UNIVERSIDADE FEDERAL DO CEARÁ

CAMPUS QUIXADÁ

CURSO DE GRADUAÇÃO EM REDES DE COMPUTADORES

VITOR REIEL MOURA DE LIMA

LAB CYBER ACADEMY: APLICAÇÃO PARA AUXILIAR A CRIAÇÃO DE LABORATÓRIOS DE TREINAMENTO CYBER RANGE

QUIXADÁ

2023

VITOR REIEL MOURA DE LIMA

LAB CYBER ACADEMY: APLICAÇÃO PARA AUXILIAR A CRIAÇÃO DE LABORATÓRIOS DE TREINAMENTO CYBER RANGE

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Redes de Computadores do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de tecnólogo em Redes de Computadores.

Orientador: Prof. Dr. João Marcelo Uchôa de Alencar.
Coorientador: Prof. Dr. Michel Sales Bonfim.

QUIXADÁ

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

L711l Lima, Vitor Reiel Moura de.
Lab Cyber Academy : aplicação para auxiliar a criação de laboratórios de treinamento cyber range / Vitor Reiel Moura de Lima. – 2023.
61 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Redes de Computadores, Quixadá, 2023.
Orientação: Prof. Dr. João Marcelo Uchôa de Alencar.
Coorientação: Prof. Dr. Michel Sales Bonfim.

1. Cibersegurança. 2. Cyber Range. 3. Docker. 4. Computação em nuvem. 5. Infraestrutura como código. I. Título.

CDD 004.6

VITOR REIEL MOURA DE LIMA

LAB CYBER ACADEMY: APLICAÇÃO PARA AUXILIAR A CRIAÇÃO DE LABORATÓRIOS DE TREINAMENTO CYBER RANGE

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Redes de Computadores do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de tecnólogo em Redes de Computadores.

Aprovada em: ___/___/____.

BANCA EXAMINADORA

Prof. Dr. João Marcelo Uchôa de Alencar (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Me. Marcos Dantas Ortiz
Universidade Federal do Ceará (UFC)

Prof. Me. Roberto Cabral Rabêlo Filho
Universidade Federal do Ceará (UFC)

Aos que acreditaram em mim, meu sincero agradecimento a Deus, aos meus pais, familiares e amigos. Muito obrigado.

AGRADECIMENTOS

Agradeço a Deus por estar me acompanhando nesse processo e por ter me dado forças para continuar. Não foi fácil, de fato. Em nenhum momento deixei de enfrentar dificuldades e obstáculos. Houveram muitas noites em que não consegui dormir, mas mesmo assim, nunca desisti. Por mais cansado e exausto que tenha estado, nunca desisti e pretendo continuar com essa ideia, seja onde for.

Agradeço à minha mãe, Roberlania Moura da Silva, que é a mulher mais incrível, forte e perfeita que conheço. Nunca deixou de me incentivar, nunca deixou que eu desistisse. Sempre me deu forças, sempre ficou ao meu lado, sempre esteve comigo quando eu mais precisei. Tudo que eu sou hoje é graças a você, e eu sou eternamente grato por tudo que você já fez e faz por mim. Tenho muito orgulho de ser seu filho. Minha rainha, você sempre será minha inspiração para continuar.

Agradeço ao meu pai, José Luciano de Lima, o homem mais incrível, engraçado e perfeito que conheço. Nunca deixou de me incentivar e sempre me fez rir nos momentos em que mais desejava desabar. Sou muito grato por tudo que me ensinou, admiro seu caráter e a forma como consegue resolver os problemas mais complicados de cabeça erguida. Sou eternamente grato por tudo que você já fez e faz por mim. Tenho muito orgulho de ser seu filho. Meu rei, você sempre será minha inspiração para continuar.

Agradeço à minha namorada por ficar ao meu lado. Aos meus dois irmãos de outra mãe, com quem dividia moradia durante a graduação. Aos meus familiares, amigos, colegas, professores e a todos que ao longo dessa jornada me apoiaram, ajudaram e contribuíram de alguma forma para o meu desenvolvimento, tanto como aluno quanto como pessoa.

Agradeço aos meus orientadores, Dr. João Marcelo Uchôa Alencar e Dr. Michel Sales Bonfim, por todos os conselhos, incentivos, correções, ideias e alternativas que vocês me apresentaram, além da paciência que tiveram comigo. Sou muito honrado em ser orientado por vocês dois, pois são os professores a quem tenho uma forte admiração e respeito.

A todos o meu mais sincero obrigado.

"Cada vez mais, nos sentimos tão seguros envoltos em nossas bolhas que só aceitamos informações, verídicas ou não, que se enquadram com nossas opiniões, em vez de basear nossas opiniões nas evidências existentes."

(Barack Obama)

RESUMO

O cenário atual da cibersegurança é marcado por um aumento constante de ataques cibernéticos em todas as áreas da computação. Nesse contexto, as empresas necessitam de profissionais treinados e capacitados para proteger suas organizações contra ameaças virtuais. Uma das abordagens para treinamento em cibersegurança é o uso de *Cyber Range*, ambientes virtuais onde profissionais podem simular e treinar para se defender contra ameaças reais. No entanto, esses treinamentos podem ter um alto custo financeiro. Embora existam ferramentas gratuitas disponíveis como é o caso do CyRM, a configuração e execução de cenários nesses ambientes são frequentemente manuais e complexas, o que pode desencorajar novos usuários. Para superar esses desafios, foi proposta a criação de uma ferramenta gratuita implementada na nuvem que visa facilitar e automatizar a criação de cenários de treinamento em cibersegurança. Essa ferramenta também disponibiliza uma aplicação de fácil acesso e interação com o ambiente de treinamento do CyRM. O experimento foi conduzido na Universidade Federal do Ceará, no campus de Quixadá, e fez uso de ferramentas de infraestrutura como código, como o *Ansible*, contêineres *Docker* e o programa *AWS Academy*. Os alunos participaram do treinamento, respondendo a um questionário guiado no *Moodle*. Os resultados indicaram satisfação por parte dos participantes. Além disso, foi preenchido um questionário no *Google Forms* para avaliar a viabilidade do uso da ferramenta em outras disciplinas de segurança. Esse modelo de treinamento foi realizado pela primeira vez no Campus, direcionado a professores e estudantes da disciplina de Segurança da Informação. O resultado final desse trabalho foi a implementação bem-sucedida do *Lab Cyber Academy* e a validação da solução como uma abordagem eficaz para o treinamento em cibersegurança.

Palavras-chave: Cibersegurança; Cyber Range; Docker; Computação em nuvem; Infraestrutura como código; Treinamento.

ABSTRACT

The current cybersecurity landscape is characterized by a constant increase in cyberattacks across all areas of computing. In this context, companies require trained and skilled professionals to protect their organizations against virtual threats. One approach to cybersecurity training is the use of Cyber Range, virtual environments where professionals can simulate and train to defend themselves against real threats. However, these training programs can come with a high financial cost. While free tools are available, such as CyRM, the configuration and execution of scenarios in these environments are often manual and complex, which can discourage new users. To overcome these challenges, the creation of a free cloud-based tool was proposed to facilitate and automate the creation of cybersecurity training scenarios. This tool also provides an easily accessible application for interaction with the CyRM training environment. The experiment was conducted at the Federal University of Ceará, at the Quixadá campus, and utilized infrastructure as code tools, such as Ansible, Docker containers, and the AWS Academy program. Students participated in the training by responding to a guided questionnaire in Moodle. The results indicated satisfaction among the participants. Additionally, a questionnaire in Google Forms was filled out to assess the feasibility of using the tool in other security-related disciplines. This training model was conducted for the first time on the campus, targeting teachers and students in the Information Security discipline. The final result of this work was the successful implementation of the Lab Cyber Academy and the validation of the solution as an effective approach to cybersecurity training.

Keywords: Cybersecurity; Cyber Range; Docker; Cloud computing; Infrastructure as code; Training.

LISTA DE ILUSTRAÇÕES

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

AWS	*Amazon Web Services*
CIDR	*Classless Inter-Domain Routing*
CLI	*Command-line Interface*
CPU	*Central Processing Unit*
CR	*Cyber Range*
CyRIS	*Cyber Range Instantiation System*
DMZ	*Demilitarized Zone*
EC2	*Elastic Compute Cloud*
GB	*Gigabyte*
GUI	Interface Gráfica do Usuário
IaC	Infraestrutura como Código
KVM	*Kernel-based Virtual Machine*
LTS	*Long-term support*
RAM	*Random Access Memory*
SI	Segurança da Informação
SO	Sistema Operacional
SSH	*Secure Socket Shell*
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
UFC	Universidade Federal do Ceará
UI	Interface de Usuário
USB	*Universal Serial Bus*
VLAN	*Virtual Local Area Network*
VM	Máquina virtual
VPC	*Virtual Private Cloud*
YAML	*YAML Ain't Markup Language*

SUMÁRIO

1 INTRODUÇÃO

De acordo com estatísticas divulgadas no primeiro semestre do ano de 2022 pelo *FortiGuard Labs*, laboratório de inteligência de ameaças da *Fortinet*[1] que é líder global em soluções amplas, integradas e automatizadas de segurança cibernética, foi possível analisar que o Brasil sofreu cerca de 31,5 bilhões de tentativas de ataques cibernéticos de janeiro a junho, quase o dobro reportado no mesmo período em 2021. E para piorar, um novo relatório revelou cerca de 360 bilhões de ciberataques na América Latina durante todo o ano de 2022, sendo que 106 bilhões de tentativas de ataques foram no Brasil, tornando o Brasil o segundo país que mais sofre ciberataques na América Latina, ficando atrás apenas do México com 187 bilhões (FORTINET, 2022; FORTINET, 2023).

Ainda de acordo com informações divulgadas pela *Fortinet*, um dos fatores que ocasionaram um aumento exorbitante de violações é pela grande escassez de profissionais com habilidades em cibersegurança no mercado (MADDISON, 2023).

Uma forma de preencher as lacunas deixadas pela falta de profissionais especializados em SI, é através do forte incentivo e acompanhamento à treinamentos focados na área de segurança da informação, que sejam capazes de simular cenários de ciberataques reais. Com isso, é possível capacitar qualquer indivíduo que esteja apto à atender a demanda das organizações, e ainda, os mesmos podem servir como ponte para passar esse conhecimento adiante e aprimorar novas equipes.

O *Cyber Range* (CR) é uma plataforma poderosa com foco no treinamento de cibersegurança. Essa ferramenta é capaz de criar ambientes virtualizados e controlados para a simulação ou emulação de cenários reais de ataques cibernéticos, e podem acompanhar exercícios práticos tanto de ataque quanto de defesa.

Existem inúmeras soluções de *Cyber Range*, que podem ter sido desenvolvidas tanto pela academia como pela própria indústria, e geralmente são aplicações de alto custo. Dentre elas, pode ser citado o CyRIS, que tem como papel fundamental fornecer opções de gerenciamento em cenários de ciberataques direcionados para treinamentos de segurança, utilizando o KVM[2] nas máquinas para a construção desses cenários virtualizados (PHAM *et al.*, 2016).

A Universidade Federal do Ceará[3] - Campus Quixadá[4] é um instituição que atual-

[1] https://www.fortinet.com/br
[2] https://www.linux-kvm.org
[3] https://www.ufc.br/
[4] https://www.quixada.ufc.br/

mente abriga seis cursos na área de Tecnologia da Informação e Comunicação (TIC). Dito isso, disciplinas ofertadas como Segurança da Informação e outras correlacionadas, assumem papel importante de capacitar e incentivar os alunos na resolução de possíveis problemas de defesa e segurança. Contudo, o Campus infelizmente não possui grandes reservas de recursos financeiros e computacionais para implantar as soluções mais relevantes existentes no mercado, como é o caso do *Cyber Range*.

Recentemente, o trabalho de Dantas (2022) propôs uma solução de *Cyber Range* para a instituição de Quixadá chamada CyRM, que tem como objetivo criar cenários de treinamento de defesa com a simulação de tráfegos de ataques. O CyRM utiliza o emulador de redes Containernet (trabalha com containers), juntamente com um conjunto de *scripts* para a execução de cenários leves de treinamento. Além disso, ele também usa o *Moodle2*[5] para criação e resolução do exercícios práticos. Contudo, todo o processo de definição e execução do cenário, bem como a interação com os *containers*, é feito via linha de comando e manualmente, o que pode não ser tão atraente principalmente para novos usuários, que ainda estão iniciando em sua capacitação em cibersegurança. Neste cenário, faz-se necessário uma solução para o gerenciamento do CyRM, com o objetivo de automatizar e facilitar todo o processo de treinamento e ampliar o uso do *Cyber Range* dentro da comunidade. Pretendemos posicionar o nosso trabalho nesta problemática.

A utilização de soluções na nuvem oferecem grandes vantagens, principalmente por garantir agilidade durante o desenvolvimento e implementação de novos projetos, e por sua flexibilidade e escalonabilidade de recursos. Com isso, é possível usufruir de recursos computacionais sob demanda, como *hardware*, armazenamento, banco de dados, rede e *software* com facilidade e um baixo custo, pois só é cobrado os recursos que realmente estão sendo utilizados (GOOGLE, 2023).

Dito isso, é designado o programa *AWS Academy*[6] para a construção dos cenários de treinamento na nuvem. Esse programa é estendido da plataforma AWS, com o objetivo de preparar e capacitar estudantes em instituições de ensino superior, fornecendo e incentivando o uso de inúmeros serviços na nuvem, a fim de promover o aprendizado e qualificação dos discentes.

A proposta principal deste trabalho é desenvolver uma aplicação disponibilizada na nuvem com funções de gerenciamento e controle, para facilitar o modo de criação dos cenários

[5] https://moodle2.quixada.ufc.br
[6] https://aws.amazon.com/pt/training/awsacademy/

de aprendizagem com CyRM. A aplicação tem foco na automação da infraestrutura necessária para os cenários, agilizando a criação de instâncias EC2 na *AWS Academy* e suas dependências, por meio de métodos de Infraestrutura como Código (IaC). Essas instâncias serão usadas para armazenar os ambientes de treinamento, exercícios práticos, além da aplicação em si. A aplicação será construída utilizando o *framework React* para o *Front-end* e a linguagem de programação Go (GoLang) para o *Back-end*.

Espera-se ainda que este trabalho possa ser utilizado não só na disciplina de Segurança da Informação na UFC - Campus Quixadá, mas que também possa ser adaptado e utilizado por outras instituições de ensino. O objetivo é facilitar a capacitação e preparação de seus estudantes, a fim de promover futuros profissionais especializados em cibersegurança e no combate de ataques cibernéticos.

1.1 Objetivos

1.1.1 Objetivo Geral

Propor e avaliar uma solução na nuvem para gerenciar e administrar laboratórios *Cyber Range*, com foco no CyRM da Universidade Federal do Ceará - Campus Quixadá.

1.1.2 Objetivo Específicos

1. Definir a infraestrutura para a plataforma *Cyber Range*, juntamente com soluções e ferramentas já existentes;

2. Implementar a solução de *Cyber Range* guiado por um caso de uso;

3. Avaliar a solução proposta através de *feedbacks* obtidos por usuários na UFC - Campus Quixadá.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, serão definidos os principais conceitos que orientaram e auxiliaram na construção deste trabalho.

