordagem nesta *sprint* sobre a disciplina.

2.5.1 Solução

- Evidência do planejamento:

Abaixo segue a imagem do planejamento pontuado para esta disciplina, disciplina que aborda sobre análise de rede e forense computacional. Realizado várias análises buscando referenciar como pré requisito o projeto deste PA.

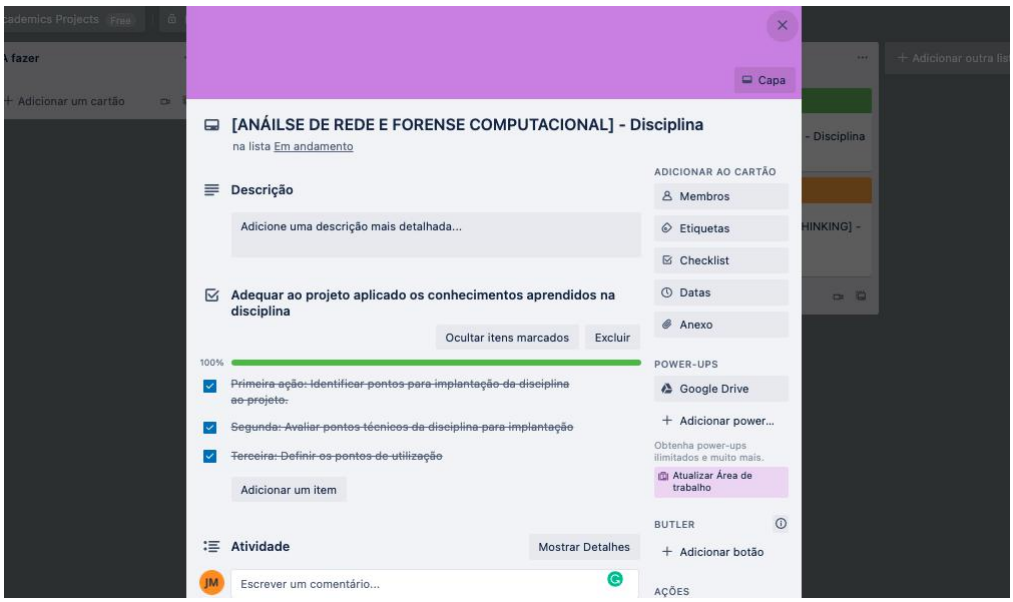


Figura 21 – Projeto da disciplina de Análise de rede e forense computacional.

Elaborado pelo autor.

- Evidência da execução de cada requisito:

Primeira ação: durante a disciplina foi observado pontos mais abstratos quais na segunda opção seriam observados e avaliados para fins de estressar um nível de aprendizado e aplicação mais técnico.

Segunda ação: depois de avaliar tópicos mais abstratos foi avaliado um nível mais técnico para fins de buscar aproveitar conceitos e ferramentas afins de ser compatível com o projeto atual.

Terceira ação: depois de absorver os conceitos e visualizar as ferramentas, não ficou claro nenhuma métrica técnica para usufruir de detalhes a fins da utilização ao projeto atual. Uma vez que a arquitetura do projeto PA é voltado a defesa e não a ação forense.

- Evidência da solução:

A abordagem da disciplina foi direcionada a regras, conceitos e ferramentas usadas por peritos para a realização de ações forense. Assim o conhecimento foi direcionado para a compreensão destes cenários. A disciplina teve muitos momentos de troca de experiências abordados pelo professor nas aulas ao vivo, o qual agregou uma visualização e compreensão mais aprofundada dos cuidados e como funciona toda a cadeia de ações quando tratados assuntos forenses.

2.5.2 Lições aprendidas

Ao concluir esta disciplina, concluído-se que as metodologias e técnicas abordadas na disciplina não se enquadram na aplicabilidade do projeto aplicado. O PA foca em ações de ataque x defesa, anulando ações mais direcionadas a pontos de abordagem forense. Os conceitos aprendidos, debatidos em sala de aula, são de extrema importância para avaliações e cenários quais toda ação técnica deve ser levada com extremo cuidado seguindo regras e buscando sempre a evidência ao mais claro e objetivo ponto de esclarecimento.

2.6 Sprint 6

2.6.1 Solução

- Evidência do planejamento:

Nesta disciplina esperava-se um conteúdo complementar para o projeto aplicado. Como implantar a segurança, formas x *frameworks* específicos usados ou recomendados para garantir uma boa qualidade na gestão de ativos, aplicações e também cultura dentro das empresas.

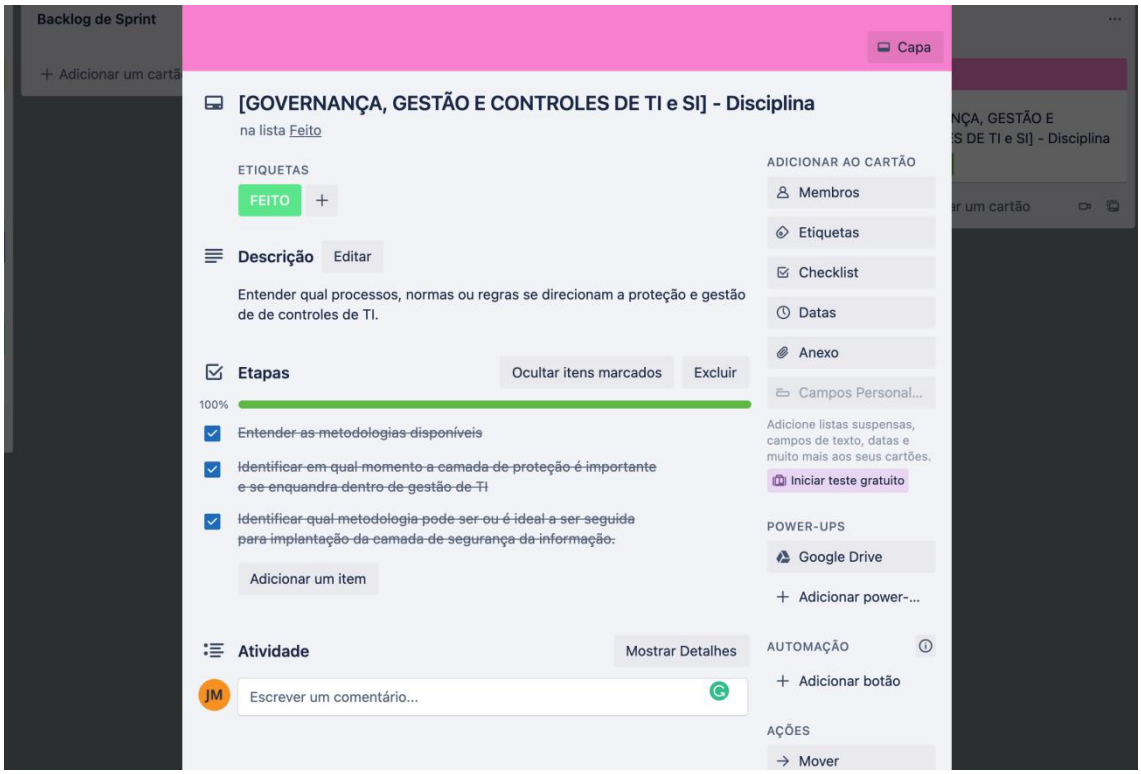


Figura 22 – Projeto de Governança, Gestão e Controle de Ti e SI.
Elaborado pelo autor.

- Evidência da execução de cada requisito:

Como boa prática de gestão de Tecnologia da Informação e Segurança da Informação, existem dois *frameworks* mais usados que podem se

complementar para implantar e seguir boas práticas. Na disciplina foi apresentado o COBIT (*Control Objectives for Information and Related Technology*), que é um *framework* amplamente reconhecido no ambiente corporativo. É um modelo de controle que garante a integridade do sistema da informação, ou seja, uma estrutura criada para governança e gerenciamento de TI. Já o outro *framework* é o ISO 27000, este podemos dividir em três breves camadas, segurança da informação, segurança cibernética e segurança de redes.

Quando não existir uma gestão de Tecnologia da Informação em uma empresa, alguns sintomas da falta desta vai gerar uma série de transtornos e também falta de clareza em definições de papéis e responsabilidades, entre todos os envolvidos. Também pode ocorrer uma má administração de recursos da companhia, não menos importante a falta destes fica claro a não preocupação e também não planejamento de um direcionamento para a elaboração de um plano de continuidade de negócios em casos de qualquer tipo de incidente. Ou seja, uma empresa que não está preparada para enfrentar uma crise de ameaça digital e também não conhecer e prezar pelos seus ativos.

