Uma imagem contendo Logotipo

Descrição gerada automaticamente

Relatório de CTF

Brooklyn Nine Nine – TryHackMe

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | Brooklyn Nine Nine – Alexandre Gualiume Coruquieri |
| **N° Revisão** | 1 |
| **Data de publicação** | 22/10/2024 |
| **Link** | https://tryhackme.com/r/room/brooklynninenine |

|  |  |  |
| --- | --- | --- |
| **Redação** | Alexandre Gualiume Coruquieri | Estudante |
| **Revisão** | João Branco | Presidente |
| **Aprovação** | João Branco | Presidente |

|  |  |  |
| --- | --- | --- |
| **Histórico de revisões** | | |
| **N°** | **Entregas** | **Descrição** |
| **0** | 14/10/2024 | Produção |
| **1** | 22/10/2024 | Revisão |
| **2** | 22/10/2024 | Aprovação |

**Sumário**

[Contextualização 3](#_Toc179216044)

[Desenvolvimento 3](#_Toc179216045)

[User flag: 3](#_Toc179216046)

[Root flag: 5](#_Toc179216047)

[Conclusão 6](#_Toc179216048)

[Referências 6](#_Toc179216049)

Contextualização

Esse CTF pode ser considerado um bom desafio para introdução a área de *pentesting,* tendo em mente o nível dos desafios e o desenvolvimento para resolvê-los. Aborda temas como: Enumeração, ataque de força bruta e escalonamento de privilégios.

Desenvolvimento

User flag:

Primeiro, é necessário fazer a enumeração para coletar informações da máquina. Para isso, usa-se o **Nmap** com o parâmetro (*-A*) para torná-lo agressivo e procurar por vulnerabilidades:

Texto

Descrição gerada automaticamente

Com o resultado do **Nmap**, é possível observar que as portas 21, 22 e 80 estão abertas, rodando os serviços de FTP (Transferência de Arquivos), SSH (Conexão Remota) e HTTP (Protocolo de Transferência de Hipertexto) respectivamente. Além disso, a enumeração indica a existência da vulnerabilidade ‘*Anonymous’* no FTP, permitindo a conexão sem o uso de senha para autenticação.

Com isso, obtém-se o arquivo *‘note\_to\_jake.txt*’ a partir da conexão FTP:

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

O arquivo é uma dica dada por Amy para que Jake troque a senha. Com isso, é possível presumir que a senha de Jake esteja contida em uma *wordlist*, que é um arquivo de texto com as senhas mais utilizadas.

Então, após a identificação do possível usuário ‘jake’, o **Hydra** pode auxiliar a encontrar a senha para estabelecer conexão com o serviço SSH:

Texto

Descrição gerada automaticamente

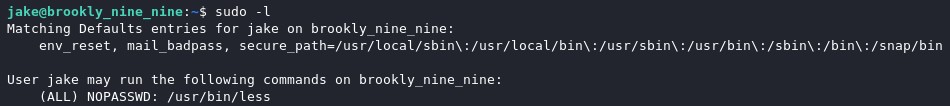
Com a senha, a conexão SSH poderá ser estabelecida e, assim, será possível encontrar a flag ‘user’ no usuário ‘holt’ da máquina:

**Texto

Descrição gerada automaticamente**

Root flag:

Para encontrar a flag de root, será necessário fazer um escalonamento de privilégios. O comando “sudo -l” mostrará quais as permissões que o usuário ‘jake’ possuí:



Ele tem a permissão para usar o comando ‘less’ sem a senha de root, esse comando permite visualizar arquivos, semelhante ao comando ‘cat’. Portanto, essa permissão pode conceder o escalonamento de privilégios para root, o site [*GTFOBins*](https://gtfobins.github.io/gtfobins/less/) contém os comandos necessários para isso.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Usando os comandos, obtém-se acesso ao root e levando a flag final:

Texto

Descrição gerada automaticamente

OBS.: Nesse CTF, a fase de escalonamento de privilégios pode ser pulada, visto que o usuário ‘jake’ há permissões de ‘less’ e então pode usar o comando “less /root/root.txt”. Porém, por fins didáticos, considere a resolução acima.

Conclusão

O desafio possuí um nível de dificuldade básico e rápido de resolver, mesmo assim, se demonstra divertido do início ao fim. Um CTF perfeito para treinar as ferramentas de *pentesting* e aprimorar a teoria sobre o assunto.

Referências

* <https://book.hacktricks.xyz/linux-hardening/linux-privilege-escalation-checklist>
* <https://gtfobins.github.io/>
* <https://book.hacktricks.xyz/network-services-pentesting/pentesting-web>