Uma imagem contendo Logotipo

Descrição gerada automaticamente

Relatório de CTF

dailybugle – TryHackMe

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | dailybugle – Enrico Santarelli |
| **N° Revisão** | 1 |
| **Data de publicação** | 20/08/2024 |
| **Link** | https://tryhackme.com/r/room/dailybugle |

|  |  |  |
| --- | --- | --- |
| **Redação** | Enrico | Membro |
| **Revisão** | Nome do revisor | Orientador |
| **Aprovação** | Nome do aprovador | Diretor |

|  |  |  |
| --- | --- | --- |
| **Histórico de revisões** | | |
| **N°** | **Entregas** | **Descrição** |
| **0** | 20/08/2024 | Produção |
| **1** | DD/MM/AAAA | Revisão |
| **2** | DD/MM/AAAA | Aprovação |

**Sumário**

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

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

[What is the Joomla version? 3](#_heading=h.3znysh7)

[What is Jonah’s cracked password? 3](#_heading=h.2et92p0)

[What is the user flag? 3](#_heading=h.tyjcwt)

[What is the root flag? 3](#_heading=h.3dy6vkm)

[Conclusão 3](#_heading=h.1t3h5sf)

[Referências 3](#_heading=h.4d34og8)

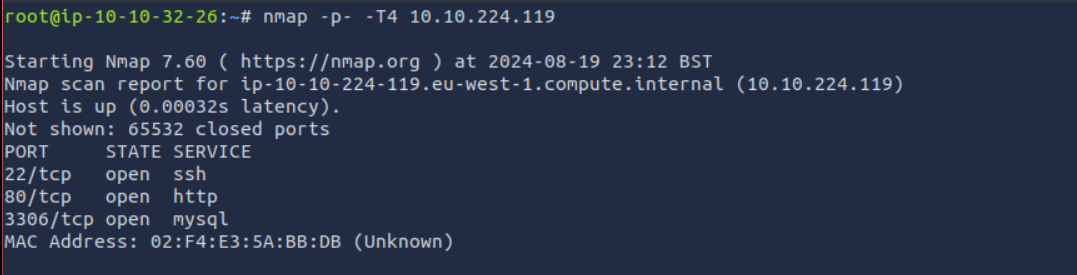
**Contextualização**

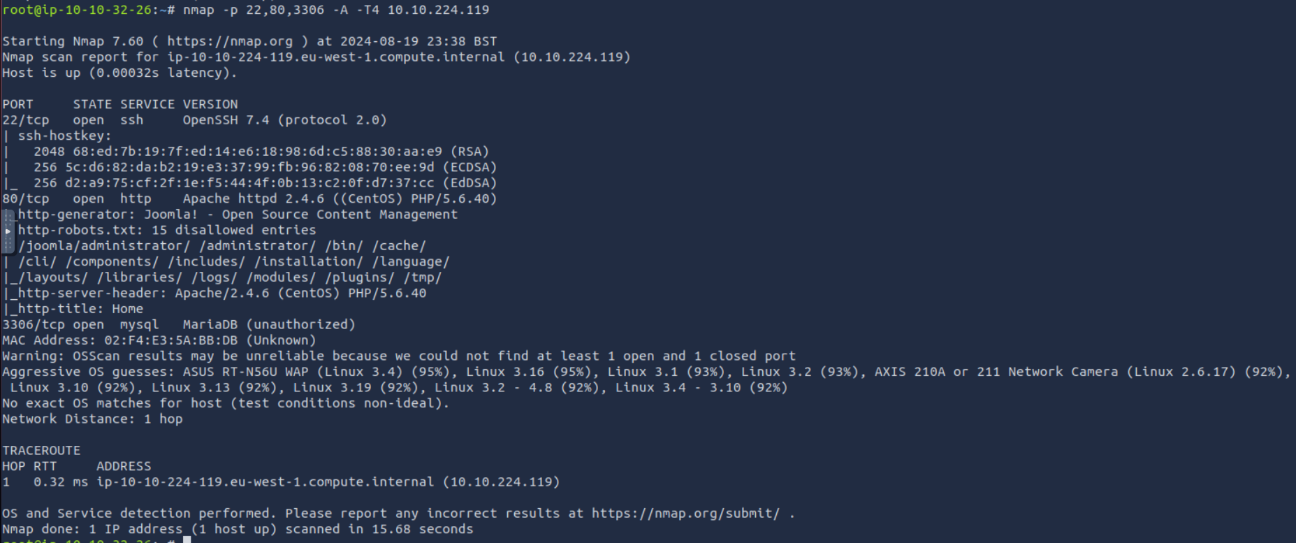
A sala é de nível médio, aborda diversas frentes do pentesting. As principais abordagens são: SQLi, hashs, exploração web e CVE. O escalonamento de privilégio da máquina é a parte mais simples da sala.

