Uma imagem contendo Logotipo

Descrição gerada automaticamente

Relatório de CTF

Mustacchio – TryHackMe

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | Mustacchio – Sérgio Guidi Trovo |
| **N° Revisão** | 1 |
| **Data de publicação** | 15/09/2024 |
| **Link** | https://tryhackme.com/r/room/mustacchio |

|  |  |  |
| --- | --- | --- |
| **Redação** | Sérgio Guidi Trovo | Estudante |
| **Revisão** | Nome do revisor | Orientador |
| **Aprovação** | Nome do aprovador | Diretor |

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

**Sumário**

[Contextualização 2](#_Toc177050160)

[Desenvolvimento 3](#_Toc177050161)

[Pergunta 1- What is the user flag? 3](#_Toc177050162)

[Pergunta 2 - What is the root flag? 9](#_Toc177050163)

[Conclusão 11](#_Toc177050164)

[Referências 11](#_Toc177050165)

Contextualização

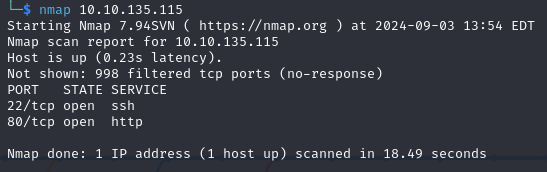
O CTF Mustacchio é um CTF de nível de dificuldade fácil que aborda etapas importantes do processo de pentesting. Esse CTF tem foco em decriptação, XXE e privilege escalation.

Desenvolvimento

Pergunta 1- What is the user flag?

**Parte superior do formulário**

Começamos o CTF realizando um escaneamento do servidor com o comando nmap.



Encontramos duas portas abertas: 22 e 80.

Acessando o serviço web encontramos a seguinte página:

Imagem em preto e branco de mulher com óculos de grau

Descrição gerada automaticamente com confiança baixa

Realizamos uma enumeração de diretórios com o comando gobuster.

Calendário

Descrição gerada automaticamente

Encontramos dois diretórios interessantes: /custom e /robots.txt. Acessando /robots.txt não encontramos nenhuma informação importante. Acessando /custom encontramos dois outros diretórios: css e js.

Tabela

Descrição gerada automaticamente

No diretório js encontramos outros dois diretórios: mobile.js e users.bak.

Tabela

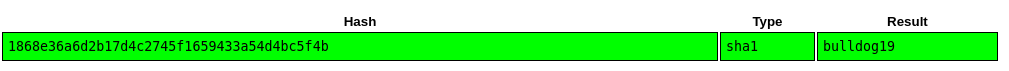
Descrição gerada automaticamente

Fazendo o download de users.bak, descobrimos que se trata de um arquivo no formato de uma database. Podemos utilizar o comando sqlite3 para abrir e manipular esse arquivo.

Texto

Descrição gerada automaticamente

Explorando a database encontramos a tabela *users* com o usuário admin e sua senha. Enviando a senha encontrada para um decodificador de hashes encontramos a senha original.



Ao tentar conectar com a máquina via ssh com as credenciais encontradas, percebemos que não é permitido.



Fazendo um nmap de todas as portas do servidor encontramos uma porta secreta rodando o serviço *ultraseek-http*.

Texto

Descrição gerada automaticamente

Acessando o serviço encontrado, nos deparamos com uma página de login.

Interface gráfica do usuário

Descrição gerada automaticamente

Com as credenciais encontradas conseguimos fazer login e acessar a página chamada “ADMINPANEL”, que nos permite enviar comentários para o site.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Inspecionando a página encontramos um comentário interessante.

Também encontramos uma file chamada *dontforget.bak*.



Acessando a file encontrada, encontramos um texto no formato xml.

Texto

Descrição gerada automaticamente

Ao tentarmos enviar esse texto na sessão de comentários da página *ADMINPANEL*, percebemos que ele é refletido na página.

Texto

Descrição gerada automaticamente

Isso nos indica que o site pode ser vulnerável a um ataque XXE (external Entity injection). Como já sabemos o formato de envio aceito do documento xml, basta mudar o conteúdo das tags para executar o exploit.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Enviando o xml acima, conseguimos ter acesso ao conteúdo da file /etc/passwd e verificar os usuários do servidor. Encontramos os usuários barry e joe.



Como no comentário encontrado quando inspecionamos o site era dito que o usuário Barry podia logar com a chave ssh, podemos tentar encontrar a chave ssh do usuário Barry para nos conectar com o servidor. Para isso, acessamos o conteúdo da file /home/barry/.ssh/id\_rsa.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Executando o comando apresentado, temos a senha do usuário barry refletida no site.

Texto

Descrição gerada automaticamente

Podemos utilizar *john the ripper* para encontrar a passkey para a chave encontrada. Primeiro convertemos a chave para um arquivo que pode ser lido pela ferramenta utilizando o comando *ssh2john*. Em seguida, utilizamos o comando john na file convertida para encontrar a passkey.

Texto

Descrição gerada automaticamente

Fazendo isso, encontramos a passkey “urieljames”.

Utilizamos a chave e a passkey encontradas para nos conectar com o servidor via ssh.

Texto

Descrição gerada automaticamente

Dentro da máquina, encontramos a flag de usuário.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

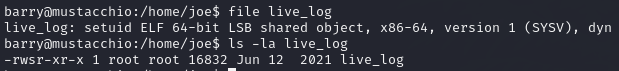
Pergunta 2 - What is the root flag?

Explorando o servidor, encontramos o diretório do outro usuário, joe. Entrando nesse diretório, encontramos uma outra file chamada live\_log.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Verificando as propriedades da file encontrada, descobrimos que se trata de uma file que pode ser executada como SUID.



Ao executar essa file, percebemos que ela exibe as requisições realizadas na página http secreta encontrada anteriormente.

Texto

Descrição gerada automaticamente

Utilizando o comando *strings*, encontramos que a file live\_log utiliza o comando tail para acessar outra file nomeada access.log.





Podemos tentar criar uma file com o nome tail para ser executada pelo servidor ao invés do binário tail já existente. Para fazer isso, precisamos adicionar ao servidor a path do local onde a nova file será criada de modo a garantir que ela será visitada antes da file já existente.



Agora, podemos criar a nova file tail e adicionar o comando /bin/bash.



Executando a file live\_log, conseguimos ganhar as permissões de usuário root e encontramos a flag de root completando o CTF.

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

Conclusão

O CTF Mustacchio é muito interessante para aprender conceitos importante de web exploitation. Embora seja um CTF fácil, ele possui algumas etapas complexas, principalmente na parte de escalação de privilégio.

Referências

<https://www.exploit-db.com/>

<https://gtfobins.github.io/>

<https://book.hacktricks.xyz/?q=f>

<https://portswigger.net/web-security/xxe>