Uma imagem contendo Logotipo

Descrição gerada automaticamente

Relatório de CTF

Pickle Rick – TryHackMe

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | Pickle Rick – Alexandre Gualiume Coruquieri |
| **N° Revisão** | 1 |
| **Data de publicação** | 04/09/2024 |
| **Link** | https://tryhackme.com/r/room/picklerick |

|  |  |  |
| --- | --- | --- |
| **Redação** | Alexandre Gualiume Coruquieri | Estudante |
| **Revisão** | Nome do revisor | Orientador |
| **Aprovação** | Nome do aprovador | Diretor |

|  |  |  |
| --- | --- | --- |
| **Histórico de revisões** | | |
| **N°** | **Entregas** | **Descrição** |
| **0** | 04/09/2024 | Produção |
| **1** | DD/MM/AAAA | Revisão |
| **2** | DD/MM/AAAA | Aprovação |

**Sumário**

[Contextualização 3](#_Toc176370905)

[Desenvolvimento 3](#_Toc176370906)

[What is the first ingredient that Rick needs? 3](#_Toc176370907)

[What is the second ingredient in Rick’s potion? 7](#_Toc176370908)

[What is the last and final ingredient? 8](#_Toc176370909)

[Conclusão 9](#_Toc176370910)

[Referências 9](#_Toc176370911)

Contextualização

O CTF Pickle Rick é voltado principalmente para exploração de vulnerabilidades web, onde desafia o participante a vasculhar os mínimos detalhes de uma página de internet para encontrar pistas para a solução do desafio.

Desenvolvimento

What is the first ingredient that Rick needs?

Usando o **Nmap** para enumerar a máquina, vemos que há um serviço HTTP rodando na porta 80, por tanto se trata de uma aplicação Web.

(nmap -A: Modo Agressivo, scan do sistema operacional e versão dos serviços)

Texto

Descrição gerada automaticamente

Ao abrir a aplicação web, vemos a seguinte página:

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

Com o próprio navegador, podemos inspecionar o código HTML da página para encontrar o username:

Texto

Descrição gerada automaticamente

Em seguida, usaremos o **Gobuster** para fazer a enumeração dos diretórios presentes na página:

Texto

Descrição gerada automaticamente

Encontramos um diretório chamado “robots.txt”. Ele é responsável por orientar as ferramentas de busca, como o Google, para que não façam a indexação das palavras contidas nesse arquivo de texto ou diretórios que não devem ser exibidos.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Vamos armazenar esse texto para usar caso precise.

Agora, vamos usar a ferramenta **Wfuzz** com uma *wordlist* específica para encontrar alguma página de login no site, tendo em vista que temos um username e uma possível senha:

Texto

Descrição gerada automaticamente

Com o Wfuzz, encontramos um diretório chamado “/login.php”. Entrando nesse diretório web, nos deparamos com uma página de autenticação:

Interface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamente

Como já possuímos o username, podemos tentar usá-lo com a palavra que achamos no “robots.txt”. Dessa forma, garantimos acesso ao sistema:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Na página, podemos ver que há um “Command Panel” que se trata de um *Remote Code Execution (RCE).* Na qual, podemos digitar comandos bash Linux e ele irá rodar diretamente na máquina do servidor remotamente.

Usando o comando “ls” para listar os arquivos do diretório, encontramos um arquivo de texto chamado “Sup3rS3cretPickl3Ingred.txt”. Digitando esse arquivo no diretório da URL, podemos encontrar o primeiro ingrediente (flag-1):

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

What is the second ingredient in Rick’s potion?

Explorando a máquina, encontramos um arquivo chamado “second ingredients” (flag-2) no diretório “/home/rick”. Porém, ao tentar usar o comando “cat” para exibir o conteúdo do arquivo, a página diz que o comando citado está desabilitado:

Uma imagem contendo Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Podemos driblar essa limitação usando o comando “less”, que é semelhante ao comando “cat” ao exibir o conteúdo de arquivos de texto, mas que possui opções como barra de rolagem para melhor visualização:

Aplicativo

Descrição gerada automaticamente com confiança baixa

(observação: é necessário usar a barra invertida para sinalizar que há um espaço no nome do arquivo, diferentemente da barra padrão “/”).

Parte superior do formulário

What is the last and final ingredient?

Comumente, a flag final está contida no diretório “/root”. Na qual, apenas quem possui permissão de root (administrador) poderá acessá-la.

Então, podemos verificar a nossa permissão de root com o comando “sudo -l”:

Interface gráfica do usuário, Texto, Aplicativo, Email, Teams

Descrição gerada automaticamente

Vemos que o sistema não exige senha para nenhum tipo de comando sudo que for digitado. Portanto, podemos exibir e entrar no diretório “root”:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Encontramos o terceiro e último ingrediente (flag-3), basta usar o comando “less” novamente para exibir o conteúdo do arquivo de texto:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente com confiança média

Parte superior do formulário

**Parte superior do formulário**

Conclusão

Esse desafio pode ser uma grande porta de entrada para interessados em *web hacking* por ser fácil e divertido. Além disso, o CTF prepara o participante para enfrentar questões comuns para desafios web. Na qual, pode ser ótimo para aprimorar e testar as ferramentas de pentesting e resolver o desafio de diversas maneiras.

Referências

* <https://book.hacktricks.xyz/network-services-pentesting/pentesting-web>
* <https://wfuzz.readthedocs.io/en/latest/user/basicusage.html>
* <https://hackertarget.com/gobuster-tutorial/>