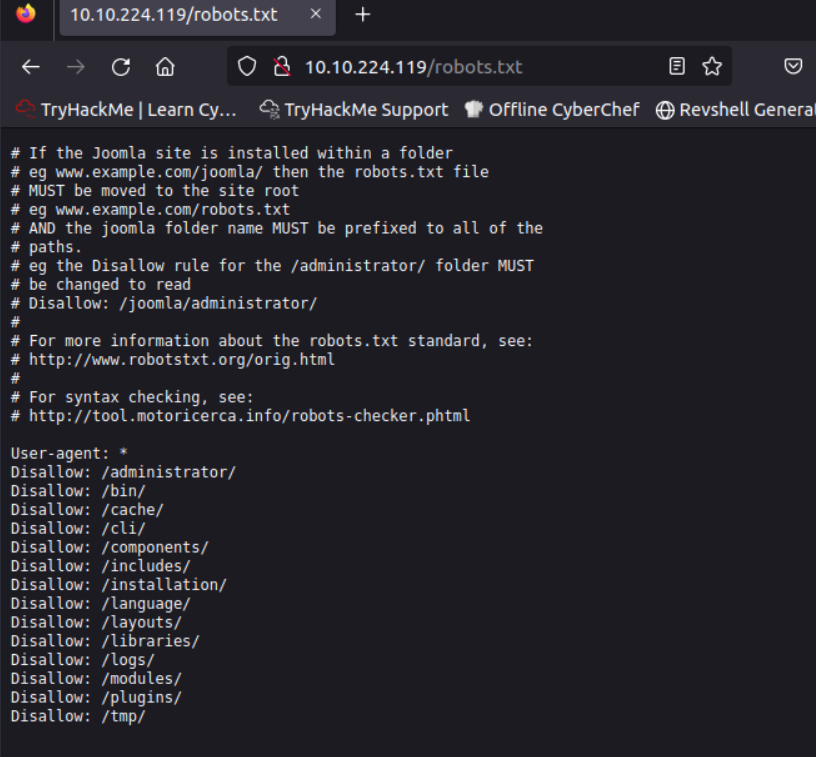
**Desenvolvimento**

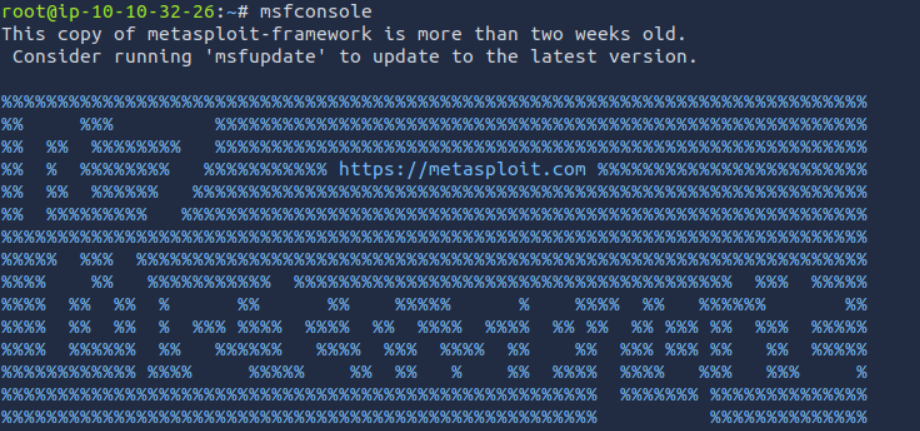
**What is the Joomla version?**

Primeiro, vamos realizar um escaneamento de portas utilizando o Nmap. Para procurar em todas as portas possíveis vamos utilizar a flag -p-.