2.1 Cibersegurança

Cibersegurança é a prática adotada para proteger computadores, servidores, dispositivos móveis, sistemas eletrônicos, redes e dados sensíveis contra ataques maliciosos, também pode ser conhecido como segurança da tecnologia da informação ou segurança de informações eletrônicas. Esta prática pode ser aplicável em inúmeros contextos, que vão desde negócios até computação móvel, e ainda, pode ser dividido em algumas categorias (KASPERSKY, 2023). Abaixo será listado e detalhado algumas práticas divididas por categoria, que se encaixam no contexto de cibersegurança, como:

- Segurança de rede: Uma prática para proteger redes de computadores contra intrusos maliciosos, podendo ser eles invasores direcionados ou simplesmente *malwares* oportunistas;
- Segurança de aplicativos: Uma prática que tem como foco manter o *software* e/ou dispositivos livres de ameaças. Afinal, aplicativos corrompidos podem fornecer livre acesso aos dados sensíveis que o mesmo deveria proteger. Por conta disso, é necessário que a segurança comece desde a fase inicial do projeto, muito antes do programa ou dispositivo ser implantado;
- Segurança de informações: Prática que tem como finalidade, defender a integridade e a privacidade dos dados, tanto em transmissão quanto em armazenamento;
- Segurança operacional: Categoria que visa definir processos e decisões para garantir o tratamento e proteção dos arquivos e dados. Isso implica definir permissões que os usuários podem ter ao acessar uma rede, além de definir os procedimentos que validam onde e como os dados poderão ser armazenados ou compartilhados;
- Recuperação de desastres e continuidade dos negócios: Esta categoria é dividida em dois pontos, o primeiro define como uma organização irá tratar um incidente de cibersegurança, ou qualquer tipo de incidente que venha causar perca de operações ou de dados. A definição das políticas de recuperação e tratamento de desastres orientam como a organização irá efetuar a restauração de suas operações e informações, com isso, é possível retornar à mesma capacidade operacional anterior ao incidente. Além disso, o segundo ponto é o

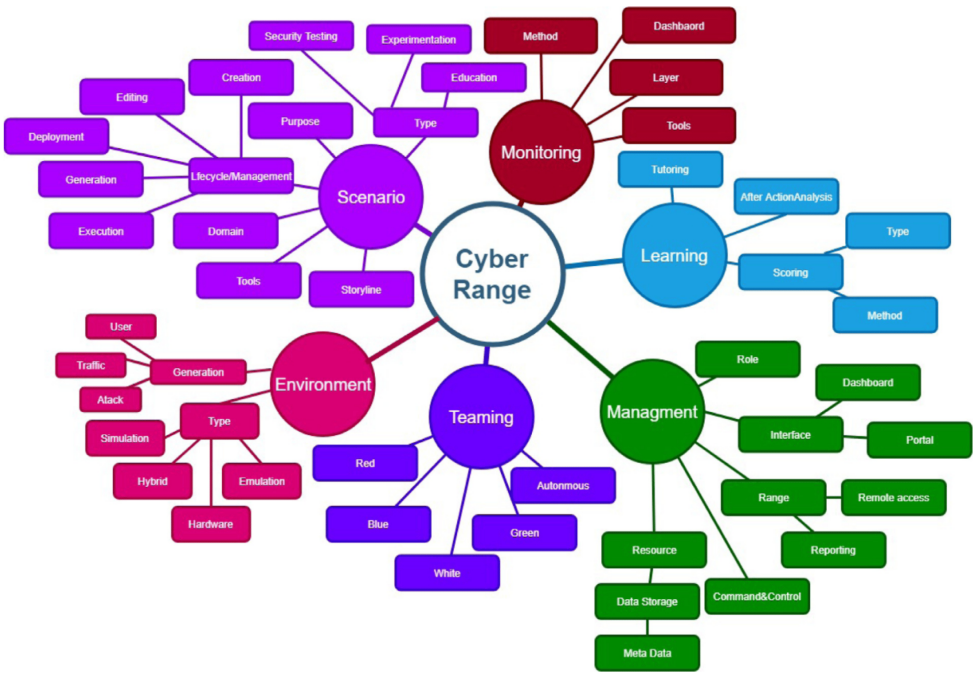
plano na qual a organização recorre para garantir a continuidade dos negócios ao tentar voltar suas operações, sem determinados recursos;

– Educação do usuário final: Categoria que tem como foco as pessoas, já que são consideradas as mais imprevisíveis no contexto de cibersegurança. Afinal, pessoas se não seguirem corretamente as práticas recomendadas de segurança, podem inserir acidentalmente um vírus ou *trojan* em um sistema que se encontrava em segurança. Práticas como instruir o usuário em deletar anexos suspeitos de *e-mails* desconhecidos, não inserir dispositivos USB não identificadas, além de inúmeras outras orientações, são importantes e vitais para garantir a segurança de qualquer organização;

2.2 *Cyber Range*

O *Cyber Range* pode ser definido como uma ferramenta com foco no treinamento de cibersegurança. Essa poderosa plataforma consegue criar ambientes virtualizados e controlados capazes de simular e/ou emular cenários reais de ataques cibernéticos. Geralmente estão acompanhados com exercícios práticos de ataque e defesa. Além disso, também é possível desenvolver *softwares* ou tecnologias relacionadas à segurança. Por ser uma ferramenta usada na criação de laboratórios de treinamento, o *Cyber Range* é uma ferramenta de suma importância para quem deseja se especializar na carreira de segurança da informação ou áreas correlacionadas.

Figura 1 – Taxonomia do *Cyber Range*



Fonte: (YAMIN *et al.*, 2020)

Na Figura 1 é ilustrado a taxonomia dos componentes que podem fazer parte da solução de *Cyber Range*. São eles:

2.2.1 Aprendendo (Learning)

Os usuários participantes são instruídos por um docente, para realizar de exercícios de SI. Os campos para aprendizagem do aluno, são descritos abaixo:

- Tutoria (*Tutoring*): sistema com o papel de realizar e armazenar uma análise aprofundada dos participantes e dos exercícios efetuados;
- Análise Pós-Ação (*After Action Analysis*): é preciso predefinir métricas antes da execução do experimento e aguardar o encerramento do mesmo, para que em seguida seja feito a análise dos dados obtidos. Por fim, os dados experimentais coletados são examinados em um grande conjunto de dados;
- Pontuação (*Scoring*): é utilizado dados interligados ao sistema de monitoramento, que ficam responsáveis por verificar o desempenho dos usuários durante a execução das atividades de SI e até de outros testes de laboratório, ferramentas de segurança, dentre outros. De acordo com a maneira de avaliação definida, existem inúmeros métodos de pontuação que são usados para avaliar os discentes ou as ferramentas para analisar o processo. Como:
  - Tipo (*Type*): tem como foco pontuar os testes e as atividades, através do *software* e/ou *hardware*. As ferramentas inserem funcionalidades para examinar os *logs* e as demais informações obtidas;
  - Método (*Method*): são identificados de acordo com um propósito, como atingir um determinado objetivo ou até examinar registros de *logs* estabelecidos para os exercícios, elaborados para os discentes ou para testes de ferramentas de segurança.

2.2.2 Gestão (Management)

É feito o gerenciamento das funcionalidades relacionadas ao controle de recursos computacionais, que produzem as *Cyber Ranges*. O presente projeto está interligado a este componente, por visar um melhor gerenciamento e controle ao CyRM, usando funcionalidades como a interface gráfica para facilitar a execução de comandos. A função de cada campo deste componente, é descrito a seguir:

- Função (*Role*): é definido por equipes, que podem ter diferentes objetivos como ataques,

defesas, configuração de ambiente, corrigir vulnerabilidades, dentre outros;

- Interface (*Interface*): facilidade no gerenciamento, afinal apresenta de forma gráfica as ações que estão ocorrendo simultaneamente, como processamentos e operações de controles realizadas pelo supervisor;
  - Painel de instrumentos (*Dashbaord*): fornece funções que auxiliam o supervisor realizar instruções de acordo com as necessidades e de maneira gráfica;
  - Entrada (*Portal*): fornece métodos para transmissão de dados com eficácia, além de ser acessível à todos os usuários, compartilhando recursos computacionais geograficamente, acarretando na facilidade do acesso;
- Faixa (*Range*): estabelece o gerenciamento do alcance com acesso remoto. Usa um *proxy* para efetuar a comunicação remota e acessar os dispositivos. Detalhado como:
  - Acesso remoto (*Remote access*): utilizado para acessar os dispositivos conectados em longas distâncias e efetuar determinadas operações, como inicializar ou desligar um sistema, efetuar atividades de instruções e ainda, monitorar em tempo real as atividades que estão em execução;
  - Comunicação (*Reporting*): tem como função descrever de maneira detalhada todo o projeto e seus componentes;
- Controle e Comando (*Command and Control*): É apresentado em forma de interface gráfica, para facilitar a execução de comandos e funções ao sistema, e ainda, pode observar os status das atividades em execução.
- Recursos (*Resource*): Armazena dados de forma autônoma, além disso, utiliza ferramentas como:
  - Armazenamento de Dados (*Data Storage*): apresenta módulos que efetuam armazenamento de dados e configurações do sistema, cenários, ferramentas de ciberataques e de defesa, além de outros componentes como:
    - Meta Dados (*Meta Data*): classificados como dados que se referem a outros dados que possuem base, e efetuam uma coleta resumida de informações referentes aos dados com base;

2.2.3 Equipe (Teaming)

Define funções por cores. Com isso, cada cor representa um time formado por um grupo de pessoas e cada time recebe uma função. Abaixo é detalhado os deveres das cores de

cada time:

- Vermelho (*Red*): o time vermelho tem como dever, atacar e identificar falhas ou vulnerabilidades nos ambientes de treinamento ou de produção;
- Azul (*Blue*): o time azul tem como dever, defender o ambiente corrigindo falhas e vulnerabilidades identificadas pelo time vermelho;
- Branco (*White*): o time branco tem como atribuição, elaborar o ambiente da equipe atacante e da equipe de defesa, definindo regras, funcionalidades, configurações, métodos de avaliação, instruções, além de inserir vulnerabilidades para que os participantes possam encontrá-las e corrigi-las;
- Verde (*Green*): o time verde tem como dever, implementar, monitorar e efetuar manutenções e correções de possíveis *bugs* e travamentos durante o exercício;
- Autônomo (*Autonmous*): o dever desse grupo é automatizar as ferramentas, cenários e outras funcionalidades que serão necessárias, com a finalidade de agilizar e economizar tempo e trabalho;

2.2.4 Ambiente (Environment)

Efetua a execução do cenário e define as ferramentas e *softwares* que serão utilizados para a implementação do laboratório, que podem ser virtualizados ou físicos. Abaixo é descrito os campos que a constituem:

- Geração (*Generation*): tem função de gerar tráfegos e alguns tipos de ataques, ou determinar tipos de usuários, como:
  - Usuário (*User*): o usuário determina as ferramentas que serão usadas no momento de gerar os dados;
  - Tráfego (*Traffic*): o tráfego é simulado em cenários emulados, com o objetivo de fornecer um exemplo mais realista possível, além de oferecer segurança durante a realização dos testes;
  - Atacar ((*Atack*): gera um cenário controlado para ataques, visando elevar ainda mais o nível de realismo, ao mesmo tempo que fornece segurança;
- Tipo (*Type*): campo que estabelece o tipo mais adequado para o ambiente de testes. Abaixo, é detalhado cada tipo que pode ser usado:
  - Simulação (*Simulation*): nesse tipo é realizado testes de aplicações, desempenho, dentre outros. Sem causar prejuízos ao *hardware* ou para o sistema;

  - Híbrido (*Hybrid*): usa dois tipos de ambientes de maneira simultânea, esse tipo efetua uma busca por vantagens de um e o complemento do outro;
  - *Hardware* (*Hardware*): possibilita que as aplicações sejam executadas no próprio *hardware*, podendo ser mais preciso por conta disso. Contudo existe uma maior possibilidade da máquina não corresponder a todos os comandos, o que pode gerar um prejuízo no desempenho;
  - Emulação (*Emulation*): tem como objetivo fornecer um cenário mais próximo possível do sistema e do *hardware* em uso, com isso, acarreta em maior produtividade e economia no geral;

2.2.5 Cenário (Scenario)

Estabelece as etapas do treinamento, definindo o SO, requisitos para o laboratório, as ferramentas e os objetivos. Com isso, é apresentado os campos deste componente com mais detalhes:

- Tipo (*Type*): retorna que os cenários podem ser dinâmicos ou estáticos. Quando definido como estático, não é possível fazer qualquer alteração enquanto a atividade estiver ocorrendo. No dinâmico é feito a inclusão do estático, logo pode ser alterado ao inserir novos componentes durante a execução do cenário, como é o caso de injetar um gerador de tráfego durante a execução de um exercício. Com isso, é evidenciado tipos de cenários do *Cyber Range*:
  - Educação (*Education*): tem como papel definir o roteiro de exercícios para o usuário se aperfeiçoar.
  - Teste de Segurança (*Security Testing*): efetua testes de segurança com o *hardware* e o *software*, em ambientes seguros e controlados;
  - Experimentação (*Experimentation*): efetua testes para validar um experimento, comprovando se o objetivo foi atingindo e se é preciso aperfeiçoá-lo;
- Propósito (*Purpose*): classifica os objetivos usados para validar novas ferramentas ou novos laboratórios;
- Ciclo/Gerenciamento(*Lfecycle/Management*): detalha todas as partes do ciclo de gerenciamento, desde seu inicio até seu encerramento. É descrito cada parte do ciclo abaixo:
  - Criação (*Creation*): ponto inicial onde é feito a criação do laboratório e de suas funções;

  - Edição (*Editing*): caso seja necessário, é possível ser editado;
  - Implementação (*Deployment*): efetua a execução de todas as funções que foram elaboradas e aplicadas;
  - Geração (*Generation*): produz dados da implementação ativa e analisa o seu comportamento geral, para tomar possíveis medidas;
  - Execução (*Execution*): efetiva a execução da aplicação no ambiente, juntamente com todas suas funcionalidades;
- Domínio (*Domain*): apresenta em qual plataforma a aplicação deve ser utilizada, como *como Internet of things* (IoT), rede, nuvem, etc;
- Ferramentas (*Tools*): são usadas para implantação ou criação do cenário, como também do enredo;
- Enredo (*Storyline*): são descrições de um ou mais cenários de exercícios, contendo as informações mais relevantes relacionados aos seus objetivos.

2.2.6 Monitoramento (Monitoring)

Os usuários são monitorados durante a execução dos exercícios. Este componente está fortemente conectado ao projeto, o monitoramento se torna essencial para coletar e analisar os resultados. A seguir, exemplificaremos suas funcionalidades:

- Método (*Method*): realiza monitoramento das atividades e dos testes, através de ferramentas manuais ou automatizadas;
- Painel de controle (*Dashboard*): administra o ambiente de acordo com as informações coletadas pelos *logs*, já os demais *softwares* são usados para fornecer relatórios detalhados;
- Camada (*Layer*): categoriza os tipos de monitoramento, dependendo dos vários tipos de camadas;
- Ferramentas (*Tools*): são listadas e selecionadas para fornecer o monitoramento das práticas de segurança tanto no *software* quanto no *hardware*, além da possibilidade de gerenciar eventos de invasão e percepção;

2.3 Virtualização baseada em *containers*

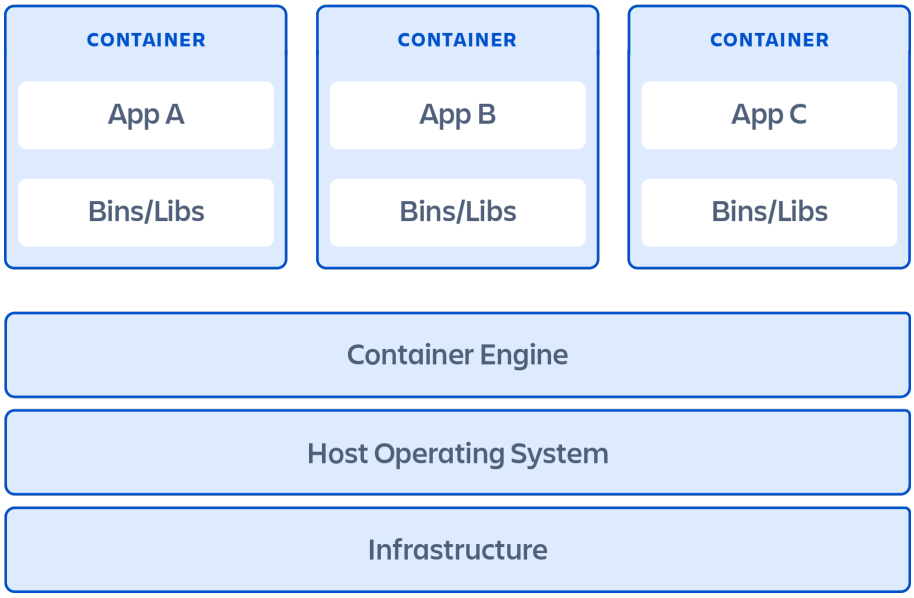
Levando em conta os métodos de virtualização presentes no mercado, *containers* e máquinas virtuais são dois tipos de tecnologias muito utilizadas para virtualização de recursos,

além de ambas serem muito semelhantes. Contudo, a virtualização é o processo no qual um recurso do sistema como memória RAM, CPU, disco rígido e até mesmo a rede, pode ser virtualizado e retratado em vários recursos. Um grande diferencial entre utilizar *containers* ao invés de máquinas virtuais é de que a VM virtualiza uma máquina por completo, já os *containers* conseguem virtualizar apenas o necessário das camadas de *software* que estão acima do nível do SO (ATLASSIAN, 2023).

Containers são pacotes de *software* leves que possuem em sua composição todas as dependências necessárias para executar sem problemas aplicativos de software contidos. As dependências podem ser desde bibliotecas do sistema, até pacotes de código externo e outras aplicações no nível do SO. Além disso, como os *containers* são leves e armazenam apenas o essencial para executar suas aplicações, é possível executar inúmeros *containers* em uma mesma máquina. Eles ainda são muito rápidos, o que facilita realizar modificações ou então destruir e recomeçar novos *containers* (ATLASSIAN, 2023).

Na Figura 2 é possível analisar a arquitetura de virtualização baseada em *containers*, começando por sua infraestrutura que pertence ao nível mais baixo. Em seguida, temos o *software* do sistema operacional *host* que é executado diretamente no *hardware* da máquina, para então chegarmos no *Container Engine* que ficará responsável por executar vários *containers* de maneira isolada no mesmo *kernel* do SO.

Figura 2 – Arquitetura da virtualização baseada em *containers*



Fonte: (ATLASSIAN, 2023)

2.3.1 Docker

Segundo informações da plataforma oficial, o *Docker* é uma ferramenta *open-source* lançado em 2013, com o objetivo de proporcionar o desenvolvimento, envio e execução de aplicações, através da virtualização do ambiente de desenvolvimento ou de produção em *containers*. Estes, por sua vez, poderão ser gerenciados em qualquer infraestrutura, já que o *Docker* possibilita a separação das aplicações de sua infraestrutura, melhorando assim o gerenciamento das aplicações. Além disso, possui funcionalidades importantes como isolamento, segurança e uma menor demanda de recursos de *hardware*, o que possibilita usufruir de várias tarefas em uma mesma máquina sem causar problemas (DOCKER, 2023).