Para avançar na implantação de *frameworks* de boas práticas é preciso assegurar o comprometimento da alta gestão de uma empresa. A mesma precisa ter um posicionamento estratégico muito bem definido. Exemplo: Missão e valores. O mapeamento e avaliação dos riscos que acercam, priorizando cada risco e entendendo os mesmos. Assim a elaboração e planejamento para definição de priorização dos processos a serem implantados, melhorados e monitorados.

- Evidência da solução:

O *framework* COBIT pode ser dividido da seguinte forma:

Camada de Governança:

- EDM = Avaliar, Orientar e Monitorar.

Camada de Gestão:

- APO = Alinhar, Planejar e Organizar;
- BAI = Desenvolver, Aquirir e Implementar;
- MEA = Monitorar, Avaliar e Analisar.

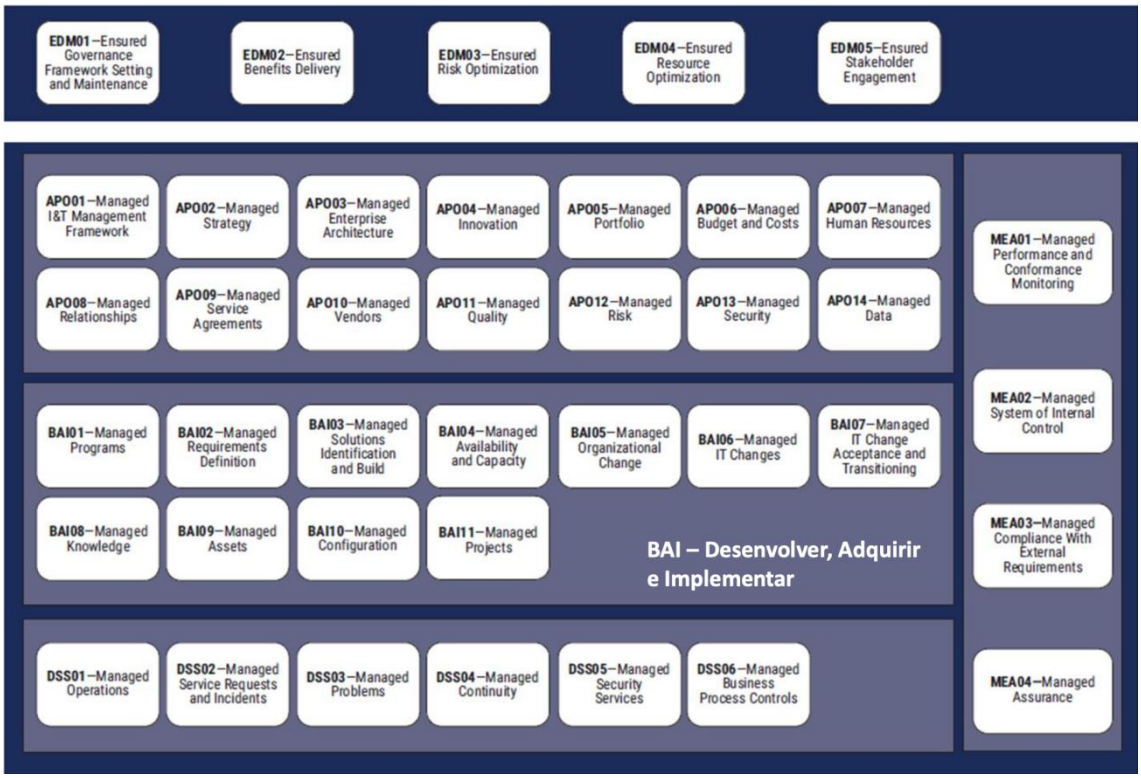


Figura 23 – Visão Geral – COBIT 2019.

Fonte: COBIT 2019 FRAMEWORK: Governance and Management Objectives.

Como esta abordagem acaba sendo bastante ampla, devido a seus vários pontos a serem explorados, observados, pode ser seguindo um pequeno *script* de implantação para adoção que pode ser seguido baseando-se pelo:

- Processo que endereçam a conformidade;
- Baixo esforço e alta percepção (*Quick - Wins*);
- Processos de alto/médio risco x impacto;
- Processos de médio / baixo risco x impacto.

Pontos que são aplicáveis do COBIT a ISO 2700/1/2.

- EDM: garantia, definição e manutenção do modelo de governança (01) e otimização dos riscos (03);
- APO: administrar a estrutura de gestão de TI (01), gerenciar acordos de serviços, gerenciar fornecedores (10), gerenciar riscos (12), gerenciar segurança da informação (13), gerenciar dados (14).
- DSS: gerenciar operações (01), gerenciar incidentes e requisições de serviços (02), gerenciar problemas (03), gerenciar continuidade (04), gerenciar serviços de segurança (05), gerenciar controles processos negócios (06).
- BAI: identificação e construção das soluções (03), gerenciar ativos de TI (09).
- MEA: monitorar, avaliar e analisar o desempenho e conformidade (01), sistema controle interno (02), conformidade com os requisitos externos (03), revisões independentes (04).

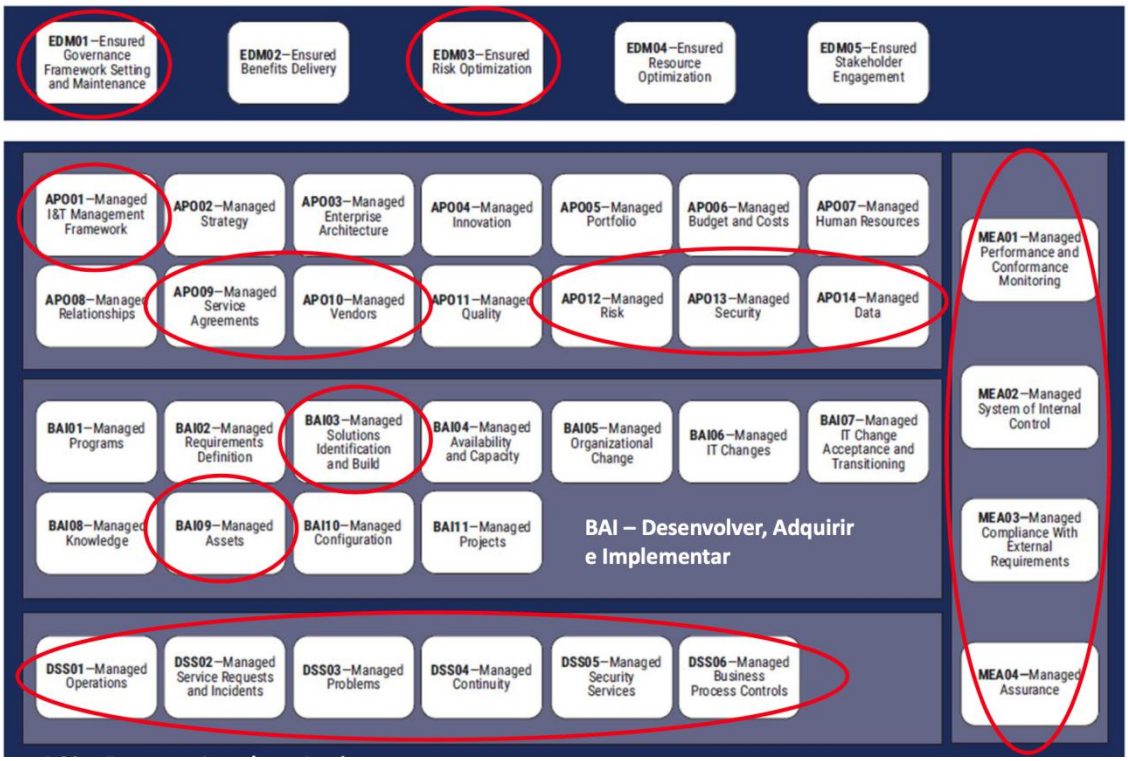


Figura 24 – Visão Geral – COBIT 2019 x Marcação dos Frameworks aplicáveis à SegInfo.

Fonte: COBIT 2019 FRAMEWORK: Governance and Management Objectives.

A ISO 27001 é composta por dois componentes, sendo eles:

- Primeiro Componente: Definição das regras e os requisitos de cumprimento da norma;
  - Contexto da Organização;
  - Liderança;
  - Planejamento;
  - Suporte;
  - Operação;
  - Avaliação de desempenho;
  - Melhorias.