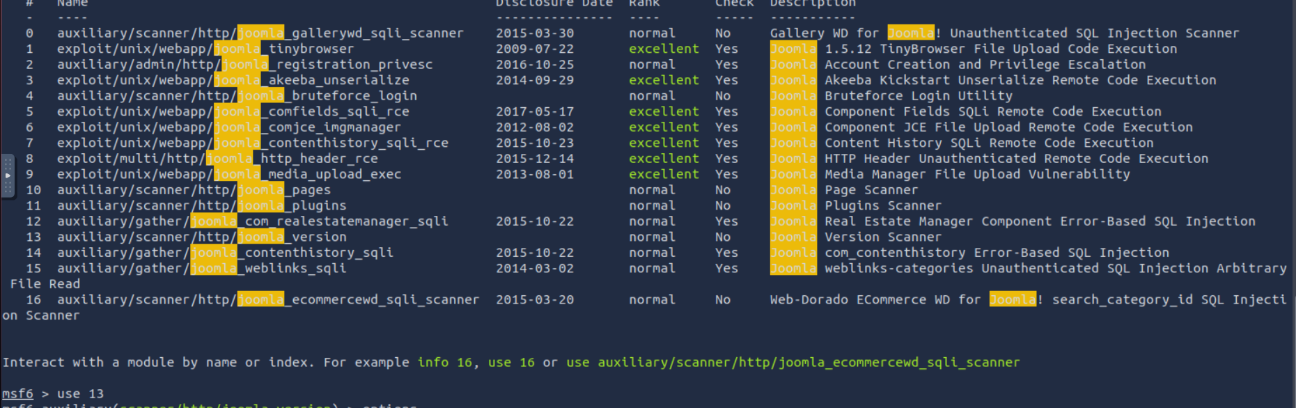
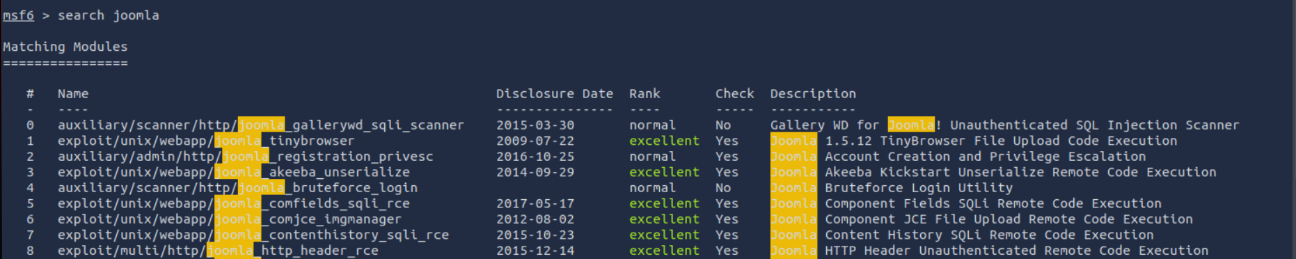


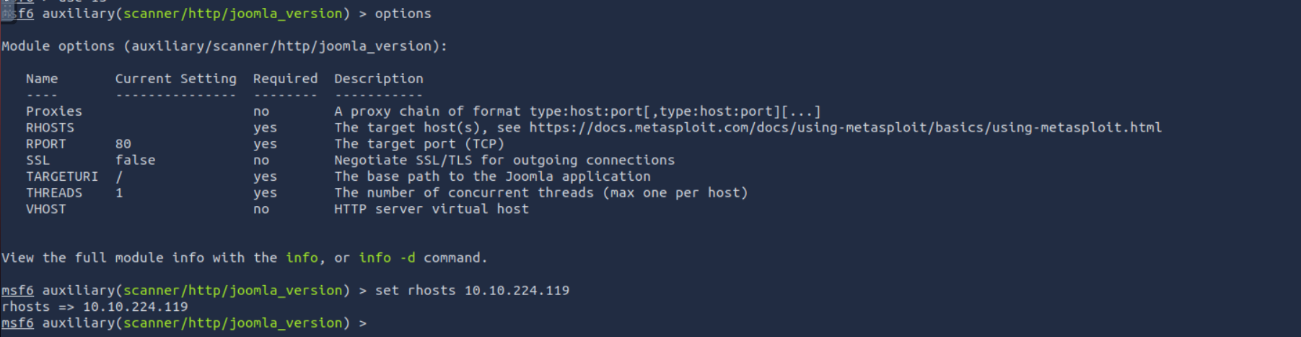
Em seguida, exploraremos mais a fundo as portas abertas, procurando por suas versões, sistema operacional e informações. Para isso, utilizamos a flag -A.