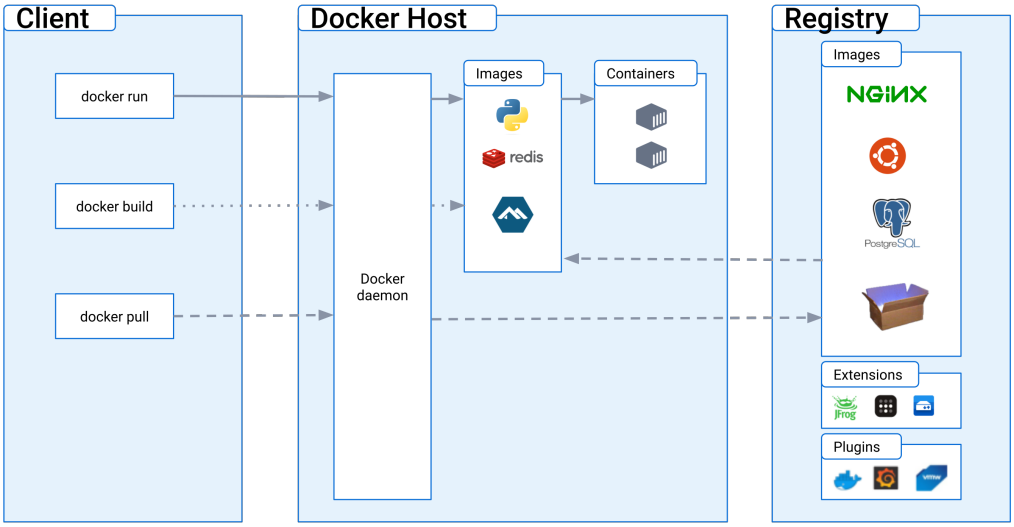
O *Docker* oferece funções como empacotar um software em uma chamada *image*, que poderá posteriormente ser executada e retornar esse mesmo *software* funcionando através de um *container*. Também é possível baixar e utilizar *images* já criadas e pré-compiladas pela comunidade através do repositório oficial e gratuito *Docker Hub*[1]. Com isso, já que os *containers Docker* são leves e carregam apenas o necessário para serem executados, podem ser facilmente compartilhados entre colegas ao mesmo tempo que fornecem garantia de que todos receberam a mesma versão do software, pois estão utilizando o mesmo *container*.

Além disso, como ilustrado na Figura 3, a arquitetura do *Docker* funciona de modo que o *Docker client* comunica-se com o *daemon* através de uma *API REST* em soquetes *UNIX* ou por meio de uma interface de rede. O *Docker daemon*, por sua vez, tem como papel construir, executar e por fim, distribuir os *containers Docker*. Dito isso, o *client* e o *daemon* podem tanto serem executados em um mesmo *host* ou separadamente, basta estabelecer a conexão entre o *Docker client* em *Docker daemon* remoto.

O *Docker Compose* é uma outra ferramenta bastante utilizada para executar e gerenciar *softwares* e/ou aplicações em vários *containers Docker*. Com o *Compose*, um único comando pode iniciar serviços, criar *images*, gerar *volumes*, *networks* e dentre outros. Tudo isso é possível através da configuração e execução de arquivos YAML, que por serem arquivos com uma linguagem facilmente legível pelo ser humano, auxilia bastante em todas as fases no ambiente de desenvolvimento e de produção.

[1] https://hub.docker.com

Figura 3 – Arquitetura *Docker*



Fonte: https://docs.docker.com/get-started/overview/

2.3.2 Emulação através de containers Docker

2.3.2.1 Mininet

O *Mininet* é um emulador de rede leve e de código aberto, usado para criar rede de *hosts* virtuais, *switches*, controladores e *links*, através da virtualização baseada em processos em um único *kernet* do SO. É implementado em *Python* e usa pequenos utilitários em C (MININET, 2023).

Os *hosts* emulados pelo *Mininet* executam o *software* de rede padrão do *Linux* para enviar e receber o tráfego. Os *switches* oferecem suporte ao *OpenFlow* para fazer a personalização do roteamento, o que é altamente flexível para determinar as rotas e seus respectivos destinos. Ele também suporta rede definida por *software*, já os controladores gerenciam os *switches* e os *hosts* para se comunicarem através da rede simulada. Além disso, o *Mininet* também oferece ambientes de testes de rede simples para o desenvolvimento de aplicações *OpenFlow*. E ainda, permite que vários desenvolvedores trabalhem de forma simultânea e independente na mesma topologia (MININET, 2023).

2.3.2.2 Containernet

O *Containernet* é um *fork* do emulador de rede *Mininet* e pode utilizar *containers Docker* como *hosts* nas topologias de rede emuladas. Com isso, é possível construir emuladores

e *testbeds* de rede/nuvem. Por conta disso, o *Containernet* é bastante usado em pesquisas pela comunidade, focados principalmente em experimentos de Computação em Nuvem, *Fog Computing, Network Function Virtualization (NFV)* e *Multi-access Edge Computing (MEC)* (CONTAINERNET, 2023).

Segundo Peuster *et al.* (2016), os autores deixam evidente algumas das vantagens de se utilizar esse poderoso *fork*, como a possibilidade de construir e destruir *containers* da rede emulada em tempo real, algo que não é possível no *Mininet*. Além disso, conta com a possibilidade de fazer alterações nas limitações de recursos, como processamento e memória RAM por *container*, em tempo real, sem a necessidade de realizar uma nova inicialização dos *containers* para efetivar essas mudanças.

Portanto, neste trabalho será utilizado o *Containernet* para a criação do laboratório de treinamento *Cyber Range*, através da emulação de rede em uma plataforma *Amazon Cloud* e com *containers Docker* para simular serviços, dispositivos, ataques e vulnerabilidades.

2.4 *React*

De acordo com a documentação oficial, o *React* é um *framework JavaScript* de código aberto, com foco na construção e renderização de interfaces de usuário (UI). A UI é formada a partir da combinação de unidades de texto, imagens e botões. O *React* possibilita a manipulação dessas unidades em componentes, que por sua vez, podem ser reutilizáveis e encaixáveis no desenvolvimento de sites ou aplicações móveis (REACT, 2023).

O *React* foi fundado em 2011 pelo engenheiro de *software* do *Facebook*[2] Jordan Walke e lançado posteriormente em Maio de 2013. Por conta disso, pode ser considerado uma tecnologia nova no mercado se for comparado com outras de mesma finalidade. Atualmente, essa poderosa biblioteca de *Front-end* é uma das mais usadas para desenvolvimento *web*, já que oferece rápida criação das aplicações por usar o *Virtual DOM*, que compara os estados passados dos componentes e atualiza apenas os itens que foram alterados. E ainda, por sua facilidade de usabilidade, já que requisita menos codificação e oferece mais funcionalidades (SIMPLILEARN, 2023).

[2] https://www.facebook.com

2.5 Go (GoLang)

Como detalhado na documentação oficial, Go é uma linguagem de programação de código aberto, sendo muito usado como *Back-end* nas aplicações e possui uma *sintaxe* semelhante à linguagem C. O Go começou a ser desenvolvido em 2007 e foi finalmente lançado em 2009. Originalmente, ele foi criado para atender internamente desenvolvedores do *Google*[3], que não estavam satisfeitos com a complexidade de outras linguagens utilizadas pela empresa (GO, 2020).

Os programas em Go são organizados em forma de pacotes. Cada pacote é composto por uma coleção de arquivos presentes no mesmo diretório, onde serão compilados de maneira conjunta. Um repositório Go pode conter um ou mais módulos, mas geralmente contém apenas um módulo, e este fica armazenado na raiz do repositório. Cada módulo pode possuir uma coleção de pacotes relacionados.

2.6 Infraestrutura como Código (IaC)

A Infraestrutura como código (IaC) pode ser definida como o gerenciamento e provisionamento da infraestrutura por meio da execução de códigos ao invés de processos manuais. Ao utilizar o IaC para automatizar o provisionamento da infraestrutura, os desenvolvedores não precisam ficar submissos em provisionar e gerenciar manualmente seus servidores, sistemas operacionais, armazenamentos e outros componentes de infraestrutura, sempre que seja necessário implementar ou criar uma nova aplicação (REDHAT, 2022a).

O controle de versão também é uma etapa importante na IaC. Afinal, os arquivos de configuração devem pertencer à fonte, assim como qualquer outro código-fonte de *software*. Além disso, caso não haja alterações no código-fonte, essas configurações também asseguram que o provisionamento sempre seja do mesmo ambiente todas as vezes, reduzindo possíveis erros indevidos e prejudiciais na infraestrutura. Na implantação da IaC, também é possível separá-la em módulos, que podem ser combinados de diferentes maneiras por meio da automação (REDHAT, 2022a).

[3] https://www.google.com

2.6.1 Ansible

O *Ansible* é uma ferramenta *open source* para automação de infraestruturas. Essa ferramenta é capaz de automatizar processos manuais de provisionamento, implantação de aplicações, gerenciamento de configurações, orquestração e dentre outros. Em comparação com outras ferramentas de gerenciamento mais simples do mercado, usuários do *Ansible* podem utilizar as regras de automação para instalar *softwares*, automatizar tarefas rotineiras, provisionar infraestrutura, melhorar a segurança, conformidade e aplicar *patches* em sistemas (REDHAT, 2021).

Para trabalhar com o *Ansible*, basta conecta-se ao que você quer automatizar, como por exemplo, plataformas de serviços em Nuvem, e implementar programas que executam instruções anteriormente inseridas manualmente. Esses programas usam módulos do *Ansible*, e cada módulo pode ser usado para atender expectativas específicas de conectividade, interface e comandos do *endpoint*. Com isso, o *Ansible* executa esses módulos através de conexão SSH com a infraestrutura. Além disso, não é necessário servidores, *daemons* ou bancos de dados adicionais. O usuário pode trabalhar através de seu terminal, um editor de texto e um sistema de controle de versões para então poder acompanhar as mudanças em seu conteúdo (REDHAT, 2021).

2.6.1.1 Ansible Playbook

Um *Ansible Playbook* é um *blueprint* para tarefas de automação, ou seja, cada módulo do *Ansible Playbook* contêm metadados que determinam quando e onde a tarefa é executada, além de qual usuário à realiza. E ainda, possibilita utilizar vários outros módulos que exercem ações de infraestrutura, que podem ser simples ou complexas. Essas ações são executadas com pouca ou nenhuma intervenção humana. Com isso, os *Ansible Playbooks* são gerados em um conjunto, grupo ou classificação de *hosts* que quando estão combinados, constituem um inventário do *Ansible* (REDHAT, 2022b).

Os *Playbooks* contribuem bastante para equipe de TI à desenvolver aplicações, serviços, nós de servidor ou até outros dispositivos, sem que ocorra uma desnecessária sobrecarga manual de criar tudo do zero. Os *Playbooks* podem ser facilmente salvos, compartilhados ou reutilizados, assim como também suas condições, tarefas e variáveis (REDHAT, 2022b).

3 TRABALHOS RELACIONADOS

Nesta seção, são descritos alguns trabalhos relacionados ao presente projeto, salientando que as buscas foram direcionadas em soluções que fizeram uso do *Cyber Range*.

3.1 CyRM: cyber range para auxiliar o ensino de defesa para alunos da disciplina de segurança da informação

No estudo realizado por Dantas (2022), foi analisado o crescente aumento no número de ataques cibernéticos e suas constantes evoluções em todas as áreas da computação, além da falta de profissionais especializados em segurança cibernética para atuar no mercado de trabalho. Ainda com base em seus estudos, foi proposto a criação do CyRM uma ferramenta gratuita implementada através do *Cyber Range*, emulação de rede com *Containernet* e a utilização de *containers Docker*. Essa ferramenta é capaz de treinar e capacitar estudantes para detectarem e corrigirem vulnerabilidades, guiados por um questionário.

Atribuindo uma extensão ao estudo de Dantas (2022), este trabalho tem como foco o desenvolvimento de uma ferramente de gerenciamento e administração que será disponibilizada na nuvem, visando maior usabilidade ao usuário, para que o mesmo possa usufruir de seus estudos usando o CyRM. A solução tem como objetivo oferecer uma maior facilidade no modo de criação e controle do ambiente de treinamento, e automatizar a execução dos *scripts* e criação da infraestrutura necessários para gerar os cenários de treinamento.

3.2 AWS EC2 Public Cloud Cyber Range Deployment

O trabalho de Beuran *et al.* (2022) analisa a necessidade de ambientes especializados para as atividades de treinamento de cibersegurança do CR, pois para Beuran *et al.* (2022) tanto o conhecimento quanto habilidades práticas em segurança cibernética devem ser adquiridos. Porém, a configuração desses ambientes podem ser uma tarefa cansativa, o que pode dificultar o uso do CR nos treinamentos de segurança. Por conta disso, é apresentado o uso da nuvem pública *Elastic Compute Cloud* (EC2) da *Amazon Web Services* (AWS) para a implantação do *Cyber Range CyRIS*, possibilitando assim a execução das atividades cibernéticas em escala e com um custo relativamente baixo.

Semelhante ao estudo de Beuran *et al.* (2022), o presente trabalho busca utilizar a nuvem pública da AWS para a criação dos ambientes de treinamento usando instâncias EC2.

No entando, será estendido o CyRM, incluindo o desenvolvimento de um aplicação para o gerenciamento e administração de cenários, bem como o *deployment* automatizado na nuvem para escalar a solução.

3.3 Network Web Traffic Generator for Cyber Range Exercises

Em Javali e Revadigar (2019), os autores revelam a importância de atividades com *Cyber Range* para validar ferramentas de segurança, além de ser um fator capaz de aprimorar as habilidades pessoais de seus usuários. Com isso, os autores analisaram o tráfego *web* executado em segundo plano nos exercícios de CR e perceberam a necessidade de emular uma rede ainda mais realista. As ferramentas existentes com esse objetivo estão limitadas apenas em gerar fluxos de pacotes simples ou reproduzir rastreamentos de rede que já foram capturados. Portanto, é proposto uma solução capaz de gerar tráfego *web* baseado no modelo de *Markov*, distribuição de *Dirichlet* e distribuição híbrida, além do desenvolvimento de uma GUI interativa em *Python*. Por fim, após avaliações de seus modelos, foi constatado que a solução pode ser usado em vários aplicativos de geração de tráfego *web*, incluindo o *Cyber Range*.

Assim como no trabalho de Javali e Revadigar (2019), nós propomos uma GUI interativa desenvolvida em *React*, combinada com *Go*, a fim de melhorar a usabilidade e gerenciamento dos cenários de treinamento para o usuário. Dessa forma, o usuário será capaz de construir seus laboratórios de estudo CR sem que haja a necessidade de consultar o terminal para isso.

3.4 Análise Comparativa

O Quadro 1 indica um resumo em forma de comparação dos aspectos mencionados nos trabalhos relacionados, em analogia ao trabalho proposto. De modo a facilitar o entendimento das principais semelhanças e diferenças do nosso trabalho e dos trabalhos relacionados.

Quadro 1 – Análise comparativa entre os trabalhos relacionados e a solução proposta.

Trabalho	Cyber Range	Desenvolvimento GUI (Frontend + Backend)	Plataforma Nuvem
(DANTAS, 2022)	CyRM	N/A	AWS
(BEURAN *et al.*, 2022)	CyRIS	N/A	AWS
(JAVALI; REVADIGAR, 2019)	N/A	Python	N/A
Este trabalho	CyRM	React + Go	AWS Academy

Fonte: Elaborado pelo autor.

Assim como no trabalho de Dantas (2022), este trabalho irá utilizar o CyRM para a criação dos cenários de treinamento em cibersegurança. Contudo, o trabalho citado não desenvolveu uma interface gráfica para gerenciar e agilizar a criação dos cenários. A nossa solução irá desenvolver uma interface visando melhorar a usabilidade e automação na criação da infraestrutura. Além disso, no trabalho de Dantas (2022), foi usando a nuvem da AWS para escalar os laboratórios, já o nosso fará uso do programa *AWS Academy*, por ser mais acessível e gratuito para os estudantes de ensino superior.