- Segundo Componente: Composto por 14 seções de controles, 35 objetivos de controle e 114 controles. Figura abaixo apresenta um resumo do conteúdo.

Seção	Descrição Seção de Controle	OC	IC
A.5	Políticas de segurança da informação	1	2
A.6	Organização da segurança da informação	2	7
A.7	Segurança em recursos humanos	3	6
A.8	Gestão de ativos	3	10
A.9	Controle de acesso	4	14
A.10	Criptografia	1	2
A.11	Segurança física e do ambiente	2	15
A.12	Segurança nas operações	7	14
A.13	Segurança nas comunicações	2	7
A.14	Aquisição, desenvolvimento e manutenção de sistemas	3	13
A.15	Relacionamento na cadeia de suprimento	2	5
A.16	Gestão de incidentes de segurança da informação	1	7
A.17	Aspectos da segurança da informação na gestão da continuidade do negócio	2	4
A.18	Conformidade	2	8
14		**35**	**114**

Figura 25 – Visão Geral – Resumo do Conteúdo da ISO27001.

Fonte: Apostila GCC – Curso de Segurança Cibernética.

Legenda:

- SC - Seção de Controle;
- OC - Objetivo de Controle;
- IC - Item de Controle.

2.6.2 Lições aprendidas

A importância de entender o objetivo de uma empresa, elaborar um planejamento estratégico de governança desta, desencadeia uma série de situações que podem tornar um negócio mais maduro para enfrentar qualquer tipo de problema. Quando falamos em tecnologia, todo um planejamento bem estruturado sobre como e onde se quer chegar, faz toda diferença nas prioridades e metodologias a serem usadas e monitoradas.

Nesta disciplina conhecemos duas metodologias muito conhecidas no mundo corporativo, também metodologias muito completas, que significa o uso destas em modelos de negócios pequenas, médios e grandes. Tanto COBIT quando ISO 27000 são dicas de como devemos pensar, planejar, monitorar cenários para que seja possível um preparo ou conservação de problemas. Fato é que ambos *frameworks* são extensos e bem detalhistas, porém isso da oportunidade de usa-los sempre de acordo com cada momento, cenário ou negócio.

Visualizando este conteúdo e trazendo o mesmo para este projeto aplicado, podemos concluir que, toda e qualquer ferramenta que tem por objetivo auxiliar em atividades de monitoramento, informações ou até mesmo resoluções de problemas mais pontuais são sempre muito bem-vindas, uma vez que o foco de uma boa gestão é entender os riscos, atuar para IoT diminui sempre que possível e atuar de modo mais preciso para evitar problemas mais agravantes.

2.7 Sprint 7

2.7.1 Solução

- Evidência do planejamento:

Nesta disciplina esperava-se um conteúdo complementar para o projeto aplicado. Como conhecer boas práticas de segurança em aplicações. Observando formas, *frameworks* e metodologias. Assim podendo conhecer para implantar camadas básicas e adicionais de segurança em qualquer aplicação.

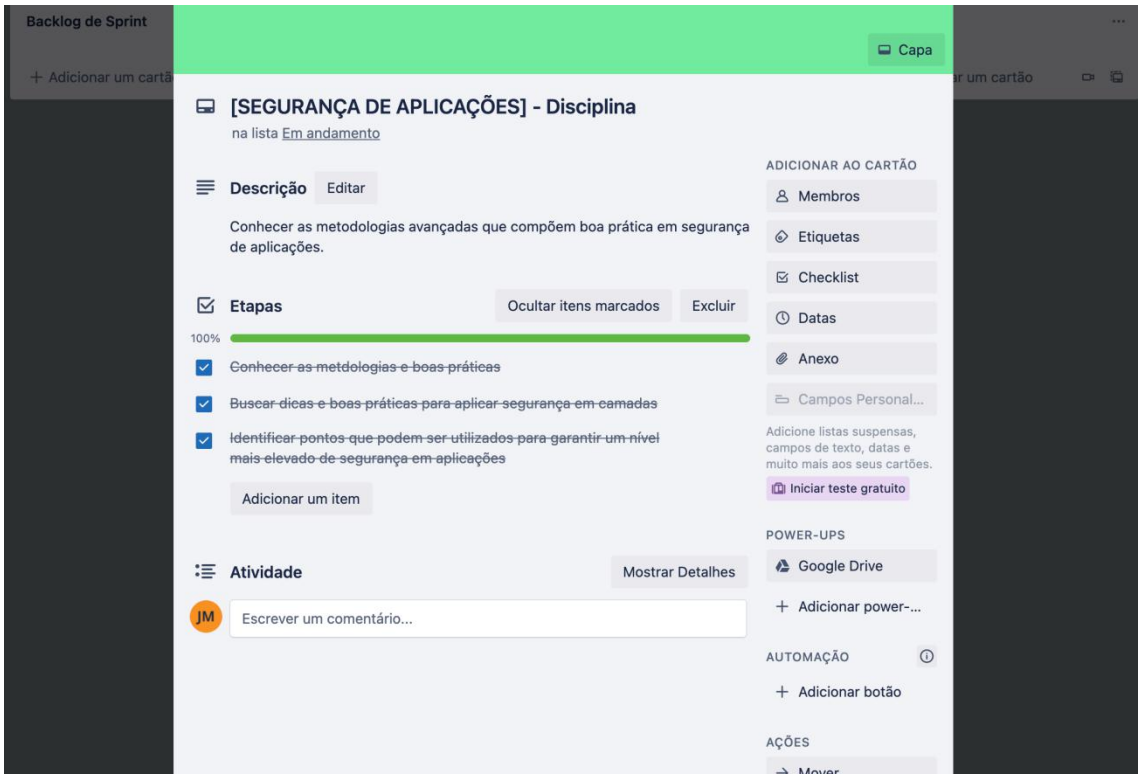


Figura 26 – Segurança de Aplicações.
Elaborado pelo autor.

- Evidência da execução de cada requisito:

Para agregar camadas de segurança em nível de aplicações, existem várias etapas que podem ser observadas e exploradas. Sejam elas: no time de desenvolvimento, onde envolve o desenvolvimento das aplicações ou na camada de segurança WEB, onde se encontram os protocolos e acontece as requisições e respostas.

As falhas que ocorrem diretamente nas aplicações muitas vezes são consequências de falta de planejamento, investimentos, conhecimentos e também a ignorância de inserir e não priorizar as bases de arquiteturas corretas para fins de entregar produtos com prazos curtos ocultando totalmente qualquer ponto direcionado a buscar estabilidades de seus produtos.

Na ISO 27002 existe uma seção que é voltada para aquisição, manutenção e desenvolvimento de *softwares*. Também não menos importante existe uma metodologia chamada OWASP que direciona boas práticas sobre vários pontos direcionados a segurança para arquiteturas WEB.

Algumas técnicas de exploração de vulnerabilidades podem ser usadas para validar aplicações quando as mesmas se encontram em fases de desenvolvimento, testes e mesmo em produção. Estas ferramentas podem auxiliar um produto a sempre manter seu padrão de segurança um passo a mais sobre o aspecto proteção. Algumas destas técnicas foram conhecidas. Pontos que podem ser levados como consideração, apenas aplicar boas práticas pode não ser o suficiente se tecnologias podem ser ou estar comprometidas, por isso a exploração de testes é um ponto muito positivo.

- Evidência da solução:

STRIDER: O mesmo foi desenvolvido por Michael Howard e David Le Blank, tendo como objetivo fornecer um mnemmônico para ameaças de segurança. Usando estas categorias é possivel trabalhar com diversas situações pontuais onde diversas prioridades podem se sobrepor. Seguem abaixo descritos as categorias:

- *Spoofing* (Falsificação): falsificação de mensagem.
- *Tempering* (Adulteração): alteração de dados durante qualquer transmissão.
- *Repudiation* (repúdio): executa uma ação e nega a mesma.
- *Information disclosure* (vazamento de informações): expor informações não autorizadas.
- *Denial of Service* (negação de serviço): indisponibiliza o sistema - “tira do ar”.
- *Elevation of privilege* (elevação de privilégios): escala privilégios de usuários sem demais autorizações.

Já a OWASP tem o objetivo principal de direcionar de forma mais educacional os desenvolvedores, designers e arquitetos para situações pontuadas a vulnerabilidades direcionadas em aplicações de arquitetura WEB. Nesta são citadas pontos como a prevenção de injeção, falhas de autenticação, exposição de dados sensíveis, controle de acesso ineficiente, má configuração de segurança e também modelos de ataques como *Cross-Site Scripting* (XSS).