A primeira coisa perceptível é a presença de um arquivo robots.txt acessível. Vamos guardá-lo pois pode ser importante futuramente.

Contudo, a versão do Joomla não foi exposta pelo mapeamento, logo vamos buscar na ferramenta Metasploit algo que possa ajudar.

Procurando por joomla encontramos diversos módulos disponíveis, porém o que nos interessa é o escaneador de versão.



Setando as opções necessárias e rodando o exploit, obtemos a versão desejada.

**Interface gráfica do usuário, Texto

Descrição gerada automaticamente**

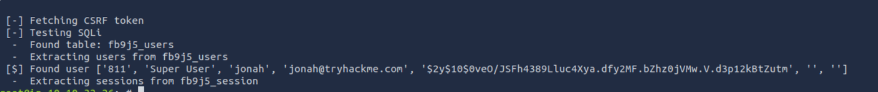
**What is Jonah's cracked password?**

Para a versão do joomla utilizada existe um script chamado joomblah que realiza testes utilizando SQL injection. Ref: <https://github.com/stefanlucas/Exploit-Joomla/blob/master/joomblah.py>.

Rodando o script com python, obtemos um usuário e uma senha em formato de hash.

Tela de computador

Descrição gerada automaticamente



Utilizando John The Ripper para quebrar o hash, obtemos então a senha do usuário.

Texto

Descrição gerada automaticamente

**What is the user flag?**

Agora, utilizando o path /administrator encontrado no robots.txt no início e as credenciais adquiridas, podemos nos logar como administrador no painel do joomla.

Interface gráfica do usuário

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo, Email, Site

Descrição gerada automaticamente

Como administrador podemos realizar um RCE para obter uma shell. A ideia é trocar o código php da página de erro e substituir por uma reverse shell. Ref: <https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/joomla>

[https://www.revshells.com](https://www.revshells.com/)

**Interface gráfica do usuário, Aplicativo, Site

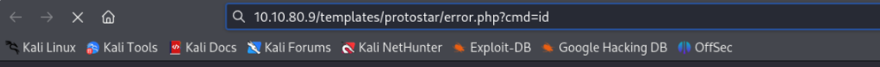
Descrição gerada automaticamente**

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

****

**Texto

Descrição gerada automaticamente**

Obtida a shell, nós agora estamos na máquina virtual com o usuário apache. Para procurar uma forma de escalar esse usuário, vamos usar o script linpeas.

**Texto

Descrição gerada automaticamente**

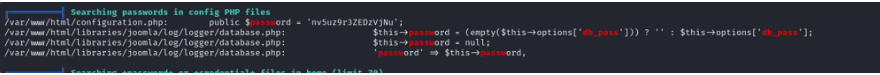
**Uma imagem contendo Gráfico

Descrição gerada automaticamente**

Existem dois pontos que chamam atenção no resultado do script: o usuário jjameson e uma senha exposta.

Texto

Descrição gerada automaticamente

****

Testando a mudança para o usuário jjameson com a senha encontrada nós conseguimos escalar para o usuário.

**Texto

Descrição gerada automaticamente**

Logo, na pasta home do usuário está a primeira flag.

Texto

Descrição gerada automaticamente com confiança média

**What is the root flag?**

Por fim, para escalonar os privilégios, rodamos o comando “sudo -l” para verificar os comandos que o usuário pode rodar sudo. De cara, nos deparamos com o comando yun, que poderia ser executado como sudo sem senha.



Logo, apenas rodando uma sequência de comandos encontrados no gtfobins nós obtemos acesso ao usuário root. Ref: <https://gtfobins.github.io/gtfobins/yum/#sudo>

Texto

Descrição gerada automaticamente

A segunda está na pasta /root na raiz do projeto.

Texto

Descrição gerada automaticamente

**Conclusão**

Esse CTF é classificado pela plataforma como nível difícil, porém, acredito que ele está no nível médio. Eu indicaria esse CTF para aqueles que já tem um conhecimento sólido das bases de pentest em CTFs e querem se desafiar.

**Referências**

* <https://github.com/stefanlucas/Exploit-Joomla/blob/master/joomblah.py>.
* <https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/joomla>
* [https://www.revshells.com](https://www.revshells.com/)
* <https://gtfobins.github.io/gtfobins/yum/#sudo>