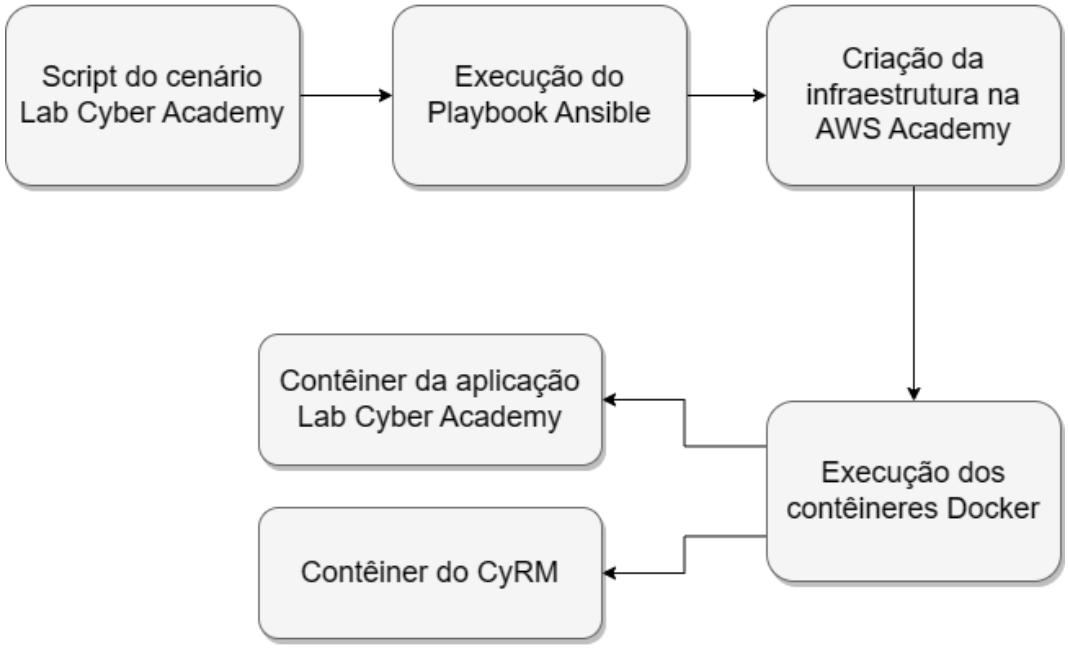
O trabalho de Beuran *et al.* (2022) faz uso da faixa cibernética CyRIS, contrariando nossa solução que utiliza o CyRM. Além disso, o trabalho de Beuran *et al.* (2022) não propôs o desenvolvimento de uma aplicação, mas faz uso da plataforma AWS para realizar análises e comparativos entre o uso do CyRIS com KVM e com AWS. Em contrapartida, iremos desenvolver uma aplicação com foco no gerenciamento e administração dos cenários, além de um implantação automatizada na *AWS Academy*.

Por fim, o trabalho de Javali e Revadigar (2019) não faz uso de um *Cyber Range* em si, mas sim da criação de uma ferramenta que gera tráfegos *web* úteis em exercícios de *Cyber Range*, já este irá usar o CyRM como faixa cibernética. Uma semelhança entre ambos os trabalhos é o desenvolvimento de uma GUI para facilitar a interação do usuário com a ferramenta por meio de uma interface gráfica. No estudo de Javali e Revadigar (2019), é usado a linguagem *Python* para o desenvolvimento da aplicação, enquanto neste é usado o *React* para o *Front-end* e *Go* para o *Back-end*. No entanto, não é designado uma plataforma de nuvem para armazenar a aplicação desenvolvida no trabalho de Javali e Revadigar (2019). Porém, é selecionado o programa *AWS Academy* como plataforma de implantação dos cenários de cibersegurança propostos pelo presente trabalho.

4 PROCEDIMENTOS METODOLÓGICOS

Nesta seção, serão apresentados os procedimentos metodológicos do trabalho proposto. Um conjunto de etapas serão adotadas, essas etapas são ilustradas no fluxograma da Figura 4 logo abaixo e descritas com mais detalhes nas próximas seções.

Figura 4 – Fluxograma do *Lab Cyber Academy*.



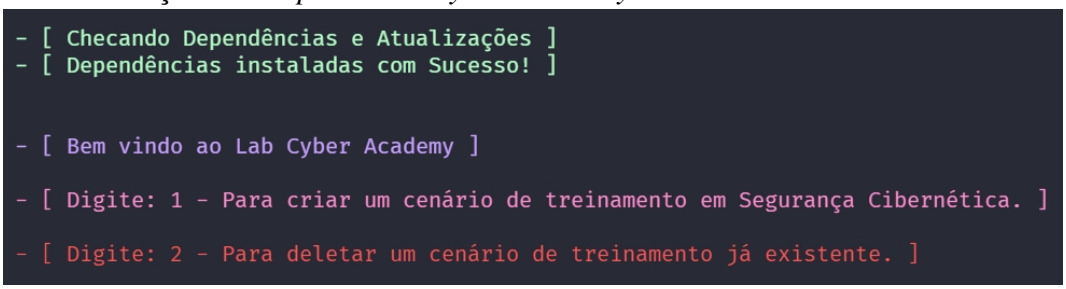
Fonte: Elaborado pelo autor.

4.1 Script do cenário Lab Cyber Academy

O uso de um *script* para automatizar a instalação das dependências necessárias, como o *Python*, *botocore* e *Ansible*, é de extrema importância, pois simplifica significativamente o processo de preparação do ambiente para a posterior execução do cenário de cibersegurança. Com esse *script*, o usuário não precisa se preocupar com a instalação manual de cada componente, economizando tempo e eliminando a possibilidade de erros decorrentes de configurações inadequadas. Além disso, esse *script* oferece a flexibilidade de criar um novo cenário de treinamento do zero ou deletar um cenário já existente, ambas de maneira automatizada. É necessário apenas confirmar o número com a opção desejada no momento da execução, como ilustrado na Figura 5. Dito isso, esse *script* para instalação das dependências está publicamente disponível em um repositório no *GitHub*[1].

[1] https://github.com/vitorreiel/lab-cyber-academy

Figura 5 – Execução do *script* do *Lab Cyber Academy*.

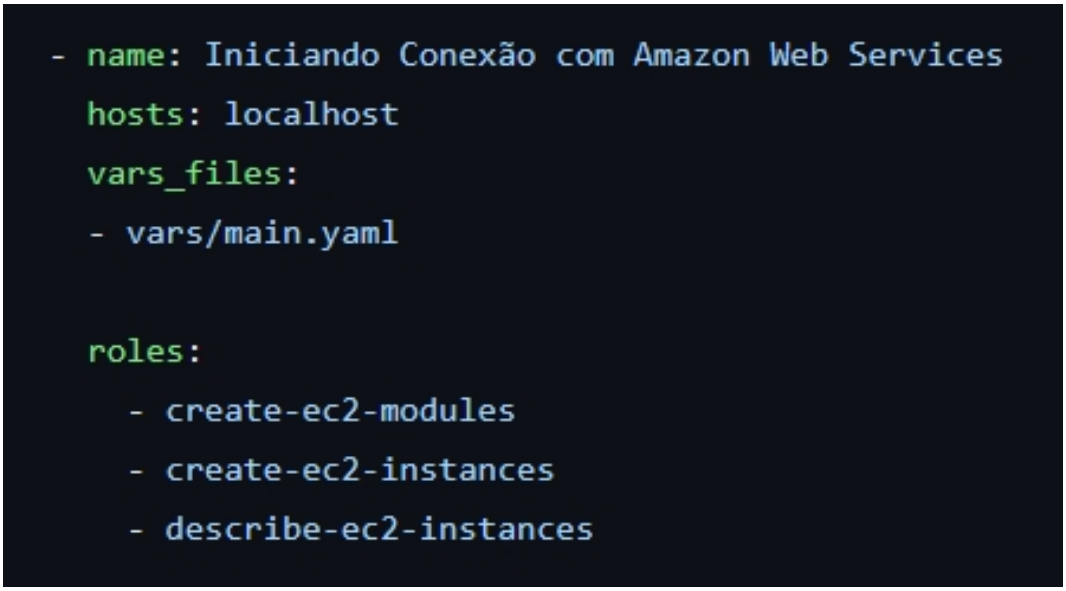


Fonte: Elaborado pelo autor.

4.2 Execução do Playbook Ansible

A execução do *Playbook Ansible* será realizada no final do *script* mencionado anteriormente. O *playbook*, como mostrado na Figura 6, desempenha um papel crucial na automação de tarefas complexas na *AWS Academy*. Ele é projetado para estabelecer uma conexão segura com a AWS Academy, utilizando as chaves de acesso da AWS CLI, e, de maneira automatizada através da execução de *roles*, criar uma *Virtual Private Cloud* (VPC) do zero, configurando todos os módulos necessários. Em seguida, ele provisiona uma instância EC2, configura a conectividade SSH para acesso remoto, e instala o *Docker* e as dependências específicas para o laboratório. Além disso, o *playbook* automatiza a criação de um novo usuário com permissões *Docker*, o que desempenha um papel fundamental no posterior treinamento de cibersegurança. Ao fazer isso, o *playbook* elimina a necessidade de intervenção manual em cada etapa, garantindo a consistência, a segurança e a eficiência no ambiente da *AWS Academy*, preparando-o para o aprendizado de forma confiável e sem contratempos.

Figura 6 – *Playbook* principal com execução das *roles* de provisionamento.



Fonte: Elaborado pelo autor.

4.3 Criação da infraestrutura na AWS Academy

A infraestrutura na *AWS Academy* é criada e configurada por *roles* do *Playbook Ansible* mencionado anteriormente. Esse processo inclui a criação de uma VPC com um bloco CIDR de 10.0.0.0/16 como mostrado da Figura 7, que serve como o espaço de endereço IP da rede. Em seguida, uma sub-rede é criada com o CIDR 10.0.0.0/20, permitindo uma segmentação eficaz da rede. Além disso, o *playbook* provisiona um *Gateway* que atua como um ponto de entrada e saída entre a VPC e a internet pública. Além disso, configura uma tabela de rotas para o roteamento adequado, estabelece um Grupo de Segurança para controlar o acesso à instância e cria um Par de Chaves SSH para garantir a segurança das conexões remotas. Posteriormente, uma instância EC2 do tipo *t2.large* com *Ubuntu Server* 22.04 LTS é implantada na VPC criada e com um armazenamento de 30GB.

Figura 7 – *Role* do *playbook ansible* com configuração da VPC.

```
- name: Criando VPC
  amazon.aws.ec2_vpc_net:
    name: Lab Cyber Academy VPC Module
    cidr_block: 10.0.0.0/16
    region: '{{ region }}'
    state: present
    aws_access_key: '{{ aws_access_key }}'
    aws_secret_key: '{{ aws_secret_key }}'
    aws_session_token: '{{ aws_session_token }}'
    tags:
      module: ec2_vpc_net
      this: works
  register: ec2_vpc_result
```

Fonte: Elaborado pelo autor.

4.4 Execução dos contêineres Docker

O uso do *Docker* desempenha um papel fundamental na otimização da infraestrutura da *AWS Academy*, ao reduzir o consumo de recursos computacionais e proporcionar flexibilidade na criação do cenário de aprendizado. Nesse contexto, o *Docker* será utilizado para criar imagens *Docker* do CyRM e para a aplicação do *Lab Cyber Academy*. Isso garante que os cenários possam ser replicados facilmente, economizando recursos, simplificando a gestão e oferecendo uma experiência de aprendizado mais eficaz e consistente para os alunos.

4.5 Contêiner da aplicação Lab Cyber Academy

O contêiner da aplicação do *Lab Cyber Academy* é uma ferramenta desenvolvida com um *front-end* em *React* e um *back-end* em *Go Lang*. Esta aplicação desempenha um papel essencial na experiência do aluno, pois oferece uma interface amigável e eficiente para interagir com os componentes do CyRM. Na Figura 8, é apresentado o arquivo ***docker-compose.yaml*** que será executado pelo *Playbook Ansible* utilizando a imagem construída através arquivo *Dockerfile*.

Figura 8 – *Docker Compose* da aplicação *Lab Cyber Academy*.

```
version: '3'
services:
  lab-cyber-academy:
    build: ./
    image: 'lab-cyber-academy/web-terminal:latest'
    container_name: lab-cyber-academy
    volumes:
      - ./config.toml:/root/config.toml
    ports:
      - '9000:9000'
```

Fonte: Elaborado pelo autor.

4.5.1 Imagem da aplicação do Lab Cyber Academy

O arquivo *Dockerfile* ilustrado na Figura 9 desempenha um papel crucial na construção de imagens *Docker* para a aplicação, pois ele é responsável por definir e automatizar a

Figura 9 – *Dockerfile* da aplicação *Lab Cyber Academy*.

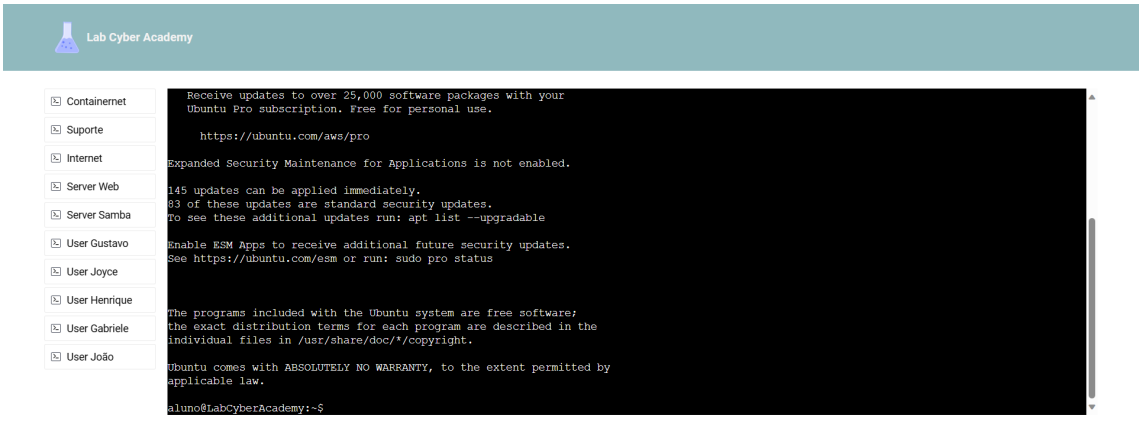
```
FROM golang:1.21-alpine AS go-env
WORKDIR /usr/src/app/go-ssh/
COPY . ./
RUN CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build -o app .

FROM node:16-alpine AS node-env
WORKDIR /usr/src/app/go-app/
COPY public ./public
RUN cd public && npm install && npm run build

FROM scratch
WORKDIR /root/
COPY --from=go-env /usr/src/app/go-ssh/app ./
COPY --from=node-env /usr/src/app/go-app/public/build ./public
EXPOSE 9000/tcp
ENTRYPOINT ["./app"]
```

Fonte: Elaborado pelo autor.

instalação de todas as dependências necessárias para o *front-end* e o *back-end*. Além disso, o *Dockerfile* configura a exposição da aplicação na porta 9000, permitindo que o contêiner baseado nessa imagem possa receber e responder as solicitações nessa porta específica.

4.5.2 Aplicação Lab Cyber Academy

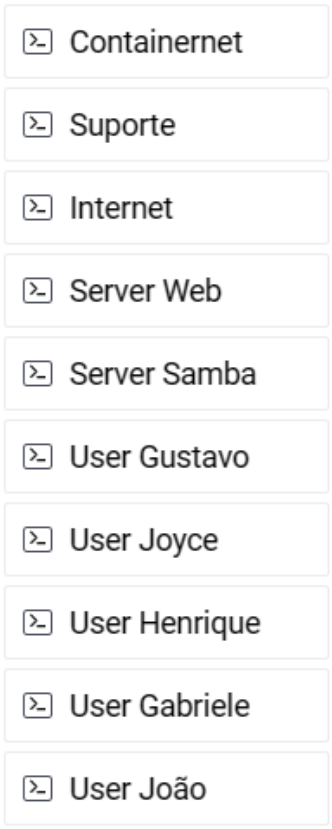
Figura 10 – Tela da aplicação *Lab Cyber Academy*.



Fonte: Elaborado pelo autor.

Ao final da execução do *Playbook*, um link ***http://ip-instancia:9000*** será disponibilizado para o aluno. Ao clicar nesse link, o aluno terá acesso à aplicação mostrada na Figura 10 em seu navegador, proporcionando uma experiência de fácil acesso e interação com o ambiente de treinamento de cibersegurança. Uma característica notável é a ausência de autenticação, simplificando o acesso do discente.

Figura 11 – Botões da interface *web* com *containers* referentes ao cenário CyRM.

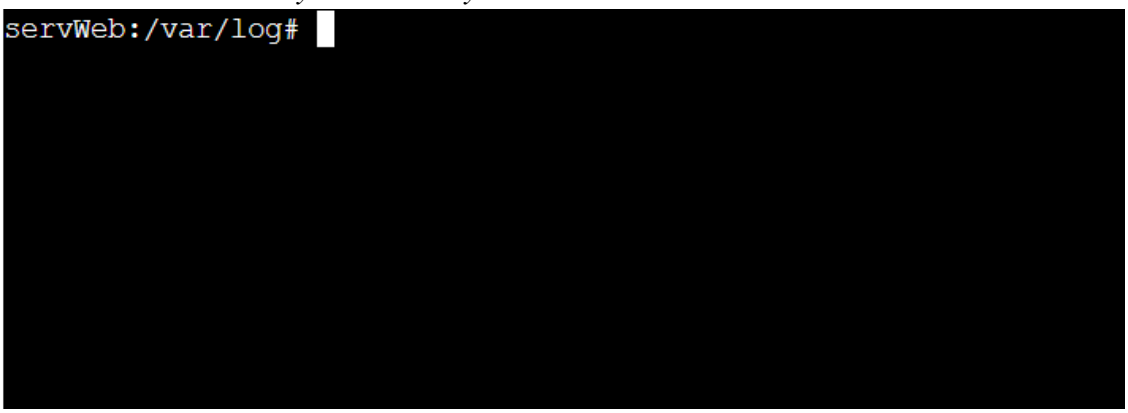


Fonte: Elaborado pelo autor.