Outro ponto citado importante que soma-se a dicas de um ciclo para implantar seria, implantar um ciclo de desenvolvimento seguro, que a grande diferença deste ciclo é implantar juntamente com um ciclo tradicional etapas direcionadas a segurança da aplicação. Importante ressaltar que estas etapas ocorrem de modo paralelo com o ciclo tradicional. Para mais claro abaixo segue representado pela cor laranjado estas etapas adicionais.

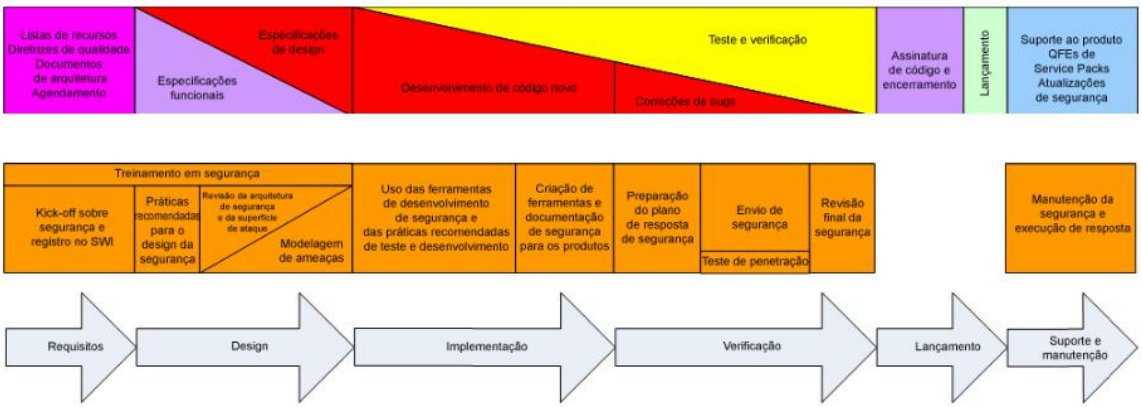


Figura 27 – Ciclo de desenvolvimento Seguro.

Fonte: Apostila Segurança Aplicada IGTI– Professor Fernando Fonseca.

2.7.2 Lições aprendidas

Muitas vezes na falta de conhecimento ou também na falta de planejamento ações acabam sendo totalmente confundidas e assim ficam de lado quando deveriam ser priorizadas. Em segurança sejam de aplicações ou ambientes de arquiteturas, as preocupações são atenuadas só apenas quando os problemas acontecem. Esta disciplina apresentou algumas metodologias direcionadas para pontos e aplicações visando a melhor prática assim somar na camada de segurança.

Visualizando todo este conteúdo e trazendo o mesmo para este projeto aplicado, podemos concluir que, é preciso muito seguir um planejamento estratégico de desenvolvimento, incluíndo neste a importância de adicionar ao escopo o cuidado e boas práticas de desenvolvimento. Existem muito pontos críticos que devem ser tratados como fatores fundamentais para entregar o máximo de qualidade em aplicações. Entre estes fatores podemos citar a segurança das extremidades e do funcionamento das aplicações independentemente de sua arquitetura.

2.8 Sprint 8

2.8.1 Solução

- Evidência do planejamento:

Disciplina que esperava-se um conteúdo apoiador para o projeto aplicado. Como conhecer boas práticas de segurança de infraestrutura. Observando formas, metodologias e referências de boas práticas . Para que assim seja possível desenvolver uma estrutura direcionada a um nível elevado de segurança.

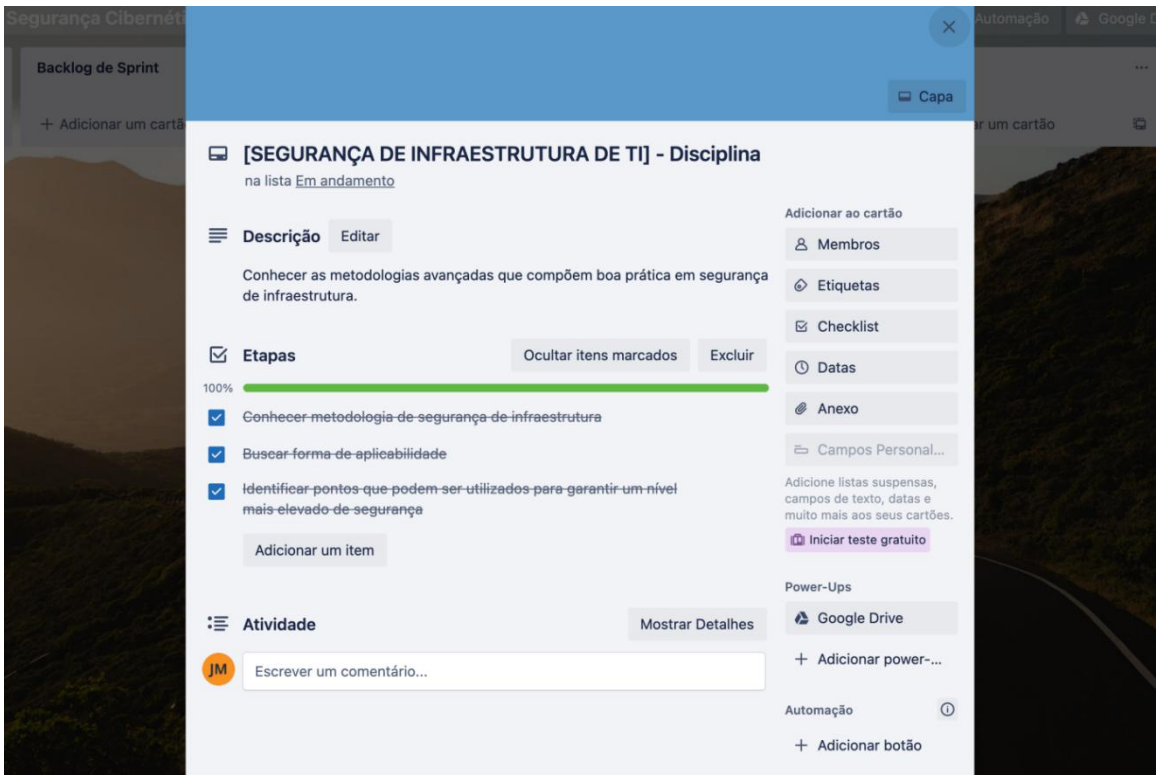


Figura 28 – Segurança de Infraestrutura de TI.

Elaborado pelo autor.

- Evidência da execução de cada requisito:

Como elaborar um cenário seguro visualizando pontos estratégicos em uma organização onde existem muitos atributos técnicos é algo muito complexto, trabalhoso e também custoso. Várias estratégicas podem ser desenhadas para garantir camadas a mais de segurança como um todo. A preocupação com a segurança de um perímetro ou uma organização é algo desafiador, por um lado se tem o alto escalão que limita ou anula recursos para estes investimentos e por outro lado a limitação técnica e de ferramentas para estas ações.

Por onde começar? Está é a pergunta que muitos se fazem, buscando ter uma resposta única, fácil e também simples. Nesta disciplina foi abordado um conceito muito interessante, amplo mas muito bem direcionado. O que de fato é, quando se analisa um ambiente de tecnologia de uma companhia se tem muitos aspectos e variaveis que devem ser analisados, com cautela e precisão.

A estratégia de defesa em profundidade, também conhecida como DiD (*Defense in Deph*) tem como objetivo buscar a eficiencia em um determinado ambiente para que assim seja possível elevar o nível de segurança de ameaças sejam elas oríundas de situações internas ou externas. Esta estratégia é dividida em algumas camadas, o que torna mais fácil de administrar, evoluir, implantar e também entender o objetivo da mesma.

- Evidência da solução:

A estratégia de defesa em profundidade é dividida em três controles e sua arquitetura que é dividida em sete pontos, que são:

Controles:

Controles Físicos: Foco em segurança direcionados a pontos físicos de sistemas.

Controles Técnicos: Foco em proteção contra intrusões, soluções direcionadas a sistemas, como exemplo: Firewall, proxy, antivirus e outros.

Controles Administrativos: Direcionados a medidas que consistem em normas, regras, políticas e procedimentos.

Arquitetura:

Camada Política e Conscientização: Esta camada tem por objetivo desenvolver ações como políticas de segurança, procedimentos e conscientização.

Camada Segurança Física: Já esta camada foca em pontos físicos, como exemplo, cameras de segurança, dispostivos de bloqueios a acessos físicos como leitores de cartões RFID ou biométricos.

