Logotipo

Descrição gerada automaticamente com confiança baixa

Relatório de CTF

Capacitação 6 – Explorando vulnerabilidades

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | Road THM – João Vitor Ferrenha |
| **N° Revisão** | 1 |
| **Data de publicação** | 27/10/2025 |
| **Link** |  |

|  |  |  |
| --- | --- | --- |
| **Redação** | João Vitor Ferrenha | 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 |

|  |  |
| --- | --- |
| **Informações do CTF** | |
| **Nível de Dificuldade** | Médio |
| **Tipo de acesso** | Gratuito |
| **Conceitos envolvidos** | Bruteforce,wordlist e webhacking |
| **Plataforma** | TryHackMe |
| **Área** | Red |

**Sumário**

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

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

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

[Pergunta 2 3](#_heading=h.2et92p0)

[Pergunta 3 3](#_heading=h.tyjcwt)

[Pergunta 4 3](#_heading=h.3dy6vkm)

[Conclusão 3](#_heading=h.1t3h5sf)

[Referências 3](#_heading=h.4d34og8)

**Contextualização**

Este relatório documenta, em ambiente controlado e exclusivamente para fins educacionais, o processo de comprometimento da máquina **Road** (TryHackMe). Adota-se sequência lógica, reconhecimento com **Nmap**, enumeração de diretórios com **Gobuster**, navegação dirigida na aplicação web, exploração para obtenção de **execução remota de comandos** e estabilização de sessão**, pós-exploração** com coleta de credenciais/serviços e, por fim, **escalada de privilégios** até a captura das flags. Todas as ações foram realizadas dentro do escopo autorizado do desafio.

**Desenvolvimento**

1. Reconhecimento e enumeração:

O trabalho iniciou-se pelo reconhecimento de portas e serviços com Nmap, a fim de estabelecer a superfície exposta e priorizar vetores de entrada. Foi executada varredura de detecção de versões e scripts padrão, e observou-se a presença de serviço HTTP e de serviço SSH como componentes principais do alvo. Esses achados determinaram a estratégia inicial: priorizar a exploração de HTTP para obtenção de acesso e manter o SSH como via potencial a ser revisitadas caso surgissem credenciais durante a pós-exploração.

Texto

O conteúdo gerado por IA pode estar incorreto.

Confirmada a disponibilidade de HTTP, procedeu-se à enumeração de diretórios e rotas ocultas com o Gobuster, utilizando wordlist adequada para ambientes web. Essa etapa revelou caminhos relevantes para investigação manual, notadamente diretórios associados a funcionalidades administrativas e o diretório de uploads acessível a partir da aplicação. Os resultados do Gobuster forneceram um roteiro objetivo para a fase seguinte: cada rota foi validada no navegador, buscando-se compreender o fluxo da aplicação, os controles de autorização implementados no front-end e eventuais pistas deixadas no código cliente.

Tela de computador com texto preto sobre fundo branco

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

1. Exploração Web:

Feita a análise usando as ferramentas de enumeração e reconhecimento o próximo passo é começar a interagir com a aplicação em si. Ao mexer um pouco nas páginas não foi achado nada que fosse valioso para exploração. Mas como é possível ver no resultado do Gobuster tem-se algumas páginas interessantes:

Interface gráfica do usuário, Site

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Aplicativo, Teams

O conteúdo gerado por IA pode estar incorreto.

A primeira delas é uma página de login para o php, ao tentar algumas senhas padrão nota-se que a aplicação a princípio está utilizando um banco de dados mysql (pela mensagem de erro). A segunda página revela uma página de login na aplicação em si, ao criar a conta teve-se acesso a seguinte página:

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Ao navegar pelas páginas da aplicação foi achado um possível ponto de exploração, o envio de fotos de perfil (que é permitido apenas para administradores):

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

Portanto, para isso precisou-se pensar numa maneira de logar como administrador, isso foi feito da seguinte maneira. Há na aplicação uma página onde pode-se alterar a senha da conta logada (possível superfície de ataque).

Interface gráfica do usuário, Aplicativo, Teams

O conteúdo gerado por IA pode estar incorreto.

Ao interceptar a requisição feita para o back-end foi possível notar que eram passados como parâmetros na própria requisição o usuário e a senha. Então dessa maneira a requisição foi alterada para mudar a senha da conta de administrador:

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Logado como administrador pode-se realizar o envio de arquivos, como o backend é em php (segundo a tela de login que temos do php) foi preparado um payload com uma reverse shell, com o objetivo de fazer o servidor se conectar à minha máquina:

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

Ao interceptar a requisição pode-se ver que o conteúdo do payload foi passado no corpo da requisição:

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Resposta:

Texto

O conteúdo gerado por IA pode estar incorreto.