Na Figura 11, apresentamos uma ilustração detalhada dos botões que compõem a interface *web* do *Lab Cyber Academy*. Esses botões são essenciais para a interação dos estudantes com os diversos *containers* que compõem o cenário do CyRM. Dito isso, para acessar um *container* específico necessário para a realização das atividades de aprendizado, os estudantes simplesmente precisam selecionar o botão correspondente na interface, que será indicado em cada questão no roteiro de estudo disponível no *Moodle*. Isso proporciona uma experiência de usuário simplificada, permitindo que os alunos foquem no conteúdo do roteiro sem a necessidade de procedimentos complicados de navegação.

Na Figura 12, é apresentado o terminal de linha de comando (CLI) da interface *web*

Figura 12 – CLI do *Lab Cyber Academy* referente ao *container Server Web*.



Fonte: Elaborado pelo autor.

do *Lab Cyber Academy*. Nesse exemplo específico, ilustramos como o terminal se configuraria para o usuário após a seleção do botão "*Server Web*". Com isso, os alunos podem executar os comandos necessários na CLI conforme especificado no roteiro de estudo, para a realização das atividades. Além disso, é importante ressaltar que a CLI se configurará de maneira distinta dependendo do botão selecionado pelo usuário.

4.6 Contêiner do CyRM

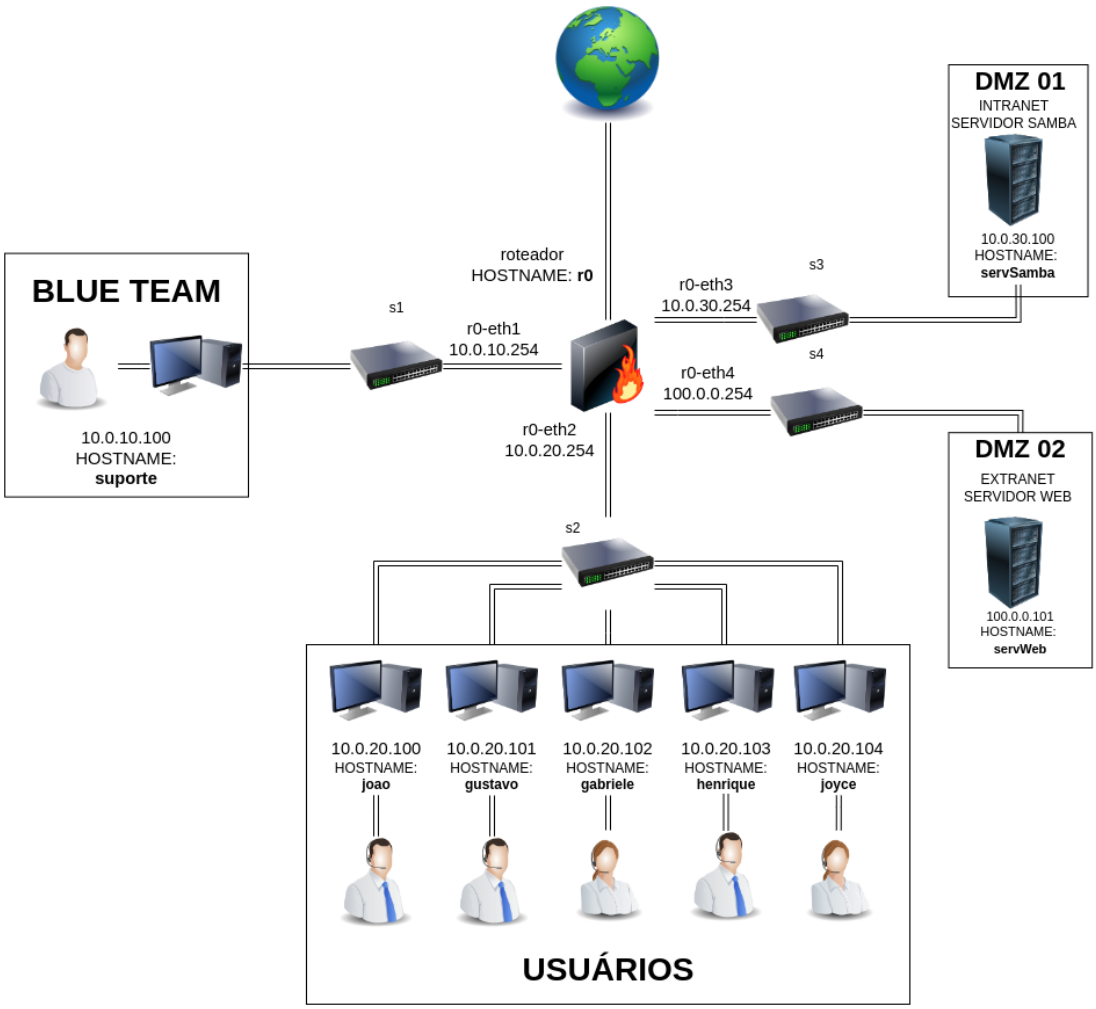
O CyRM permite a execução de cenários controlados e práticos voltados para o aprendizado de defesa. Esses cenários são construídos em torno de topologias emuladas, que empregam o *Containernet* como a ferramenta principal para a emulação de redes. Para operar o CyRM, são utilizados *scripts* desenvolvidos *Python*, fazendo uso das bibliotecas fornecidas pelo *Containernet*. A execução desses *scripts* é facilitada por um contêiner *Docker* baseado na imagem ***containernet_cyrm***[2], fornecendo um ambiente consistente e eficiente para criação e gerenciamento desses cenários complexos. Essa abordagem permite que os alunos vivenciem situações realistas de segurança cibernética de forma prática e segura, contribuindo significativamente para o seu aprendizado e preparação para desafios no campo da cibersegurança.

4.6.1 Definição do cenário do CyRM

A topologia ilustrada na Figura 13 representa uma rede corporativa com topologia estrela. Essa infraestrutura de rede é projetada para emular o acesso à Internet e é composta por quatro *Virtual Local Area Network* (VLAN): DMZ 01, DMZ 02, USUÁRIOS e *BLUE TEAM* (equipe de suporte).

[2] https://github.com/rafaelpinheiroo1/cyrm

Figura 13 – Topologia do CyRM.



Fonte: (DANTAS, 2022)

A DMZ 01 representa uma VLAN de *Intranet*, ou seja, uma rede acessível apenas internamente, com a sub-rede 10.0.30.0/24. Nesse ambiente, encontra-se exclusivamente um servidor de arquivos *Samba*, cuja função principal é facilitar o compartilhamento de arquivos e diretórios entre os funcionários da empresa. O servidor possui o endereço IP 10.0.30.100 (*servSamba*) e está conectado ao *switch* s3.

A DMZ 02 é configurada como uma VLAN de *Extranet*, com a sub-rede 100.0.0.0/24. Nesse ambiente, encontra-se um servidor Web dedicado que desempenha um papel essencial, oferecendo os serviços de armazenamento da empresa na internet, tornando-os acessíveis aos clientes em qualquer local. Além disso, o servidor *Web* disponibiliza um serviço SSH para permitir que a equipe de suporte se conecte e realize manutenções necessárias quando necessário. Adicionalmente, o servidor incorpora uma função de *syslog-ng*, que é responsável por armazenar e gerenciar todos os registros de *logs* relacionados às atividades do serviço. Isso permite que a

equipe de suporte identifique tentativas de acesso não autorizado. É importante destacar que a configuração do SSH apresenta uma abordagem simplificada, incluindo a utilização de portas padrão, nomes de usuário e senhas simples, o que potencialmente pode facilitar o acesso não autorizado por atacantes. O endereço do servidor é 100.0.0.101 (*servWeb*) e está conectado ao *switch* s4.

A VLAN USUÁRIOS é uma rede interna privada destinada aos funcionários encarregados de interagir com os clientes, prestando suporte e resolvendo questões. Neste cenário, ocorreu um incidente em que dois usuários deliberadamente atacaram o servidor *Samba*, enviando uma grande quantidade de pacotes simultaneamente. Esse comportamento resultou em uma sobrecarga do serviço, caracterizando um ataque de negação de serviço distribuído (*DDoS, Distributed Denial of Service*). Para realizar esse ataque, utilizamos a ferramenta *hping3*. As máquinas envolvidas no ataque foram alocadas na sub-rede 10.0.20.0/24 e conectadas ao *switch* s2, totalizando cinco PCs. Esse incidente destaca a importância de medidas de segurança e monitoramento para mitigar riscos e proteger os serviços críticos de uma organização.

A VLAN *BLUE TEAM* configura uma rede interna privada, compreendendo os funcionários do suporte técnico. Esses profissionais desempenham um papel crucial na administração e gestão da infraestrutura de rede, servidores e terminais. São responsáveis por garantir que os serviços das aplicações permaneçam disponíveis e solucionar quaisquer problemas que possam surgir na infraestrutura. As estações de trabalho do suporte estão equipadas com ferramentas essenciais, incluindo o *smbclient*, utilizado para conexões com o servidor *Samba*, visando manutenção e tarefas relacionadas. Além disso, o *Tshark* é empregado para a captura e análise de tráfego, oferecendo uma interface de linha de comando (CLI) eficiente para utilização em terminais *Linux* e minimizando o consumo de recursos. O SSH *Client* é utilizado para conexões com o servidor *Web*, permitindo a resolução de qualquer pendência. Adicionalmente, o *software geoiplookup* está instalado para identificar a origem geográfica de um atacante com base no endereço IP. A sub-rede 10.0.10.0/24 foi alocada para essa VLAN, e as máquinas estão conectadas ao *switch* s1. Vale ressaltar que nesta VLAN, encontra-se apenas uma estação de trabalho, com o endereço IP 10.0.10.100 (suporte).

O roteador r0 desempenha o papel de fornecer a interconectividade entre todas as VLANs e a Internet. O r0 é um roteador baseado em *Linux* com um *firewall* integrado (*netfilter/iptables*), projetado para filtrar o tráfego que passa por ele. Além disso, o *Tshark* está instalado para capturar e analisar o tráfego que percorre a rede.

A INTERNET representa a rede externa da corporação, na qual foi simulado o tráfego e a ocorrência de um ataque de força bruta ao servidor *web*. Esse ataque foi realizado com a ferramenta *THC Hydra*, com o propósito de tentar identificar combinações corretas de login e senha (DANTAS, 2022).

4.7 Roteiro de Estudo

Definimos o plano de treinamento por meio de um questionário disponibilizado no *Moodle* do Campus da UFC em Quixadá, conforme detalhado no Apêndice A. A Figura 14 exemplifica uma das perguntas incluídas. Este questionário abordou os aspectos relacionados à detecção, diagnóstico e mitigação de ataques no ambiente em questão, direcionando sua atenção para os servidores Samba e Web. A configuração do questionário seguiu as seguintes diretrizes:

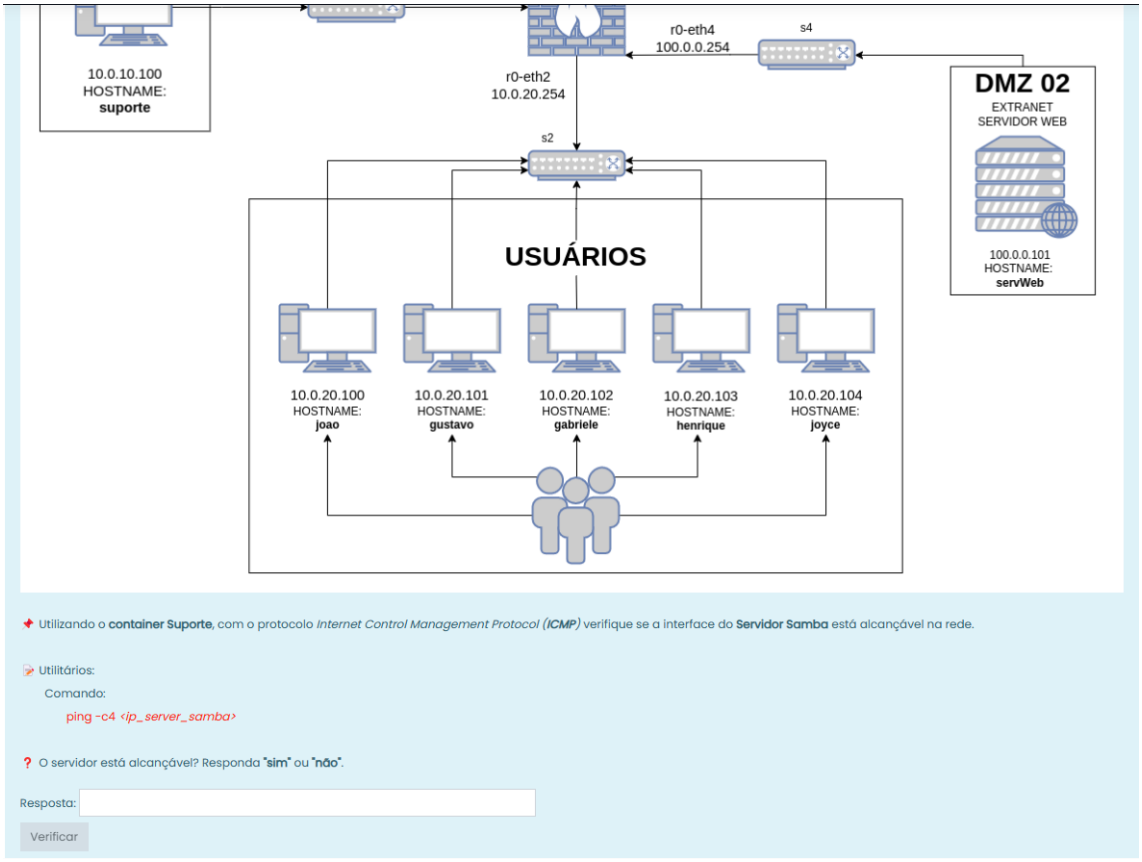
- Exibir uma questão por vez;
- Navegação sequencial, ou seja, uma vez avançada para a próxima pergunta, não há possibilidade de retorno;
- Cada pergunta apresenta a topologia do cenário, permitindo múltiplas tentativas de resposta. Cada tentativa é acompanhada de uma dica destinada a orientar o usuário rumo à resposta correta. O propósito é fornecer treinamento, não realizar intensivas avaliações.

A seguir, uma breve descrição sobre o relato do roteiro com adaptações para o presente projeto.

No servidor Samba, um ataque de *DDoS* ocorreu, deixando o servidor inacessível. O usuário João reportou dificuldades em acessar o servidor de compartilhamento e solicitou ao suporte para investigar a causa desse problema. O aluno, encarregado do suporte, foi incumbido de identificar o problema. Ao testar a conectividade com o comando *ping*, confirmou que o servidor estava acessível, mas o serviço Samba não podia ser alcançado. Ao acessar diretamente o servidor e usar a ferramenta *netstat* para analisar quais eram as conexões *TCP* que o servidor estava captando, identificou conexões com o PC do usuário Henrique.

Em vista disso, o aluno acessou o roteador central r0 e executou o *Tshark* para capturar o tráfego na interface conectada à VLAN USUARIOS. Ao analisar o tráfego, o estudante notou múltiplas requisições originadas do mesmo endereço identificado no *netstat*, com todas essas informações foi possível diagnosticar um ataque. Com isso, o aluno iniciou a mitigação bloqueando o tráfego de entrada no r0 com o *firewall iptables* e desconectando o *host* final do usuário para que não consiga mais enviar pacotes. Apesar da aparente conclusão da mitigação,

Figura 14 – Questão do Roteiro.



Fonte: Elaborado pelo autor.

o aluno percebeu que o serviço Samba ainda estava inacessível. Realizando novas análises, descobriu que o PC da usuária Joyce também estava realizando o mesmo ataque, efetuando assim uma nova mitigação. Após verificar novamente a disponibilidade do serviço, confirmou que o problema foi resolvido.

No Servidor Web, ocorreu um ataque de dicionário no serviço *SSH*. Diante disso, o suporte tomou providências para detectar, diagnosticar e mitigar o ataque. Utilizando novamente a ferramenta *netstat*, o aluno identificou um endereço IP estranho tentando se conectar ao servidor. Ao examinar os arquivos de *logs*, diagnosticou várias tentativas de acesso remoto do mesmo endereço IP utilizando diferentes usuários e senhas, caracterizando um ataque de dicionário ou força bruta. O estudante para mitigar esse ataque, alterou a porta padrão do *SSH* e a senha do usuário, além de bloquear o tráfego do endereço IP atacante no r0 com *iptables*. Após essas medidas, realizou novas verificações e constatou que os ataques finalmente cessaram.

5 EXPERIMENTOS

Nesta seção, descrevemos o processo de condução dos experimentos e analisaremos os resultados obtidos.

5.1 Descrição dos Experimentos

Nesta fase, conduzimos um teste de usabilidade da prova de conceito com os alunos da disciplina de Programação de Script no Campus da UFC de Quixadá, envolvendo a participação de 25 alunos. O experimento foi realizado de maneira presencial.

Para a execução deste teste, cada aluno criou uma instância temporária na *AWS Academy* com a seguinte configuração:

- Sistema Operacional: Ubuntu Server 22.04 LTS;
- Tipo de Instância: t2.micro (1 vCPUs e 1GB de RAM);
- Armazenamento: 30 GB.