Camada Segurança de Perímetro: Direciona a proteção entre o mundo externo e a estrutura interna. Ou seja, um ponto estratégico que é a borda entre as conexões externas e internas.

Camada Segurança rede interna: Tem seus esforços direcionados a pontos e situações que ocorrem internas a uma determinada rede. Assim ferramentas que atuam para este fim são adotadas nesta camada.

Camada Segurança Host: Concentra-se em proteger dispositivos de ponta, como *Laptop*. Os desafios nesta camadas são elevados pois estes dispostivos geralmente possuem alterações diárias, com instalação de novas aplicações, uso em ambientes diversos dentre outros. Assim o foco desta camada é sobre estes dispositivos.

Camada Segurança de Aplicação: Atua em *softwares* que são executados pela companhia, seja ela uma fabricante de *software* ou consumidora da mesma. Aplicações podem possuir diversos pontos de falha, os famosos *bugs*,

e muitas vezes se não exisitir nenhuma preocupação com estes, as brechas podem acontecer e problemas mais agravantes podem surgir.

Camada Segurança de Dados: Foco onde se concentra os dados da companhia. Esforços direcionados a proteção dos dados que são armazenados ou circulados dentro de um determinada estrutura de TI.

Para simplificar algumas ferramentas ou tecnologias direcionadas por camadas, seguem abaixo duas figuras que simplifca e facilita a compreensão desta arquitetura.

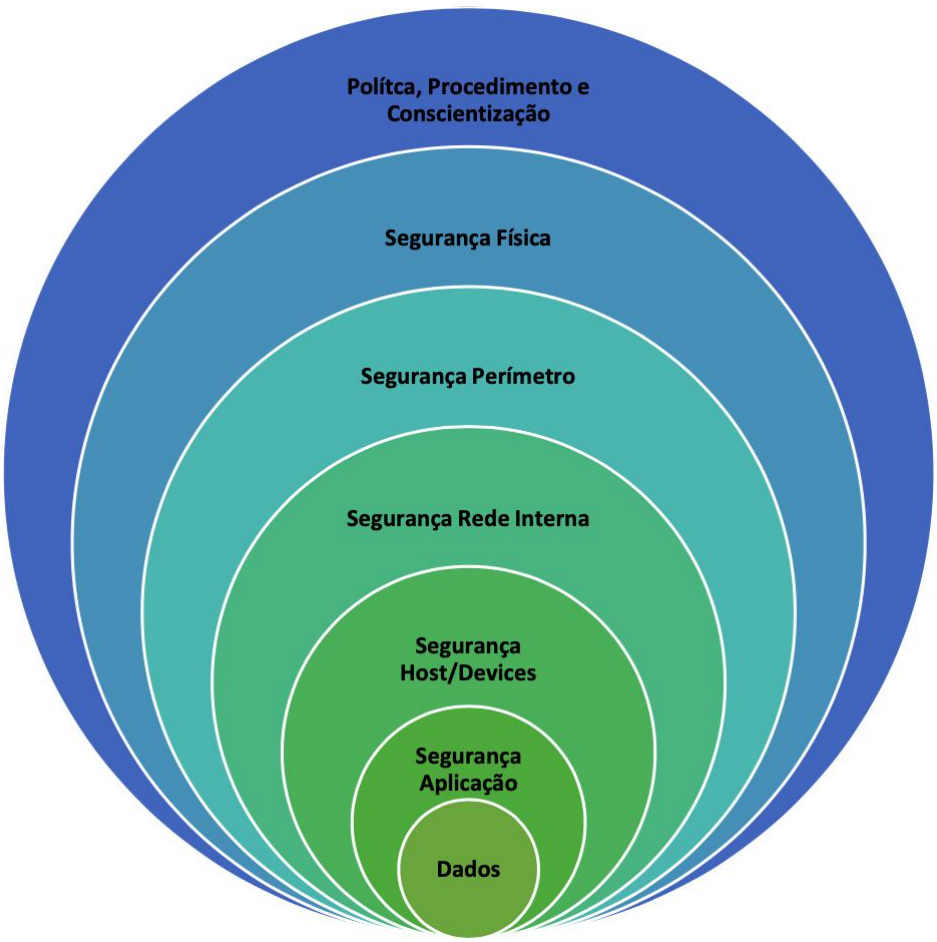


Figura 29 – Arquitetura – Defesa em Profundidade.

Apostila – Segurança de Infraestrutura de TI.

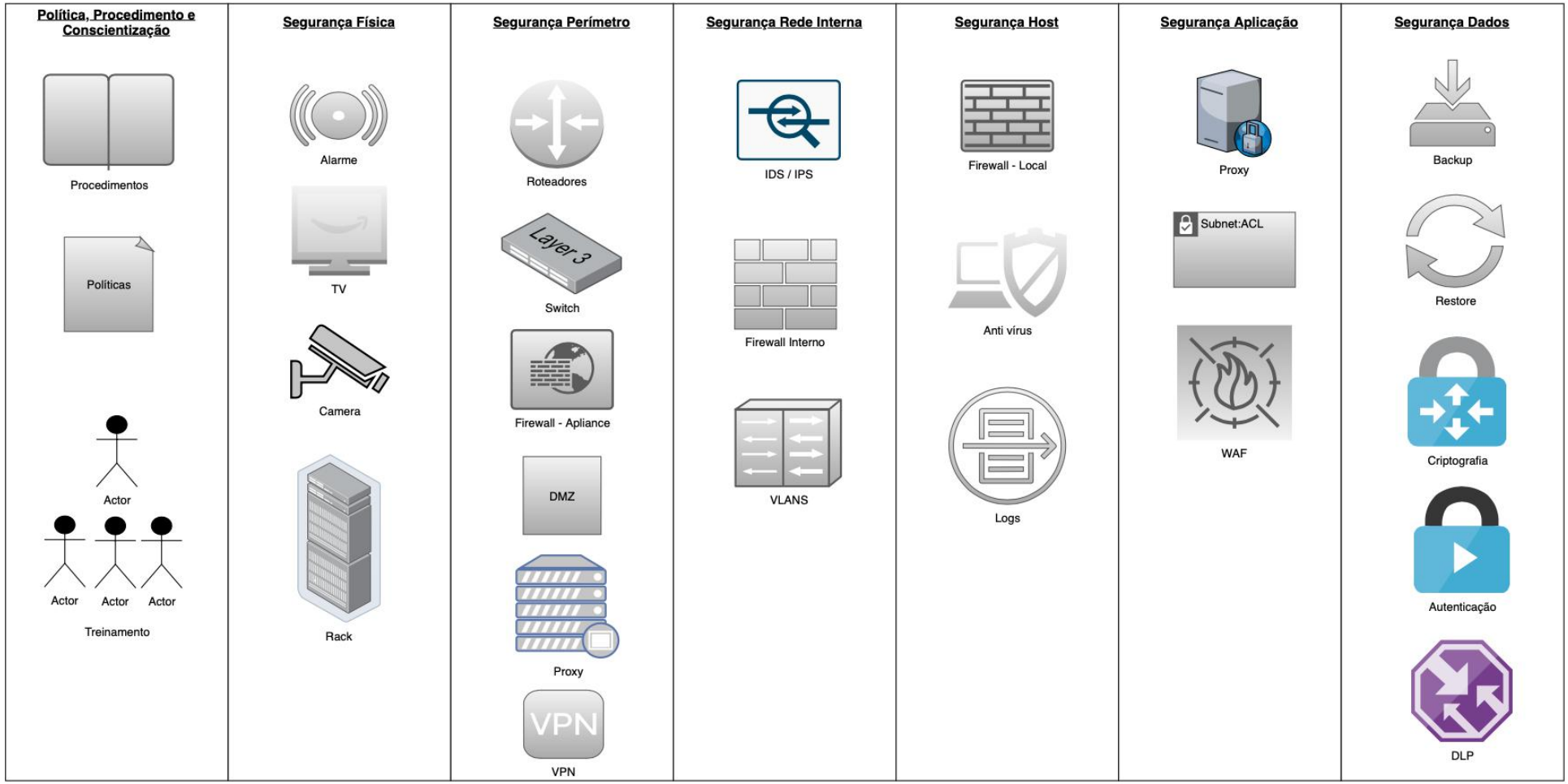


Figura 30 – Arquitetura – Defesa em Profundidade.

Elaborada pelo autor.