O servidor aceitou o arquivo enviado, o próximo passo seria executar essa reverse shell. Foi feito um fuzzing manual tentando achar diretório de upload, mas sem sucesso. Foi então que depois de algum tempo decidi olhar melhor a resposta http e foi quando:

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Um comentário escondido dizendo o diretório em que as fotos eram salvadas (/v2/profileimages), nesse ponto começou-se a exploração.

1. Exploração:

Ao colocar a minha máquina para esperar conexões usando o netcat e entrar no diretório da minha imagem (/v2/profileimages/payload.php), foi feita a conexão reversa:

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Achou-se a primeira flag de usuário, agora o próximo passo era escalar os privilégios atrás da flag de root.

Texto

O conteúdo gerado por IA pode estar incorreto.

Segundo o /etc/psswd nota-se dois usuários, o root e o webdeveloper. Existe também o mysql e o mongodb no sistema.

Texto

O conteúdo gerado por IA pode estar incorreto.

Tentando achar um bit suid para escalar privilégios, mas não era o caminho. Comecei então a navegar pelos diretórios da máquina em busca de achar algo que eu pudesse logar nos bancos de dados.

Texto

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Foi achado um arquivo que expõe o usuário e senha do banco, dessa maneira o próximo passo era logar:

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Foi possível notar que o mysql realmente estava guardando os dados da aplicação, mas nada mais valioso que isso, anda que ajudasse a escalar os privilégios, então partiu-se para o mongodb.

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Achou-se as credenciais do usuário webdeveloper e então conectou-se via ssh na máquina:

Texto

O conteúdo gerado por IA pode estar incorreto.

Texto

O conteúdo gerado por IA pode estar incorreto.

Pode-se notar que o usuário pode rodar esse executável de backup, podendo ser possível escalar os privilégios. Ao checar com o comando “strings”:

Texto

O conteúdo gerado por IA pode estar incorreto.

Constatou-se a possibilidade de executar o utilitário de backup do sistema (/usr/bin/sky\_backup\_utility) com privilégios elevados sem solicitar senha, e — ponto crucial — com preservação da variável de ambiente LD\_PRELOAD. Essa preservação constitui a falha de configuração que viabiliza a escalada: quando um binário é iniciado via sudo e a política não limpa LD\_PRELOAD, o carregador dinâmico (ld.so) injeta a biblioteca compartilhada indicada antes de transferir o controle ao programa-alvo. Dessa forma, código controlado pelo atacante é executado no contexto de root. Em condições normais, binários SUID/SGID ignoram LD\_PRELOAD por segurança; contudo, quando a elevação ocorre por sudo com ambiente incorretamente preservado, o bloqueio padrão é contornado pela própria política do sudo.

O caminho mais prático foi preparar uma biblioteca compartilhada maliciosa que, ao ser pré-carregada, abrisse uma shell com privilégios de superusuário

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Com a biblioteca pronta, a exploração consistiu em **injetá-la** ao iniciar o utilitário: sudo LD\_PRELOAD=/tmp/exploit.so /usr/bin/sky\_backup\_utility. A execução resultou em shell com uid=0 e, na sequência, foi possível acessar o diretório /root e capturar a flag de root.





**Conclusão**

O comprometimento foi alcançado por meio de cadeia coerente e reprodutível: o Nmap delineou a superfície; o Gobuster converteu serviços expostos em rotas investigáveis; a navegação sustentou a exploração web que resultou em execução remota e estabilização de sessão; a enumeração local forneceu credenciais válidas para SSH; e a configuração inadequada de sudoers, ao preservar LD\_PRELOAD, viabilizou a escalada até root. Como medidas de mitigação, recomenda-se: (i) validação rigorosa de uploads; (ii) higiene de segredos e logs, evitando persistir credenciais; (iii) revisão de políticas do sudo para impedir a preservação de variáveis de ambiente críticas; e (iv) endurecimento do sistema operacional, restringindo pré-carregamento em execuções privilegiadas e auditando binários/rotinas invocados com privilégios elevados.

**Referências**

Hacking Articles - <https://www.hackingarticles.in/linux-privilege-escalation-using-ld_preload>

GTFOBins - https://gtfobins.github.io