Em seguida, os estudantes se conectaram a essa instância por meio do serviço SSH, clonaram o repositório *GitHub* da prova de conceito e executaram o *script* para automatizar a preparação da infraestrutura, criação do ambiente de treinamento e da aplicação, finalizando na exibição do *link* para acessar a aplicação no navegador. No início, fornecemos uma breve descrição da atividade e do funcionamento da ferramenta, como também, esclarecimento de dúvidas que surgiram durante a realização da atividade. Previamente, foi criada uma disciplina no *Moodle* do Campus para disponibilizar o questionário da ferramenta, os alunos foram matriculados e respondiam às questões enquanto interagiam com os comandos exigidos por elas no CLI da aplicação.

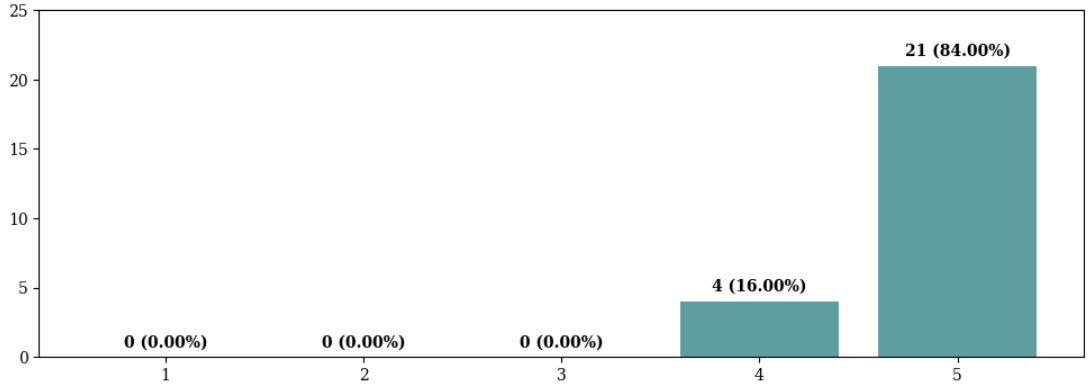
Ao término da atividade, os discentes responderam a um questionário de avaliação no *Google Forms*, proporcionando *feedback* sobre a experiência de uso, bem como possíveis *bugs* e cenários de lentidão encontrados. A Quadro 2 apresenta as perguntas do *Google Forms* às quais os estudantes responderam, totalizando 12 perguntas de avaliação.

5.2 Avaliação dos Resultados

Nesta seção, examinamos os resultados provenientes do preenchimento do questionário de avaliação. É importante destacar que todos os 25 alunos participantes responderam ao questionário.

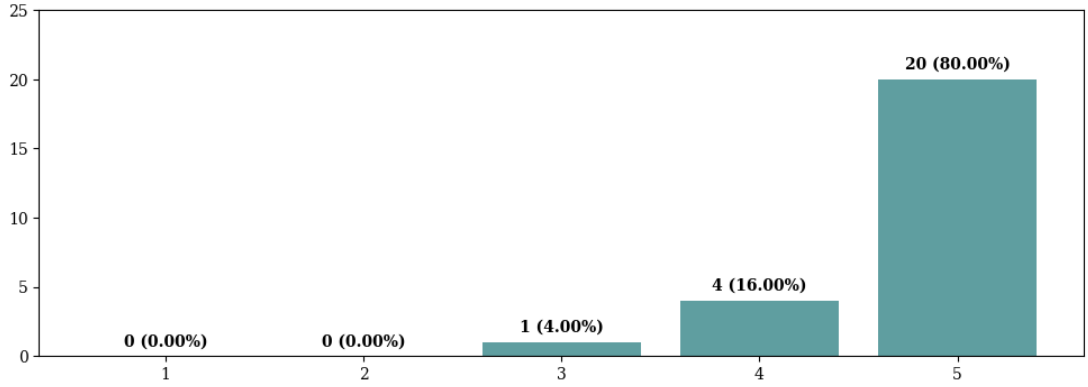
As **Questões 01 e 09** foram formuladas com o propósito de avaliar o nível de satisfação do usuário em relação às orientações para configurar o *script*, bem como para a execução da ferramenta em si. Os resultados dessas avaliações estão apresentados nas Figuras 15 e 16. O gráfico exibido na Figura 15 reflete, em uma escala de 1 (péssimo) a 5 (ótimo), que os estudantes não enfrentaram grandes dificuldades, ou nenhuma, durante a configuração inicial da ferramenta. Os dados representados na Figura 16 indicam que a maioria dos alunos (80%) expressou uma satisfação excelente em relação às instruções fornecidas para o uso da ferramenta.

Figura 15 – **Questão 01**: Qual o nível de satisfação na configuração inicial da ferramenta?



Fonte: Elaborado pelo autor.

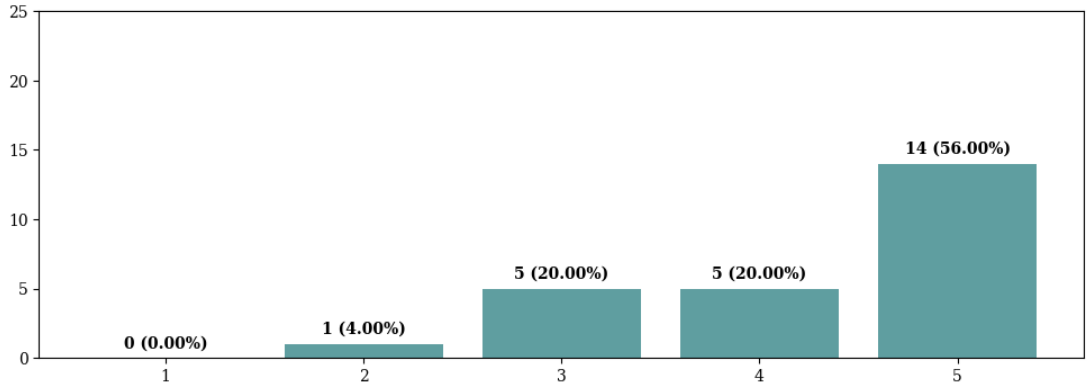
Figura 16 – **Questão 09**: Em que escala a documentação e as instruções fornecidas foram úteis para orientá-lo no uso da ferramenta?



Fonte: Elaborado pelo autor.

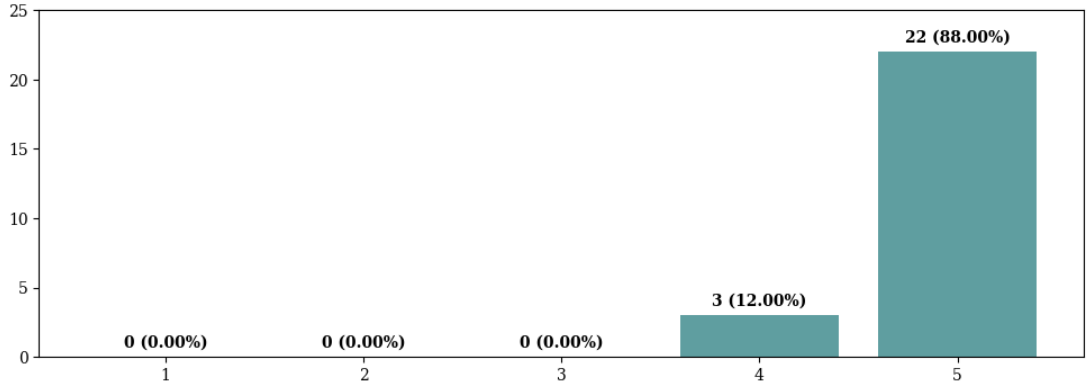
As **Questões 02, 03, 04** tinham o objetivo de avaliar as condições de execução da automatização da infraestrutura. Os resultados dessas avaliações podem ser observados nas Figuras 17, 18 e 19. No gráfico apresentado na Figura 17, em que a escala varia de 1 (péssimo) a 5 (ótimo), constata-se que a maioria dos estudantes (56%) raramente identificou *bugs* durante a execução. No gráfico da Figura 18, os alunos (88%) expressaram grande satisfação com a

Figura 17 – **Questão 02**: Como você avalia a ocorrência de bugs ou problemas durante a configuração da infraestrutura?



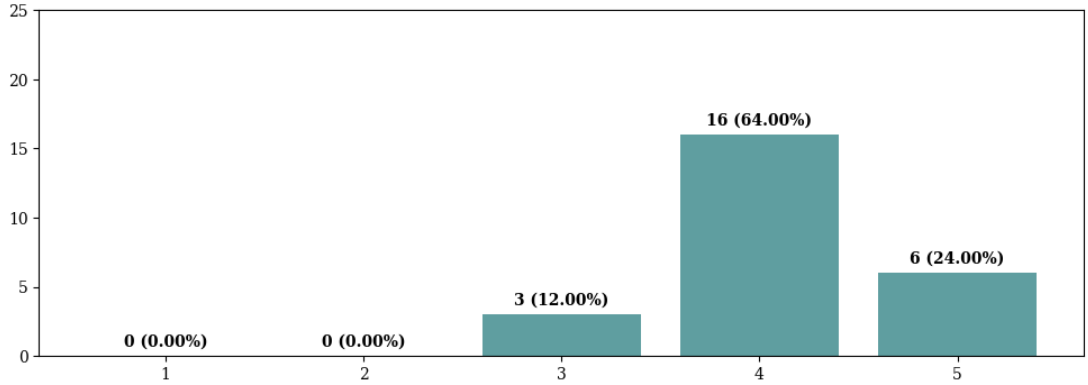
Fonte: Elaborado pelo autor.

Figura 18 – **Questão 03**: Como você avalia a eficácia da ferramenta na facilitação da criação dos cenários de forma automatizada?



Fonte: Elaborado pelo autor.

Figura 19 – **Questão 04**: Em que escala ocorre cenários de lentidão durante a criação da infraestrutura?



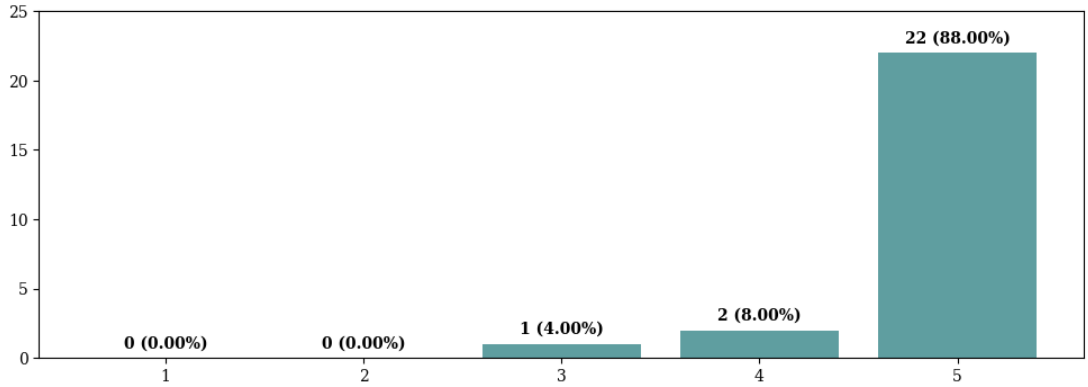
Fonte: Elaborado pelo autor.

facilidade proporcionada pela ferramenta na automatização da criação do ambiente. O gráfico da Figura 19 revela que a maior parte dos discentes (64%) classificou os cenários de lentidão na infraestrutura com nota 4, indicando um nível satisfatório.

A **Questão 05** foi formulada com o objetivo de avaliar o acesso ao link da aplicação

fornecido ao término do processo de criação do ambiente. O resultado dessa avaliação pode ser identificado na Figura 20. Diante disso, o gráfico exibido na Figura 20 reflete, em uma escala de 1 (péssimo) a 5 (ótimo), que a maioria dos alunos (88%) atribuiu nota máxima (5) à facilidade de acessar a aplicação por meio do link disponibilizado ao término da execução do *script*.

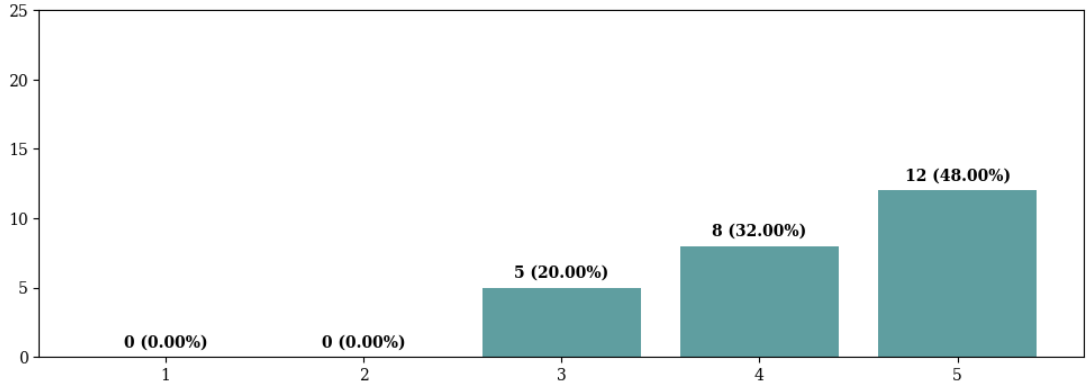
Figura 20 – **Questão 05**: Como você avalia a facilidade de acessar à aplicação por meio do link fornecido?



Fonte: Elaborado pelo autor.

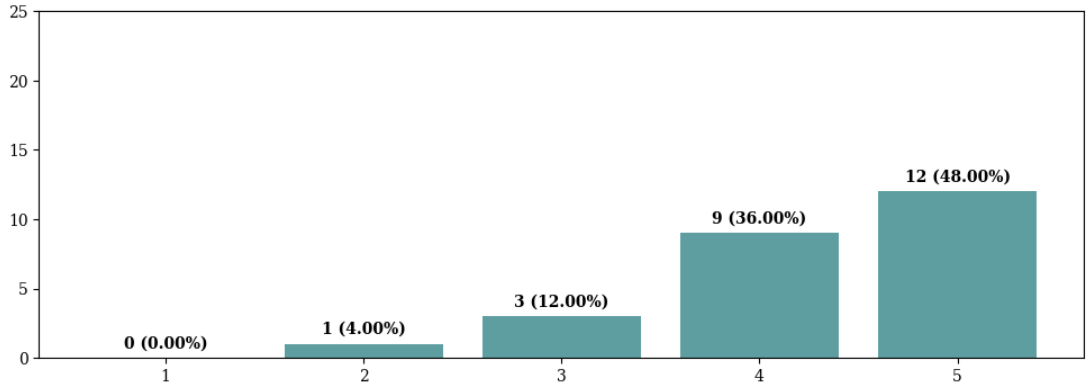
As **Questões 06, 07 e 08** foram elaboradas com o objetivo de avaliar o nível de satisfação dos usuários em relação às condições de execução da aplicação durante a atividade. Os resultados dessas avaliações podem ser conferidos nas Figuras 21, 22 e 23. O gráfico representado na Figura 21, em uma escala de 1 (péssimo) a 5 (ótimo), destaca que a maioria dos usuários (48%) encontrou poucos ou nenhum *bug* durante o uso da aplicação. No gráfico da Figura 22, a maioria dos estudantes (48%) identificou raramente cenários de lentidão durante a utilização da aplicação. Na Figura 23, a maioria dos alunos (84%) atribuiu nota máxima (5) em relação à responsividade e facilidade na interação entre os contêineres e no CLI da aplicação.

Figura 21 – **Questão 06**: Qual o nível de ocorrência de bugs ou problemas durante a execução da aplicação?



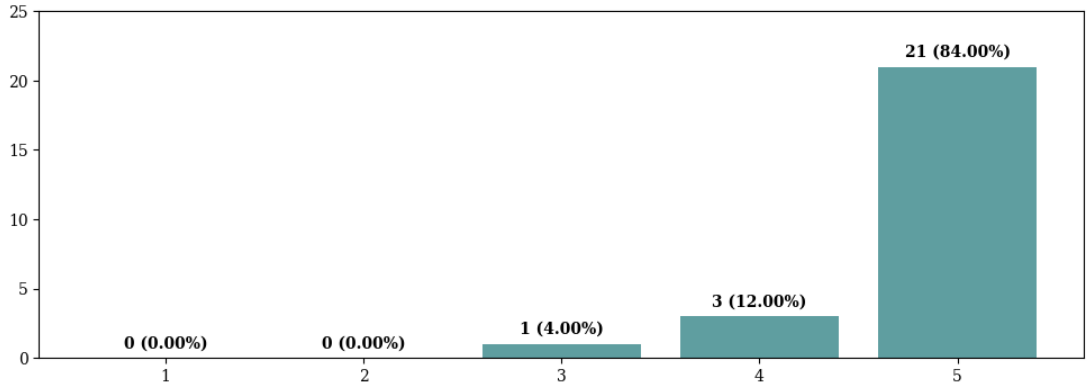
Fonte: Elaborado pelo autor.

Figura 22 – **Questão 07**: Como você avalia a ocorrência de cenários de lentidão durante o uso da aplicação?



Fonte: Elaborado pelo autor.