2.8.2 Lições aprendidas

Em busca de desenvolver um ambiente seguro muitas vezes se espera a utilização de *softwares* que visam cobrir todos os aspectos complexo de um cenário empresarial ou não relacionado a segurança. Este assunto é muito mais complexo, porém é preciso entender que ações básicas fomentam resultados que se somam a excepcionais *softwares* que são usados para prevenir determinados pontos.

Nesta disciplina podemos conhecer uma arquitetura que divide-se em várias camadas e assim deixa claro o quanto é complexo realizar ações especificas e precisas do aspecto segurança. Porém esta arquitetura demonstra que existem vários angulos que é preciso avaliar para que assim seja possível cobrir a fins de garantir um ambiente mais seguro.

Depois de conhecer e estudar mais a fundo esta arquitetura, podemos definir que o MSIS se enquadra na camada de Segurança de rede interna, onde busca realizar / identificar pontos de seus dispostivos conectados em uma determinada rede, seja está corporativa ou home. Assim podemos definir mais uma ferramenta que poderá ser utilizada de forma estratégica nesta camada.

3. Considerações Finais

3.1 Resultados Finais

Para o desenvolvimento deste trabalho aplicado vários pontos técnicos foram estudados durante as disciplinas que formataram todo o curso de Pós Graduação. Estes agregaram de forma significativa no conhecimento técnico e na estratégia usada para a realização de vários pontos do projeto. O objetivo principal do MSIS foi de desenvolver uma aplicação que como característica seja apoiadora na identificação de equipamentos conectados em uma rede e também identificar as portas que se encontram abertas neste mesmo ambiente. Em várias citações acima já foi mencionada a importância e ponto de apoio que o MSIS terá como aplicação.

Após estudar várias técnicas, compreender situações de redes, realizar testes em ambientes reais, a formatação de uma aplicação direcionada a cumprir com seus requisitos propostos no projeto e também facilitar uma ação de uso, a Figura 31 representa exatamente o menu da aplicação.

Figura 31 – Menu do MSIS.

Elaborada pelo autor.

A arquitetura da aplicação foi projetada para funcionar visando facilitar detalhes como, a realização da coleta dos dados, arquivamento de seus resultados em um determinado arquivo onde podemos concluir que este vai servir como um banco de dados e assim facilitar as consultas que possam ocorrer sobre este banco de dados.

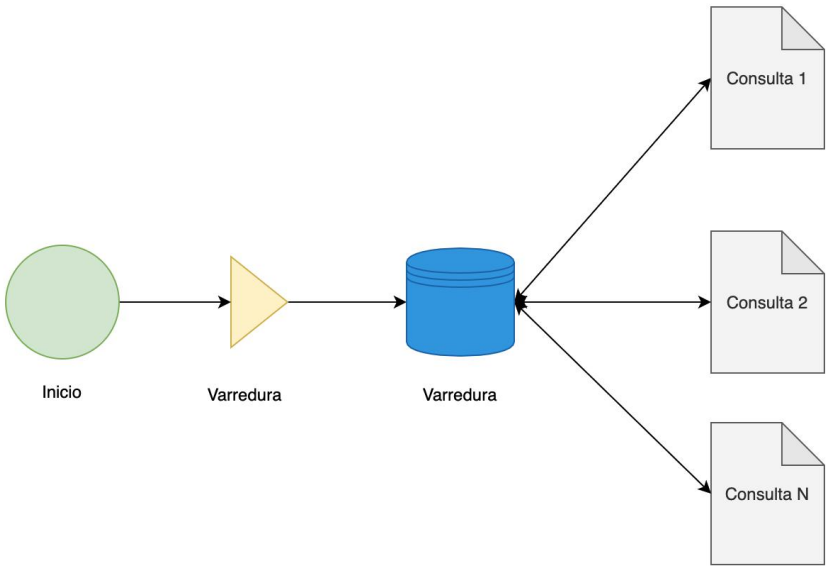


Figura 32 – Arquitetura de funcionamento do MSIS.
Elaborada pelo autor.

Após a elaboração da arquitetura do funcionamento da aplicação sendo a varredura, coleta dos dados, armazenamento e as consultas. A próxima etapa foi focar na organização da aplicação. Onde é realizado a divisão em oito alternativas. Foram classificadas da seguinte forma:

Alternativa 1 = Monitoramento de Redes - ação que inicia a varredura completa em uma determinada rede. A função utilizada para realizar esta varredura é baseada sobre a ferramenta nmap. Os atributos utilizados para que a varredura busque dados qual é o proposto pelo MSIS são:

- Estrutura 01: nmap -sVS -O [RANGE_IP_VARREDURA]
- Estrutura 02: > arquivo_sistema
- Resultado Final: nmap -sVS -O [RANGE_IP_VARREDURA] > arquivo_sistema

```
print ("=========================================")
print ("\033[05;31mVarredura de Rede...\033[00;37m")
os.system ('nmap -sVS -O 192.168.10.0/24 > arquivo_sistema')
print ("=========================================")
```

Figura 33 – Alternativa 01 – Comando de varredura na rede e arquivamento de dados.
Elaborada pelo autor.

Na estrutura 01 - o objetivo do NMAP é realizar a varredura e capturar informações como os serviços sendo executados, versão dos mesmos e também os sistemas operacionais quais identificados. Os serviços que estão sendo executados são identificados pelas portas que estarão abertas em cada dispositivos conectados a rede.

Na estrutura 02 - o objetivo é pegar todo o resultado da estrutura 01 e salvar no arquivo qual foi denominado de arquivo_sistema.

Alternativa 2 - Esta alternativa já atua sobre a consulta dos dados coletados na alternativa anterior, ou seja, onde já temos a varredura finalizada, os dados arquivados e agora é só realizar a consulta no banco de dados e apresentar todos os endereços de IP que apresentaram retorno de resposta. Para a exploração desta resposta foi utilizado comandos baseados na linguagem *Shell Script*.

```
print ("=========================================")
print ("opcao 2- \033[05;31mIPS - Descobertos\033[00;37m")
os.system ('cat arquivo_sistema | grep "Nmap scan" | cut -d \' \' -f5')
print ("=========================================")
```

Figura 34 – Alternativa 02 – Comando de leitura do bando de dados (arquivo) e filtro sobre o retorno de encontro do endereço de IP.
Elaborada pelo autor.

Exemplo de resposta adquirido no processo de captura realizado em uma determinada rede:

```
=========================================
opcao 2- IPS - Descobertos
192.168.20.1
192.168.20.100
192.168.20.101
192.168.20.102
192.168.20.103
192.168.20.104
192.168.20.105
192.168.20.107
=========================================
...
Pressione ENTER para retornar ao MENU
```

Figura 35 – Tela do MSIS – Menu – Opção 2 – IPs descobertos na rede após varredura.
Elaborada pelo autor.

Alternativa 3 - Também atua como consulta sobre a alternativa 1. As respostas que são apresentadas nesta opção é de todas as portas abertas encontradas na varredura que foram encontradas. Para a exploração desta resposta foi utilizado comandos baseados na linguagem Shell Script.

```
print ("=========================================")
print ("\033 Portas Abertas \033")
os.system ('cat arquivo_sistema | grep "open"')
print ("=========================================")
```

Figura 36 – Alternativa 03 – Comando de leitura do bando de dados (arquivo) e filtro sobre o retorno de encontro das portas abertas.
Elaborada pelo autor.

Exemplo de resposta adquirido no processo de captura realizado em uma determinada rede:

```
0/tcp   open  http    TP-LINK WR741ND WAP http config
1900/tcp open  upnp    ipOS upnpd (TP-LINK TL-WR741ND WAP 4.0; UPnP 1.0)
8081/tcp open  blackice-icecap?
6112/tcp open  dtspc?
8089/tcp open  unknown
8081/tcp open  blackice-icecap?
8081/tcp open  blackice-icecap?
1080/tcp open  socks5     (No authentication; connection failed)
8888/tcp open  tcpwrapped
22/tcp open  ssh     OpenSSH 8.1p1 Debian 1 (protocol 2.0)
=========================================
...
Pressione ENTER para retornar ao MENU
```

Figura 37 – Tela do MSIS – Menu – Opção 3 – portas abertas na rede após varredura.
Elaborada pelo autor.

