Logotipo

Descrição gerada automaticamente com confiança baixa

Relatório de CTF

Brooklyn Nine Nine – TryHackMe

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | Brooklyn Nine Nine – Isabella Passarelli |
| **N° Revisão** | 1 |
| **Data de publicação** | 01/09/2025 |
| **Link** | <https://tryhackme.com/room/brooklynninenine> |

|  |  |  |
| --- | --- | --- |
| **Redação** | Isabella Passarelli | Estudante |
| **Revisão** | Nome do revisor | Orientador |
| **Aprovação** | Nome do aprovador | Diretor |

|  |  |  |
| --- | --- | --- |
| **Histórico de revisões** | | |
| **N°** | **Entregas** | **Descrição** |
| **0** | 01/09/2025 | Produção |
| **1** | DD/MM/AAAA | Revisão |
| **2** | DD/MM/AAAA | Aprovação |

|  |  |
| --- | --- |
| **Informações do CTF** | |
| **Nível de Dificuldade** | Fácil |
| **Tipo de acesso** | Gratuito |
| **Conceitos envolvidos** | FTP, hydra, SUID |
| **Plataforma** | Tryhackme |
| **Área** | Red |

**Sumário**

[Contextualização 3](#_heading=h.gjdgxs)

[Desenvolvimento 3](#_heading=h.1fob9te)

Pergunta 1 - [User flag 3](#_heading=h.3znysh7)

Pergunta 2 - [Root flag 4](#_heading=h.2et92p0)

[Conclusão 5](#_heading=h.1t3h5sf)

[Referências 5](#_heading=h.4d34og8)

**Contextualização**

Esse CTF tem como finalidade ser um desafio introdutório na área de pentesting, abordando conceitos como hydra, FTP e SUID.

**Desenvolvimento**

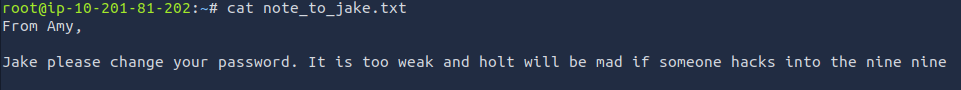
**User flag**

O comando **nmap -sC -sV 10.201.20.76** foi utilizado como uma etapa inicial de reconhecimento de rede. Essa é uma prática padrão em pentesting, onde o objetivo é obter informações sobre o alvo antes de tentar explorar vulnerabilidades.

Texto

O conteúdo gerado por IA pode estar incorreto.

Em seguida foi utilizado um conjunto de comandos para explorar um serviço de FTP encontrado no alvo. Primeiro, o comando **ftp 10.20.120.76** foi usado para iniciar a conexão com o servidor FTP na máquina do desafio. Em seguida, ao ser solicitado o nome de usuário, a conta **Anonymous** foi utilizada, que é uma tentativa comum para acessar servidores FTP sem a necessidade de credenciais. O campo de senha foi deixado em branco, o que, nesse caso, permitiu o acesso. Uma vez dentro do servidor, o comando **ls** foi usado para listar os arquivos disponíveis, onde o arquivo note\_to\_jake.txt foi encontrado. Para obtê-lo, o comando **get** foi executado, baixando o arquivo para a máquina local do atacante. Finalmente, após sair da sessão FTP com o comando bye, o conteúdo do arquivo foi lido localmente usando o comando **cat note\_to\_jake.txt**. A mensagem encontrada dizia: "From Amy, Jake please change your password. It is too weak and holt will be mad if someone hacks into the nine nine". Essa informação revela uma pista crucial para o desafio: a senha de um usuário, provavelmente "Jake", é fraca, o que sugere um próximo passo.



Como a pista da nota da Amy sugeriu que a senha do usuário “jake” era fraca, o comando **hydra -l jake -P /usr/share/wordlists/rockyou.txt 10.201.20.76 -t 4 ssh** foi utilizado para realizar um ataque de força bruta, que é a tentativa automática e massiva de senhas para obter acesso.

Teclado de computador

O conteúdo gerado por IA pode estar incorreto.

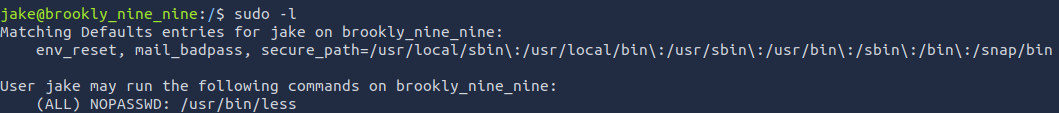
Após a senha fraca ser encontrada, o próximo passo foi utilizar o comando **ssh jake@10.201.20.76** para iniciar uma sessão segura como o usuário jake. Uma vez logado, o comando **pwd** mostrou o diretório atual do usuário jake. Em seguida, foi utilizado o **cd /home** para mudar para o diretório principal de usuários. Lá, o comando **ls** revelou a existência da pasta de outro usuário, o holt. Entrando na pasta com **cd holt** e rodando **ls** novamente, o arquivo user.txt foi encontrado. Por fim, o comando **cat user.txt** foi usado para exibir o conteúdo do arquivo, que continha a primeira flag do CTF: **ee11cbb19052e40b07aac0ca060c23ee**.

Texto

O conteúdo gerado por IA pode estar incorreto.

**Root flag**

Nessa etapa, o objetivo foi escalonar os privilégios de usuário para se tornar root. Para isso, o primeiro passo foi utilizar o comando **sudo -l**, que lista todos os comandos que o usuário atual tem permissão para executar como sudo. Ao rodar o comando, descobriu-se que o usuário podia executar o comando less com privilégios elevados. Para explorar essa vulnerabilidade, o site GTFOBins foi consultado, e nele foi encontrado um método para obter um shell de root usando o less. Em seguida, o comando **sudo less /etc/profile** foi executado para abrir um arquivo de sistema com privilégios de root. Dentro do less, o comando **!/bin/sh** foi utilizado, que abre um novo shell. Como o less estava rodando com sudo, o novo shell também foi aberto com os privilégios de root, garantindo o controle total sobre a máquina e a capacidade de encontrar a próxima flag.



Após obter acesso de root à máquina, foi utilizado o comando **whoami** para confirmar que agora era o superusuário. Em seguida, o comando **cd /root** foi utilizado para ir diretamente ao diretório principal do usuário root, que é o local padrão para armazenar arquivos importantes do sistema. Dentro desse diretório, o comando **ls** foi executado para listar o conteúdo. Por fim, o comando **cat root.txt** foi usado para ler o conteúdo do arquivo root.txt, revelando a flag final do desafio: **63a9f0ea7bb98050796b649e85481845**.

Texto

O conteúdo gerado por IA pode estar incorreto.

**Conclusão**

O desafio, com seu nível de dificuldade básico e rápido de resolver, seguiu uma lógica de testes de intrusão, começando com a enumeração de serviços, a busca por vulnerabilidades, o acesso inicial com credenciais fracas e, finalmente, a escalação de privilégios para obter controle total do sistema.

**Referências**

GTFOBINS. **Less: Shell**. 2019. Disponível em: <https://gtfobins.github.io/gtfobins/less/#shell>. Acesso em: 01 set. 2025.