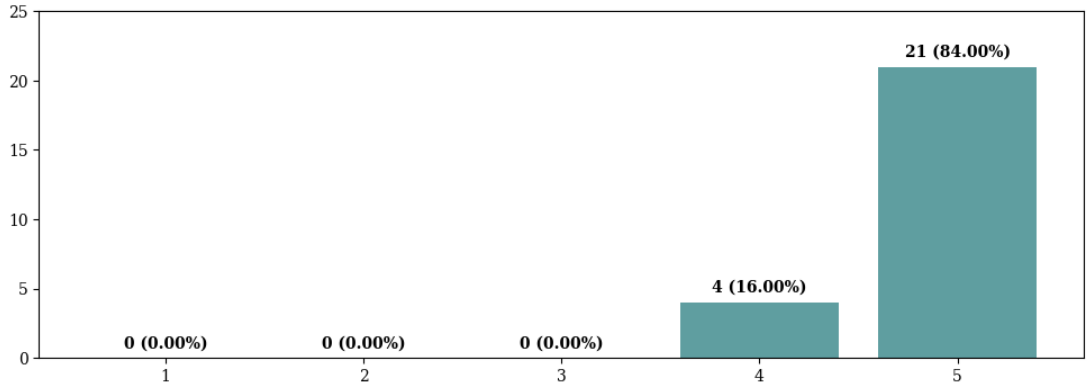
Figura 23 – **Questão 08**: Qual o nível da sua satisfação sobre a responsividade e facilidade de interação na execução do laboratório?



Fonte: Elaborado pelo autor.

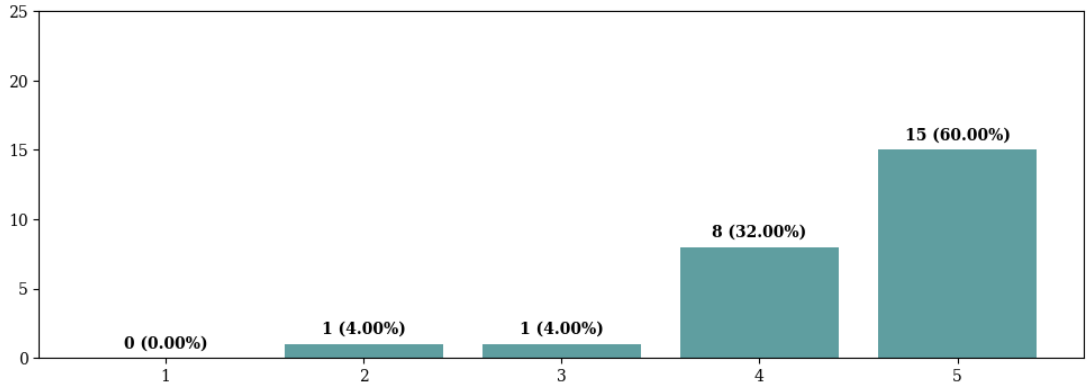
As **Questões 10, 11 e 12** foram formuladas com o propósito de avaliar a complexidade e os desafios das atividades, bem como determinar se estas são adequadas para usuários iniciantes e se são recomendáveis para alunos interessados na área de cibersegurança. Os resultados dessas avaliações estão representados nos gráficos das Figuras 24, 25 e 26. O gráfico da Figura 24, em uma escala de 1 (péssimo) a 5 (ótimo), revela que a atividade foi positivamente avaliada pela maioria dos discentes (84%) em termos de complexidade e desafios. No gráfico da Figura 25, observa-se que a maioria dos estudantes (60%) considera as atividades adequadas para iniciantes em cibersegurança. Por fim, no gráfico da Figura 26, a maioria dos alunos (88%) atribuiu nota máxima (5) ao recomendar o *Lab Cyber Academy* para colegas interessados em treinamentos em cibersegurança.

Figura 24 – **Questão 10**: Em que escala os cenários criados atenderam às suas expectativas em termos de complexidade e desafio?



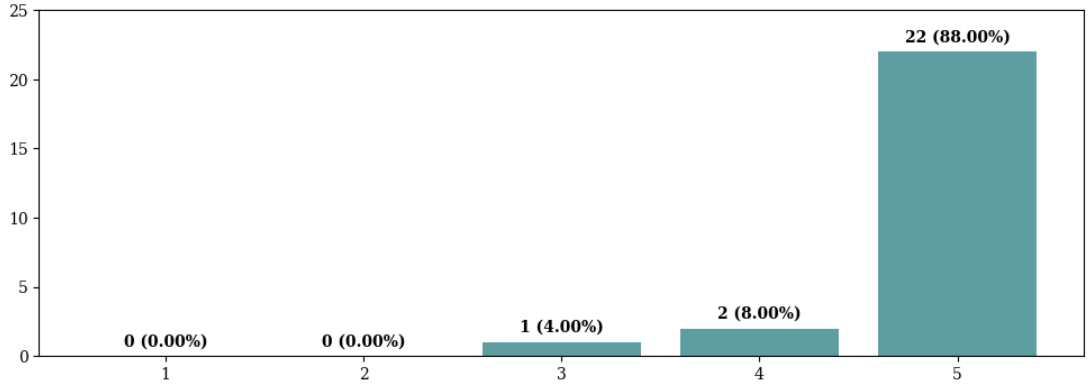
Fonte: Elaborado pelo autor.

Figura 25 – **Questão 11**: Em que escala o Lab Cyber Academy é adequado para usuários iniciantes em cibersegurança?



Fonte: Elaborado pelo autor.

Figura 26 – **Questão 12**: Em que escala você recomendaria o uso dessa ferramenta para outros alunos interessados em treinamento em cibersegurança?



Fonte: Elaborado pelo autor.

5.3 Discussão Final

Analisamos os resultados dos gráficos fornecidos pelos alunos e observamos que a configuração inicial da ferramenta não representou grandes desafios para eles. As instruções e

Quadro 2 – Perguntas do questionário de avaliação.

Número	Questões
1	Qual o nível de satisfação na configuração inicial da ferramenta?
2	Como você avalia a ocorrência de bugs ou problemas durante a configuração da infraestrutura?
3	Como você avalia a eficácia da ferramenta na facilitação da criação dos cenários de forma automatizada?
4	Em que escala ocorre cenários de lentidão durante a criação da infraestrutura?
5	Como você avalia a facilidade de acessar à aplicação por meio do link fornecido?
6	Qual o nível de ocorrência de bugs ou problemas durante a execução da aplicação?
7	Como você avalia a ocorrência de cenários de lentidão durante o uso da aplicação?
8	Qual o nível da sua satisfação sobre a responsividade e facilidade de interação na execução do laboratório?
9	Em que escala a documentação e as instruções fornecidas foram úteis para orientá-lo no uso da ferramenta?
10	Em que escala os cenários criados atenderam às suas expectativas em termos de complexidade e desafio?
11	Em que escala o Lab Cyber Academy é adequado para usuários iniciantes em cibersegurança?
12	Em que escala você recomendaria o uso dessa ferramenta para outros alunos interessados em treinamento em cibersegurança?

Fonte: Elaborado pelo autor.

documentações de apoio foram consideradas úteis e objetivas, proporcionando uma orientação eficaz durante o uso da ferramenta.

Os estudantes destacaram de maneira positiva a facilidade e eficácia da ferramenta na construção automatizada do ambiente na Nuvem, exigindo pouca interação do usuário. Além disso, expressaram satisfação ao acessar a aplicação em seus navegadores por meio do *link* disponibilizado no terminal após a execução do *script* responsável pela criação do ambiente de treinamento. Apesar da maioria dos alunos não ter enfrentado muitos *bugs*, lentidões ou problemas durante a automatização da infraestrutura, alguns estudantes ainda conseguiram identificar a presença desses problemas durante sua execução.

A avaliação dos usuários também foi positiva em relação à facilidade e qualidade da aplicação, especialmente na fácil interação entre os contêineres, permitindo alternar entre eles com simples cliques nos botões específicos de cada contêiner e também pela sua CLI para executar os comandos relacionados às questões da atividade. Embora a maioria dos estudantes não tenham encontrado muitos problemas ou *bugs* durante a execução da aplicação, alguns alunos ainda conseguiram identificar a presença desses problemas.

Os discentes avaliaram positivamente o laboratório, considerando que atende às suas necessidades em termos de complexidade e desafio. A ferramenta foi vista como recomendável para novos usuários interessados em treinamento em cibersegurança. Apesar da classificação

majoritária como adequada para iniciantes, houve uma pequena consideração de que as atividades do treinamento podem ser um pouco avançadas para esse público.

A análise dos dados coletados indica que o *Lab Cyber Academy* alcançou um índice de satisfação significativo. Este laboratório busca simplificar a criação de cenários de aprendizagem em *Cyber Range* de forma automatizada, implementada na nuvem, interativa por meio de uma aplicação e com questões guiadas em um ambiente emulado que simula serviços, componentes e cenários realistas de um ambiente corporativo.

6 CONCLUSÕES E TRABALHOS FUTUROS

Este projeto desenvolveu uma ferramenta de automação e interação simplificada por meio de uma aplicação implementada na nuvem com CyRM, destinada a auxiliar no ensino de defesa para alunos da disciplina de Segurança da Informação e áreas correlacionadas na UFC Campus Quixadá. A motivação para essa iniciativa derivou da identificação da necessidade de uma plataforma que se adequasse à realidade da instituição, oferecendo uma abordagem facilitada. Isso se tornou crucial devido à falta de treinamento especializado nessa área e à limitação de recursos de processamento e memória nos equipamentos da faculdade.

Ao conduzir pesquisas, utilizamos ferramentas fundamentais para o desenvolvimento deste trabalho, incluindo *Docker, Ansible* e o próprio CyRM. O *Docker*, por sua versatilidade, permitiu a criação de diversos cenários para executar a aplicação e simular ambientes de *Cyber Range*. O *Ansible* utilizado como infraestrutura como código (IaC), realizou o provisionamento completo do ambiente de treinamento e a execução dos cenários na nuvem.

Os objetivos específicos definidos para atingir o objetivo geral envolveram, primeiro, a escolha do CyRM como ferramenta de *Cyber Range*, considerando as soluções já existentes e a identificação de ferramentas aplicáveis, como *Docker* e *Ansible*. Em seguida, implementamos a solução de *Cyber Range* com base em um caso de uso, focando no treinamento de segurança e conduzindo um experimento bem-sucedido com alunos da disciplina de Segurança da Informação na UFC Campus Quixadá.

Na etapa final, avaliamos a solução proposta na turma de Programação de Scripts da UFC em Quixadá. Apesar de identificarmos a necessidade de algumas melhorias, a plataforma cumpriu seu propósito e foi considerada satisfatória pelos participantes do experimento.

Consequentemente, este trabalho permitiu o desenvolvimento de uma plataforma de automação para estudos de *Cyber Range*, utilizando contêineres *Docker*, métodos de IaC com *Ansible* e uma aplicação implementada na nuvem para facilitar a interação com os cenários. A plataforma é gratuita e está disponível no *GitHub* para uso por alunos e professores.

Em pesquisas futuras, sugerimos investir em melhorias para resolver os problemas de lentidões identificados durante o experimento. Além disso, também sugerimos a criação de novos *containers* por meio da interface *web* para o cenário do CyRM. E ainda, explorar e adaptar novos tipos de *Cyber Range*, além da implementação de uma opção que permita a seleção do *Cyber Range* que será utilizado no laboratório. Essa adaptação visa incorporar novos cenários de treinamento ao projeto.

REFERÊNCIAS

ATLASSIAN. **Containers vs. virtual machines**. 2023. Disponível em: https://www.atlassian.com/microservices/cloud-computing/containers-vs-vms. Acesso em: 24 maio 2023.

BEURAN, R.; ZHANG, Z.; TAN, Y. Aws ec2 public cloud cyber range deployment. **IEEE**, 2022 IEEE European Symposium on Security and Privacy Workshops (EuroSPW), p. 433–441, 2022. Disponível em: https://ieeexplore.ieee.org/document/9799381. Acesso em: 24 maio 2023.

CONTAINERNET. **Containernet | Use Docker containers as hosts in Mininet emulations.** 2023. Disponível em: https://containernet.github.io/. Acesso em: 29 maio 2023.

DANTAS, A. R. P. **CyRM**: cyber range para auxiliar o ensino de defesa para alunos da disciplina de segurança da informação. 2022. Disponível em: http://www.repositorio.ufc.br/handle/riufc/68364. Acesso em: 24 maio 2023.

DOCKER. **Docker Overview**. 2023. Disponível em: https://docs.docker.com/get-started/overview/. Acesso em: 24 maio 2023.

FORTINET. **Brasil é o segundo país que mais sofre ataques cibernéticos na América Latina**. 2022. Disponível em: https://www.fortinet.com/br/corporate/about-us/newsroom/press-releases/2022/brasil-e-o-segundo-pais-que-mais-sofre-ataques-ciberneticos-na-a. Acesso em: 26 maio 2023.

FORTINET. **Fortinet relata que a América Latina foi alvo de mais de 360 bilhões de tentativas de ataques cibernéticos em 2022**. 2023. Disponível em: https://www.fortinet.com/br/corporate/about-us/newsroom/press-releases/2023/fortiguard-labs-reports-destructive-wiper-malware-increases-over-50-percent. Acesso em: 26 maio 2023.

GO. **Using Go at Google**. 2020. Disponível em: https://go.dev/solutions/google/. Acesso em: 25 maio 2023.

GOOGLE. **Vantagens e desvantagens da computação em nuvem**. 2023. Disponível em: https://cloud.google.com/learn/advantages-of-cloud-computing?hl=pt-br. Acesso em: 27 jun. 2023.

JAVALI, C.; REVADIGAR, G. Network web traffic generator for cyber range exercises. **IEEE**, 2019 IEEE 44th Conference on Local Computer Networks (LCN), p. 308–315, 2019. Disponível em: https://ieeexplore.ieee.org/document/8990880. Acesso em: 24 maio 2023.

KASPERSKY. **O que é cibersegurança?** 2023. Disponível em: https://www.kaspersky.com.br/resource-center/definitions/what-is-cyber-security. Acesso em: 24 maio 2023.

MADDISON, J. **Relatório anual da Fortinet revela um aumento nas violações atribuídas à falta de habilidades em segurança cibernética**. 2023. Disponível em: https://www.fortinet.com/br/corporate/about-us/newsroom/press-releases/2023/fortinet-annual-skills-gap-report-uncovers-increase-breaches-attributed-to-lack-of-cybersecurity-skills. Acesso em: 26 maio 2023.

MININET. **Mininet Overview**. 2023. Disponível em: https://mininet.org/overview/. Acesso em: 29 maio 2023.

PEUSTER, M.; KARL, H.; ROSSEM, S. van. Medicine: Rapid prototyping of production-ready network services in multi-pop environments. **IEEE**, 2016 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), p. 148–153, 2016. Disponível em: https://ieeexplore.ieee.org/document/7919490. Acesso em: 29 maio 2023.

PHAM, C.; TANG, D.; CHINEN, K.-i.; BEURAN, R. Cyris: a cyber range instantiation system for facilitating security training. **ACM Digital Library**, Proceedings of the 7th Symposium on Information and Communication Technology, p. 251–258, 2016. Disponível em: https://dl.acm.org/doi/10.1145/3011077.3011087. Acesso em: 27 jun. 2023.

REACT. **Describing the UI**. 2023. Disponível em: https://react.dev/learn/describing-the-ui. Acesso em: 16 jun. 2023.

REDHAT. **O que é um Ansible?** 2021. Disponível em: https://www.redhat.com/pt-br/technologies/management/ansible/what-is-ansible. Acesso em: 24 maio 2023.

REDHAT. **O que é infraestrutura como código (IaC)?** 2022. Disponível em: https://www.redhat.com/pt-br/topics/automation/what-is-infrastructure-as-code-iac. Acesso em: 24 maio 2023.

REDHAT. **O que é um Ansible Playbook?** 2022. Disponível em: https://www.redhat.com/pt-br/topics/automation/what-is-an-ansible-playbook. Acesso em: 24 maio 2023.

SIMPLILEARN. **The Best Guide to Know What Is React**. 2023. Disponível em: https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs. Acesso em: 16 jun. 2023.

YAMIN, M. M.; KATT, B.; GKIOULOS, V. Cyber ranges and security testbeds: Scenarios, functions, tools and architecture. **ScienceDirect**, Computers Security, v. 88, p. 101636, 2020. Disponível em: https://www.sciencedirect.com/science/article/pii/S0167404819301804. Acesso em: 29 maio 2023.

APÊNDICE A – QUESTIONÁRIO APLICADO NO TREINAMENTO PARA OS ALUNOS

Questão 01. O usuário João reportou que não estava conseguindo acessar suas pastas compartilhadas no Servidor Samba. Você, como membro da equipe de suporte, ficou encarregado de identificar a causa e resolver o problema. Vamos começar! Utilitários: Container: Server Samba Comando: ifconfig. Qual o endereço IP do Servidor Samba?

Questão 02. Suponha que você tenha um usuário e um diretório remoto no servidor Samba. Com isso, podemos verificar se o serviço do Samba está realmente acessível remotamente através da execução de um comando smbclient. Para tanto, é necessário executar esse comando juntamente com algumas informações de usuário pelo Suporte. Utilitários: Container: Suporte Informações do usuário: Usuário: suporte Senha: badpass Diretório remoto: suporte Comando: smbclient //10.0.30.100/suporte -U "suporte". Aviso: Depois de executar o comando acima, responda o que se pede. Foi possível se conectar ao servidor? Responda sim ou não

Questão 03. Utilizando o container Suporte, com o protocolo Internet Control Management Protocol (ICMP) verifique se a interface do Servidor Samba está alcançável na rede. Utilitários: Comando: ping -c4 <ip_server_samba>. O servidor está alcançável? Responda "sim"ou "não".