Alternativa 4 - Esta realiza a consulta sobre a base de dados já coletadas anteriormente. As respostas que são apresentadas nesta opção é a apresentação dos sistemas operacionais que foram lidos junto aos dispositivos encontrados na rede. Para a exploração desta resposta foi utilizado comandos baseados na linguagem Shell Script.

```
print ("=========================================")
print ("\033 Sistemas Operacionais descobertas \033")
os.system ('cat arquivo_sistema | grep "OS CPE:"')
print ("=========================================")
```

Figura 38 – Alternativa 04 – Comando de leitura do bando de dados (arquivo) e filtro sobre o retorno de encontro dos sistemas operacionais.
Elaborada pelo autor.

Exemplo de resposta adquirido no processo de captura realizado em uma determinada rede:

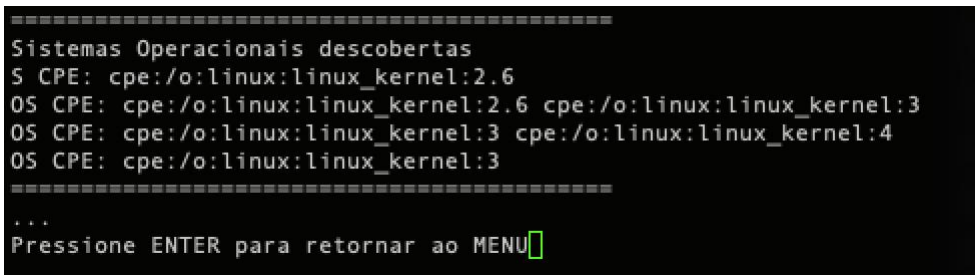


Figura 39 – Tela do MSIS – Menu – Opção 4 – sistemas operacionais dos dispositivos conectados na rede.
Elaborada pelo autor.

Alternativa 5 - Também realiza a consulta sobre a base de dados já coletadas anteriormente. As respostas que são apresentadas nesta opção é a apresentação dos MACs que foram lidos juntos aos dispositivos encontrados na rede. Para a exploração desta resposta foi utilizado comandos baseados na linguagem *Shell Script*.

```
print ("========================================")
print ("\033 MAC \033")
os.system ('cat arquivo_sistema | grep "MAC Address:"')
print ("========================================")
```

Figura 40 – Alternativa 05 – Comando de leitura do bando de dados (arquivo) e filtro sobre o retorno de encontro dos MACs.
Elaborada pelo autor.

Exemplo de resposta adquirido no processo de captura realizado em uma determinada rede:

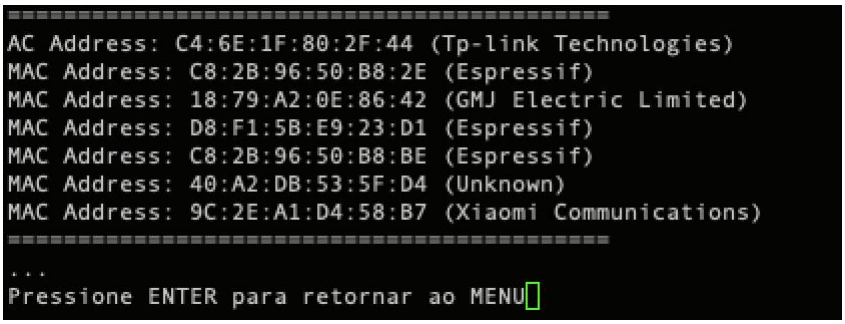


Figura 41 – Tela do MSIS – Menu – Opção 5 – MAC Address dos dispositivos conectados na rede.
Elaborada pelo autor.

Alternativa 6 e 7 - Ambas vão atuar na realização de consulta também sobre as informações já coletadas. Porém, na alternativa 6 - exibe todo o resultado da consulta e a alternativa 7 - ela realiza um tipo de filtro e apresenta uma gama de informações específicas sobre apenas o endereço de IP que o usuário

escolher. Para ambas explorações destas respostas foram utilizados comandos baseados na linguagem *Shell Script*.

Alternativa 6 - apenas apresenta o arquivo gerado pela varredura de forma simples e objetiva. Comando para este foi:

```
print ("=========================================")
print ("\033A Informacoes completas \033")
os.system ('cat arquivo_sistema')
print ("=========================================")
```

Figura 42 – Alternativa 6 – Comando de leitura do bando de dados (arquivo).
Elaborada pelo autor.

Alternativa 7 - finaliza o processo de consultas deixando a escolha do usuário sobre qual endereço de IP o mesmo quer visualizar mais informações.

```
print ("=========================================")
print ("A Informacoes por IP")
ip = input("Digite o IP:")
os.system ('sed -e \'/./{H;$!d;}\' -e \'x;/'+ip+'/!d;\' arquivo_sistema')
print ("=========================================")
```

Figura 43 – Alternativa 7 – Script que possibilita ao usuário a escolha sobre qual endereço de IP o mesmo quer aprofundar sua pesquisa.
Elaborada pelo autor.

Exemplo de resposta adquirido no processo de captura realizado em uma determinada rede:

```
=========================================
A Informacoes por IP
Digite o IP:192.168.20.102

Nmap scan report for 192.168.20.102
Host is up (0.029s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE          VERSION
8081/tcp open  blackice-icecap?
MAC Address: D8:F1:5B:E9:23:D1 (Espressif)
No exact OS matches for host (If you know what OS is running on it, see https
://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=11/6%OT=8081%CT=1%CU=35011%PV=Y%DS=1%DC=D%G=Y%M=D8F15B
OS:%TM=6186C108%P=arm-unknown-linux-gnueabihf)SEQ(SP=95%GCD=1%ISR=CF%TI=I%C
OS:I=I%II=RI%SS=O%TS=U)SEQ(SP=97%GCD=1%ISR=CF%TI=RD%CI=I%II=RI%TS=U)SEQ(SP=
OS:76%GCD=1%ISR=CF%TI=RD%CI=I%TS=U)OPS(O1=M5B4%O2=M5B4%O3=M5B4%O4=M5B4%O5=M
OS:5B4%O6=M5B4)WIN(W1=16D0%W2=16D0%W3=16D0%W4=16D0%W5=16D0%W6=16D0)ECN(R=Y%
OS:DF=Y%T=80%W=16D0%O=M5B4%CC=N%Q=)T1(R=Y%DF=Y%T=80%S=O%A=S+%F=AS%RD=0%Q=)T
OS:2(R=N)T3(R=Y%DF=Y%T=80%W=16D0%S=O%A=S+%F=AS%O=M5B4%RD=0%Q=)T4(R=Y%DF=Y%T
OS:=80%W=16D0%S=A%A=S%F=AR%O=%RD=0%Q=)T5(R=Y%DF=Y%T=80%W=16D0%S=A%A=S+%F=AR
OS:%O=%RD=0%Q=)T6(R=Y%DF=Y%T=80%W=16D0%S=A%A=S%F=AR%O=%RD=0%Q=)T7(R=Y%DF=Y%
OS:T=80%W=16D0%S=A%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=80%IPL=38%UN=0%RIPL=G
OS:%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=S%T=80%CD=S)
=========================================
...
Pressione ENTER para retornar ao MENU
```

Figura 44 – Tela do MSIS – Menu – Opção 7 – Resultado filtrado apenas pelo endereço de IP que foi inserido de forma manual – neste exemplo – 192.168.20.102.
Elaborada pelo autor.

Uma vez que o usuário realizou todas as suas pesquisas, sanou suas dúvidas existem ainda uma forma do mesmo fazer o encerramento da aplicação, bata imputar o código 99 que a aplicação se encerra. A arquitetura foi desenvolvida sobre a linguagem *Python* e *Shell Script*.

Os testes que foram executados para entender o funcionamento, realizar a leitura dos dados apresentados neste projeto, foram extraídos de uma rede *home office*, porém em uma VLAN onde só são executados dispositivos de Internet das Coisas (IoT). Ou seja, um local que possuí "n" equipamentos conectados na internet sobre uma rede apenas para os mesmos se comunicarem. Para facilitar a compreensão deste cenário a Figura 45 tem por objetivo deixar mais clara as distribuições dos equipamentos distribuídos na rede.