Questão 04. Agora, a partir do Server Samba, liste os processos em execução no sistema. Utilitários: Comando: timeout 1s top | grep smbd. O serviço do Samba (smbd) está em execução? Responda "sim"ou "não".

Questão 05. Ainda no Server Samba, utilize o comando netstat para verificar se existem conexões TCP ativas com o serviço Samba. Utilitários: Comando: timeout 20s netstat. Existem conexões com o serviço Samba? Responda "sim"ou "não".

Questão 06. Ainda no Server Samba, através do comando ifconfig, verifique se está chegando muitos ou poucos dados na interface ethernet. Utilitários: Comando: ifconfig servSamba-eth0 Análise: - Número total de bytes transmitidos (TX bytes) - Número total de bytes recebidos (RX bytes). Como está o tráfego no servidor? Responda muito tráfego ou pouco tráfego.

Questão 07. De acordo com a questão anterior, o alto tráfego na interface ethernet do Server Samba está indisponibilizando o serviço Samba para novos acessos. Isso pode ou não ser classificado como um ataque. Vamos analisar isso! O tshark é uma ferramenta de linha de comando para análise de tráfego de rede, sendo uma alternativa ao Wireshark. Sabe-se que o

roteador r0 possui o tshark instalado. A partir do r0, é possível usar o tshark para capturar e salvar (.pcap) o tráfego na interface correspondente a VLAN DMZ 01. Mas antes, faça o que se pede abaixo: Utilitários Apenas observe esse comando: r0 tshark -i <interface_name> -w saida.pcap -a duration:5 -d tcp.port==445,nbss Aviso: Observe a conexão com o Switch (s3), que por sua vez, se conecta em uma outra interface com IP 10.0.30.254, referente ao roteador principal (r0). Feito a análise do cenário anteriormente. Qual o nome da interface do roteador principal (r0) onde o tráfego deverá ser capturado?

Questão 08. Agora precisamos analisar o tráfego para buscar possíveis anormalidades. Utilizando o Tshark execute a captura por 5 segundos e filtre na porta e serviço específico. Dito isso, esteja conectado ao Containernet e identifique quais endereços IP estão enviando tráfego para o Servidor Samba. Utilitários Comandos: r0 tshark -i r0-eth3 -w saida.pcap -a duration:5 -d tcp.port==445,nbss r0 tshark -r saida.pcap. Dentre os IP's capturados, existe algum endereço IP que esteja enviando muito tráfego? Se existir, responda qual o endereço IP. Se não, responda simplesmente "não".

Questão 09. Com a análise de tráfego da questão anterior, você conseguiu identificar um alto tráfego gerado pelo IP 10.0.20.103 do User Henrique, pertencente a VLAN USUÁRIOS. Além disso, você descobriu que ele não está acessando o serviço Samba neste momento. Ou seja, provavelmente é um ataque sendo disparado pelo PC de Henrique. Portanto, é necessário realizar a mitigação deste ataque. Inicialmente você poderá realizar o bloqueio deste tráfego no roteador r0. Para tanto, você pode usar o iptables que consiste em um firewall com a funcionalidade de bloquear e liberar o tráfego. Lembre-se de ainda estar conectado ao Containernet! Utilitários Comandos: r0 iptables -I FORWARD -s <ip_atacante> -j DROP r0 tshark -i r0-eth3 -w saida.pcap -a duration:5 -d tcp.port==445,nbss r0 tshark -r saida.pcap Análise: Verifique se o fluxo diminuiu na interface correspondente a VLAN DMZ 01! Em caso positivo, você teve sucesso no bloqueio! Foi possível bloquear o ataque? Responda sim ou não.

Questão 10. Você já bloqueou o tráfego com o iptables, impossibilitando o atacante de prosseguir. Contudo, as conexões TCP previamente estabelecidas ainda se encontram ativas. Nesse caso, é necessário que desconecte o User Henrique do switch que faz parte da VLAN USUÁRIOS. Lembre-se de ainda estar conectado ao Containernet! Para realizar esse procedimento puxe o cabo do host que está afetando o serviço. Utilitários Comando: r0 link henrique <switch_name> down Aviso: Em caso o comando acima retornar alguma mensagem de erro, não se preocupe. Após ter realizado o processo anterior, verifique se ainda está ocorrendo o ataque!

r0 tshark -i r0-eth3 -w saida.pcap -a duration:5 -d tcp.port==445,nbss r0 tshark -r saida.pcap. O servidor Samba ainda está sofrendo o ataque do User Henrique? e qual o nome do switch ao qual ele estava conectado? Responda no seguinte formato: sim,switch_name ou não,switch_name

Questão 11. Pronto, possivelmente mitigamos o ataque que estava sendo realizado pelo User Henrique. Nesse caso, vamos testar o acesso ao serviço Samba novamente. A partir da conexão com Suporte, verifique se o diretório remoto do usuário Suporte no servidor Samba está acessível via smbclient. Seguem os dados novamente: Utilitários: Container: Suporte Informações do usuário: Usuário: suporte Senha: badpass Diretório remoto: suporte Comando: smbclient //10.0.30.100/suporte -U "suporte". Consegue obter acesso? Responda com "sim"ou "não".

Questão 12. Na questão anterior ainda não foi possível o Suporte obter acesso ao serviço Samba. Nesse caso, provavelmente qualquer outro usuário também não está conseguindo! Como isso é possível? Você como suporte realizou várias atividades até encontrar o atacante e fez a mitigação do ataque. Que problemão! Uma das possibilidades seria que o Server Samba ainda está sendo atacado, mas agora por outro dispositivo. Para verificar, realize o mesmo procedimento que fez com o User Henrique. Capture e análise o tráfego novamente. Caso seja preciso, filtre para verificar se há um novo atacante. Em caso positivo, faço o bloqueio do tráfego de origem e desconecte o atacante da rede. Finalmente, tente conectar ao Server Samba novamente para verificar se possui acesso! Utilitários Passo a passo conectado ao Containernet: 1º) Use o seguinte comando para capturar o tráfego r0 tshark -i r0-eth3 -w saida.pcap -a duration:5 -d tcp.port==445,nbss 2º) Use o seguinte comando ler o tráfego capturado r0 tshark -r saida.pcap 3º) Filtre o ip do atacante r0 tshark -r saida.pcap | awk ´$4 ~ /04$/ {print $4; exit}´ 4º) Use o seguinte comando para bloquear o tráfego r0 iptables -I FORWARD -s <ip_atacante> -j DROP 5º) Use o seguinte comando para desconectar do switch: r0 link <hostname> <switch_name> down Agora estando conectado ao Suporte: 1º) Conectar ao servidor samba via smbclient com a máquina do suporte smbclient //10.0.30.100/suporte -U "suporte". Informações de Usuário: Usuário: suporte Senha: badpass Diretório remoto: suporte.

Conseguiu identificar algum novo atacante? Se sim, qual o nome (hostname) do atacante? Foi possível acessar com smbclient? Responda com o seguinte formato: Se tiver atacante: sim,hostname_atacante,sim ou sim,hostname_atacante,não Se não tive atacante: não

Questão 13. Após várias análises, você mitigou os ataques que estavam indisponibilizando o serviço Samba. Agora estes serviços estão normalizados! Quais eram os endereços dos

atacantes? Responda exatamente nesse formato: <IP do primeiro atacante> e <IP do segundo atacante> Dica: Verifique a ilustração do cenário logo acima e observe a VLAN USUÁRIOS.

Questão 14. Os ataques foram originados na rede externa ou na rede interna da empresa?

Questão 15. Quais eram os nomes dos hostnames dos atacantes? Responda nesse formato: <hostname do atacante 1> e <hostname do atacante 2> Dica: Verifique a ilustração do cenário logo acima e observe a VLAN USUÁRIOS.

Questão 16. No mundo da tecnologia há vários tipos de ataques, cada um com seu devido propósito. No ataque mitigado, observamos que a indisponibilidade do serviço Samba foi provocada por uma inundação de pacotes disparado por dois atacantes. Como é o nome desse ataque? Responda em letra minúscula!

Questão 17. De modo que seja possível realizar a proteção dos dados contra ameaças internas e externas, é importante que haja a garantia de alguns princípios básicos da segurança da informação. Confidencialidade que garante que os dados sejam acessíveis e somente pessoas autorizadas possam usufruir desses dados. Integridade é o pilar que garante que a informação trafegada não foi deletada ou corrompida. Disponibilidade garante que a informação esteja sempre disponível e acessível para os usuários. Há outros que também são muito importantes, como a Autenticidade e a Irretratabilidade (Não-Repúdio). Qual princípio da segurança da informação que o ataque DDoS violou? Digite a resposta em minúsculo.

Questão 18. Você, como suporte, analisou o servidor Samba, pois o mesmo estava inacessível, e com essas análises chegou a conclusão que o serviço estava sofrendo um ataque DDoS (Distributed Denial of Service). Em seguida, você realizou a mitigação, bloqueando e desconectando o acesso dos atacantes, solucionando assim o problema! O CEO da corporação está preocupado com a segurança dos dados sigilosos da empresa após um incidente e pediu que você avaliasse o Servidor Web. O objetivo é garantir que o serviço funcione sem interrupções e que seja protegido contra cibercriminosos. Além disso, o servidor permite o acesso remoto via SSH, mas somente a equipe de suporte tem acesso às credenciais. Utilitários Nota: Utilizando a máquina Suporte, com o protocolo Internet Control Management Protocol (ICMP) verifique se a interface do Server Web está alcançável na rede. Comando: ping -c4 <ip_server_web>. O servidor está alcançável? Responda "sim"ou "não".

Questão 19. Agora, vamos testar se a partir do Suporte é possível acessar remotamente o Server Web via SSH na porta 22. Utilitários Informações de acesso: Usuário: root

Senha: 123456 Comando: ssh <usuario>@<ip_server_web> Nota: Salve o comando usado para efetuar o SSH no Server Web, ele será necessário para responder à questão em caso de sucesso. Conseguiu acessar? Responda da seguinte forma: sim,comando ou não

Questão 20. Agora que você verificou que o suporte consegue acessar remotamente o Server Web, vamos fazer algumas análises para ter certeza de que todos os serviços do servidor web estão funcionando corretamente. Utilitários Mantendo a conexão SSH a partir do Suporte para o Server Web, utilize o comando netstat para verificar as conexões TCP ativas. Comando: timeout 20s netstat. Existe alguma anormalidade? sim ou não. Em qual protocolo (não é tcp)? Qual o state das conexões com anormalidade? Qual o IP Foreign Address? É um IP da rede interna ou externa? OBSERVAÇÃO: Responda somente com letras minúsculas. Separe as respostas por vírgula e sem espaço.

Questão 21. De acordo com a questão anterior, descobrimos que o IP externo 1.178.218.56 está tentando acessar o serviço SSH no Server Web. Contudo, somente o Suporte tem autorização para isso. Portanto, vamos investigar mais a fundo. Vale ressaltar que o serviço SSH está acessível, logo pode não ser um ataque DDoS. Contudo, ainda existem chances de ser um ataque. Nos sistemas operacionais Linux existem vários arquivos de logs para determinada funcionalidade, todos armazenados na pasta /var/log. Portanto, o próximo passo seria verificar o arquivo de log do serviço SSH que trata das tentativas de login, pois precisamos verificar se o suposto atacante está testando pares de login/senha aleatórios para acertar as credenciais verdadeiras e usufruir do Servidor Web. Utilitários Nota: Saia da sessão SSH com o Server Web usando o comando: exit. Feito isso, se conecte ao Server Web. Agora, leia o arquivo log de autenticação que possua as tentativas de conexão no servidor. Comando: cd ../var/log ls tail -n 20 <nome_arquivo>.log Análise: Verifique a quantidade de tentativas de conexão, senhas inválidas e o tempo entre cada tentativa de conexão. Qual o nome do arquivo que armazena os logs de tentativas de login? Realmente possui múltiplas tentativas de conexão partindo de um IP suspeito? Responda neste formato: <nome do arquivo>,sim ou <nome do arquivo>,não.

Questão 22. Com a checagem do arquivo de auth.log, você conseguiu diagnosticar o problema: o endereço IP 1.178.218.56 está tentando um acesso não autorizado ao Server Web, através de múltiplas tentativas de conexão. Isso é um ataque! Portanto, é necessário realizar a mitigação deste ataque. Inicialmente, você poderá realizar o bloqueio deste tráfego no roteador r0. Para tanto, você pode usar o iptables, que consiste em um firewall com a funcionalidade de bloquear e liberar o tráfego. Vamos então fazer o bloqueio do tráfego! Utilitários Nota: Esteja

conectado no Containernet Comando: r0 iptables -I FORWARD -s 1.178.218.56 -j DROP r0 iptables -L Análise: Agora conecte-se ao Server Web e verifique o arquivo auth.log novamente, observe os horários dos ataques e se estão surgindo novas tentativas de conexão. tail -n 20 /var/log/auth.log. As tentativas de conexão cessaram? Responda sim ou não.

Questão 23. Com o passo realizado na questão anterior, foi possível bloquear o tráfego originado pelo IP do atacante. Contudo, outros ataques podem surgir provenientes de outros IP's. Nesse caso, a nossa atividade de mitigação ainda está incompleta. Um ponto que facilita a execução de ataques em serviços na Internet é o uso da porta padrão. No caso do nosso serviço SSH, estamos usando a porta padrão 22. Portanto, vamos dificultar a vida de futuros atacantes alterando a porta padrão para outra completamente diferente. Seguem os passos necessários que dever ser executados diretamente no Server Web: Utilitários Passo a Passo: 1º) Liste os processos no Server Web e identifique o PID do serviço SSH ps aux Exemplo do pid que deverá ser encerrado: 19 root 0:00 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups 2º) Matar o processo SSH no Server Web kill pid <id_do_processo> 3º) Verifique novamente se o serviço SSH realmente foi encerrado ps aux 4º) Inicialize o serviço SSH na porta 40157 /usr/sbin/sshd -D -p 40157 2>&1 & 5º) Verifique se o serviço SSH realmente está em execução e na porta 40157 ps aux Nota: Para testar se realmente trocou a porta corretamente, entre em Suporte e acesse remotamente o Server Web, utilizando a nova porta. Conseguiu acessar? sim ou não. Qual comando você usou para acessar? OBSERVAÇÃO: respostas separadas por vírgulas e sem espaço.

Questão 24. Estamos melhorando a segurança do servidor, mas podemos fazer ainda mais. Percebeu que a senha de acesso ao SSH é 123456? Não acha muito fácil? Utilizar senhas mais complexas irá dificultar a atividade de novos atacantes. A Cartilha de Segurança da Internet, no Fascículo Senhas, possui boas práticas para a escolha, uso e armazenamento de senhas de forma segura. Portanto, você deve fazer a troca da senha de acesso. Para tanto, atualize a senha atual para outra encontrada no arquivo /root/pass.txt do Server Web. Para a troca de senha do usuário root utilize os seguintes comandos: Utilitários Comando: cat /root/pass.txt Aviso: Salve a senha acima em alguma local, você irá precisar dela para o próximo passo! passwd root Nota: Vai pedir a nova senha (introduza a senha que estava guardada no arquivo /root/pass.txt) Vai pedir para repetir a senha digitada (introduza a senha novamente) Agora para verificar se o procedimento teve sucesso, se conecte em Suporte e faça um acesso SSH ao Server Web usando a nova senha. Foi possível trocar a senha do usuário root e fazer o SSH utilizando a nova senha?

sim ou não.

Questão 25. Com a mitigação completa e as configurações do servidor mais seguras, o CEO da empresa pode ficar mais tranquilo! Agora, sabendo qual é o IP do atacante, vamos verificar de qual país que o ataque se originou. Para tanto, a ferramenta geoiplookup possui a finalidade de identificar qual o país de origem. Utilitários Nota: Tenha certeza de estar conectado em Suporte Comando: geoiplookup 1.178.218.56. Qual o nome de país? Responda a primeira letra em caixa alta. Como neste exemplo: China

Questão 26. Na atualidade da tecnologia que vivemos, há vários tipos de ataques, cada um com seu devido propósito. Podendo observar que o atacante utilizou dicionários contendo logins e senhas com o propósito de acertar as credenciais verdadeiras, a fim de obter acesso ao serviço para possivelmente roubar e alterar informações. Qual nome do ataque descrito anteriormente? Responda em letra minúscula!

Questão 27. De modo que seja possível realizar a proteção dos dados contra ameaças internas e externas, é importante que haja a garantia de alguns princípios básicos da segurança da informação. Confidencialidade que garante que os dados sejam acessíveis e somente pessoas autorizadas possam usufruir desses dados. Integridade é o pilar que garante que a informação trafegada não foi deletada ou corrompida. Disponibilidade garante que a informação esteja sempre disponível e acessível para os usuários. Há outros que também são muito importantes, como a Autenticidade e a Irretratabilidade (Não-Repúdio). Qual princípio da segurança da informação o ataque de força bruta poderia ter violado, caso tivesse sucesso? Digite a resposta em minúsculo.