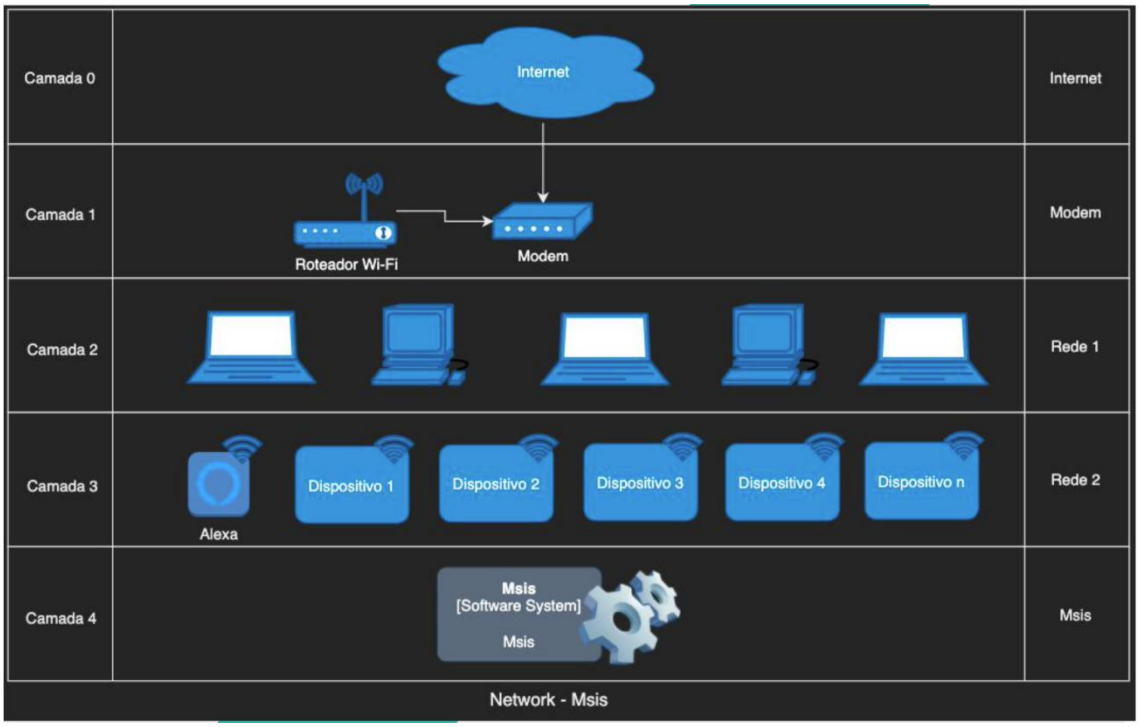


Figura 45 – Rede de execução dos testes.

Elaborada pelo autor.

Na Camada 0 temos a conexão com a internet, evoluindo para a Camada 1 o ponto que fica o modem de internet mais o roteador, neste caso dois dispositivos diferentes. A Camada 2 é uma camada da rede isolada onde é conectado apenas dispositivos como computadores (*desktop* ou *laptop*) - sua base é sobre o range 192.168.10/24. Já na Camada 3 é nova rede isolada que fica localizado todos os dispositivos de internet das coisas conectados na internet. Nesta classificação temos Alexa, sensores de luz, TV, Xbox 360 e *Raspberry* PI. Seu range é 192.168.20/24. A Camada 4 foi exibida na Figura

para fins de demonstração uma vez que o MSIS vai atuar, ou seja, o usuário pode por livre escolha executar o mesmo em qualquer VLAN desejada.

Pontos Positivos:

O conhecimento adquerido em várias disciplinas pontuam como bagagem para o entendimento de ferramentas, tecnologias e arquiteturas que foram de extrema importância para a elaboração da estratégia e também execução do projeto. Não menos importante a satisfação de elaborar o projeto e como resultado poder visualizar tais detalhes que chamaram muita a atenção como porque porta X, Y ou Z se encontram abertas?

A facilidade de executar a aplicação, poder de forma objetiva identificar todos os dispositivos conectados na mesma, intender de forma objetiva quais dispositivos estão respondendo, analisar seus sistemas operacionais e assim poder concluir determinadas ações, fazem com que este trabalho agrega de forma pontual e objetiva para quem visa a segurança e entender melhor o comportamento de seus componentes em sua rede.

Pontos Negativos:

Sobre o aspecto tempo muitas ações e estratégias tiveram que ser priorizadas de modo a ganhar-se tempo, ou seja, vários pontos que podem ser melhorados no projeto ficam com cenários de melhorias. Existem muitas ferramentas que podem agregar ainda mais nas varreduras, porém a limitação de tempo foi fator determinante para focar em trazer um resultado mais preciso e objetivo neste ponto do projeto.

Algumas disciplinas do curso direcionaram sobre algum aspecto, mas teve detalhes que faltaram na disciplina e que foi buscado de forma independente em fontes externas. Talvez se pontos mais técnicos tivessem sido focados nortes diferentes ou até projeções distintas poderiam auxiliar para um fator de tomada de decisão em determinados contextos.

O projeto não foi tratado para ser executado em qualquer arquitetura computacional, ou seja, acaba sendo um limitador para o uso do mesmo. Também é um fator que pode ser aplicado como melhoria do mesmo. As consultas neste momento podem ser realizadas pela própria aplicação ou de forma manual sobre um terminal e entendendo e usando os comandos corretos.

Dificuldades enfrentadas:

Pontos como usar determinada ferramenta de apoio para realizar determinadas funções, este foi um ponto que gerou muitas dúvidas. A projeção de trabalhar com uma base de informações e extrair a mesma de forma que facilita a interpretação das mesmas.

Durante as disciplinas, muitos conteúdos foram direcionados de modo em que as informações eram bastante neutras ou superficiais, isso gerou um trabalho a mais para buscar o alinhamento e compreender se determinada tecnologia, ferramenta se adaptaria ao projeto. Muitos testes foram executados, arquitetura repensada de várias formas.

3.2 Contribuições

Diversas vezes temos por hábito adicionar dispositivos em nossas redes sem ao menos avaliar qual impacto o mesmo pode ocasionar, dispositivos que por padrão de fábrica vem com suas configurações extremamente incorretas para uma prática mínima de segurança. Quando inseridos estes dispositivos para realizar determinadas avaliações é preciso entender sobre um aspecto muito técnico, resultado, ninguém busca entender, simplesmente estes são inseridos.

O MSIS tem este objetivo, ser uma ferramenta simples e prática, que desperte a simplicidade de querer entender como estão configurados os dispositivos e fazer perguntas como:

> Mas será mesmo que este dispositivo precisa de tantas portas abertas?
>
> Porque tem tantos dispositivos conectados em minha rede?

Hoje já existem diversas ferramentas no mercado, porém o MSIS tem seu objetivo de ir além, a facilidade de executar o mesmo em rede, ler todas as informações básicas e caso necessidade as mais avançadas, a evolução deste é cruzar todos estes dados coletados com fontes externas para que quem esteja analisando os mesmos consiga tomar as melhores decisões sempre visando proteger seus ativos e sua segurança em seu ambiente.

3.3 Próximos passos

Desenvolver no MSIS mecanismos de poder visualizar informações relacionadas as boas práticas. Como base de pesquisa utilizar as informações coletadas para realizar pesquisas em fontes seguras a fim de entender se alguns pontos dos dados coletados apresentam ou indicam contextos de riscos

que podem atribuir a detalhes agravantes e indicar pontos de riscos a determinado dispositivo ou ambiente.

Também é válido atribuir o desenvolvimento de mais relatórios para visualizar de modo estratégico todos os dados coletados. O seu objetivo é apresentar um resultado adequado a mais situações. Há situações em que pode ser melhorado também mecanismo de realizar a varredura.

Quanto ao fator de deixar o MSIS acessível é desejável a melhoria sobre a sua estrutura. A ferramenta para que seja possível ser executada em modo de contêineres e também a possibilidade de utilização em ambientes de multiplataformas, por exemplo: Windows, Linux ou Mac.

Bibliografia

MAROSTEGA,J.A et al. **MSIS - MODELO DE DEFESA CIBERNÉTICO UTILIZANDO TÉCNICAS DE ATAQUES EM PARALELO A REDES SEM FIO EM TERRITÓRIO SUSPEITO COM VEÍCULO AÉREO NÃO TRIPULADO.** Dissertação de Mestrado - UNISINOS - São Leopoldo RS - p.80. 2020.

KALI – Portal - **The most advanced Penetration Testing Distribution** – 2021 - https://www.kali.org/ - acesso em: maio/2021.

NMAP – Portal **- Nmap Security Scanner –** 2021 **- https://nmap.org/ -** acesso em: abril/2021**.**

VIRTUALBOX - Portal **- Welcome to VirtualBox.org -** 2021 **- https://virtualbox.org/ -** acesso em: fevereiro/2021.

SHELLSCRIPT - Portal **- Shell Scripting Tutorial -** 2021 **- https://shellscript.sh/index.html -** acesso em: abril/2021.

LINDE'N, E. **A latency comparison of iot protocols in mes.** 2017.