**Jurassic Park**

**TryHackMe**



Começando o desafio, fazemos uma enumeração com o nmap e descobrimos somente as portas 22 e 80 abertas. Com isso, vamos tentar explorar a aplicação web.

Texto

Descrição gerada automaticamente

Então sabendo disso, podemos entrar no site da porta 80 para tentarmos explorarmos.

Tela de celular com publicação numa rede social

Descrição gerada automaticamente

Entrando nele temos uma tela de início com o tema do Jurassic Park, o que deixa o CTF bem interessante.

Fazendo mais enumerações na aplicação, conseguimos acessar o diretório /assets que está com directory listening.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Não achamos nada de interessante nele, apenas alguns vídeos e imagens temáticas.

Então seguindo, analisamos o código fonte da aplicação e nele descobrimos o cominho para o shop.

Texto

Descrição gerada automaticamente

Quando entramos no shop, conseguimos selecionar um produto e somos redirecionados diretamente para uma página que informa sobre ele.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Podemos ver que existe um parâmetro chamada “**id**” e estamos atualmente no 3, então sabendo disso vamos tentar um SQL Injection básico.

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

Demos o comando sleep na aplicação e fomos retornados para uma página troll que nos bloqueou.

Então podemos tentar jogar essa requisição no sqlmap para ver se conseguimos algo.

Uma imagem contendo Texto

Descrição gerada automaticamente

Na nossa requisição colocamos as tags –random-agent (pois somos bloqueados que usamos o user-agent do sqlmap) e especificamos o banco de dados MySQL, pois se trata de uma aplicação em php.

Com isso, depois de um tempo conseguimos chegar em um resultado.

Texto

Descrição gerada automaticamente

Vimos que a aplicação realmente é vulnerável à SQL Injection, então agora podemos começar a exploração enumerando toda a base.

**Databases:**

Texto

Descrição gerada automaticamente

**Tables:**

Texto

Descrição gerada automaticamente com confiança baixa

**Dump:**

Diagrama

Descrição gerada automaticamente

Além disso, vimos que os usuários foram omitidos, porém como o próprio CTF nos diz, existe um usuário chamado dennis.

Agora para seguirmos com a exploração, vimos anteriormente que existe o ssh habilitado na aplicação, então podemos tentar nos conectar com o usuário “**dennis**” e uma das senhas encontradas.

Texto

Descrição gerada automaticamente

Conseguimos então nos conectar utilizando a senha “**ih8dinos**”.

Agora que temos acesso ao servidor, vamos começar a explorá-lo para obtermos as flags.

Texto

Descrição gerada automaticamente

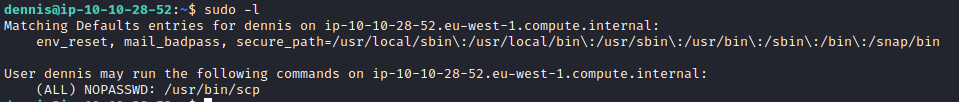
Encontramos a primeira flag no diretório do usuário dennis e nele também vimos um arquivo interessante chamado “**test.sh**”.

Tela preta com letras brancas

Descrição gerada automaticamente

Vimos que existe a flag 5 no diretório do root, mas não há nada mais que possamos fazer com esse script por enquanto.

Vamos então tentar escalar nosso privilégio no sistema e para começar vamos ver se temos algum comando que podemos dar como sudo.



Executando o sudo -l, vimos que podemos usar o scp como sudo, então agora é muito simples escalarmos nosso acesso.

Pesquisando no GTFObins, vimos os comandos que podemos dar para escalar nosso acesso com o scp.

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Então replicamos e pronto, facilmente escalamos o acesso.

Texto

Descrição gerada automaticamente

Agora o próximo passo foi conseguir uma shell interativa para não ficarmos muito travados na linha de comandos. Para fazer isso utilizamos o python.



Agora como tínhamos visto antes, podemos rodar o script “**test.sh**” para pegarmos de maneira mais simples a flag 5.

Texto

Descrição gerada automaticamente

Agora continuando a busca pelas 2 flags restantes, conseguimos achar a terceira no histórico de comandos do dennis “**.bash\_history**”.

Texto

Descrição gerada automaticamente

Agora para acharmos a segunda flag realizamos uma busca recursiva que nos auxiliou um pouco.

Texto

Descrição gerada automaticamente

Então tendo seu diretório, conseguimos pegar a flag facilmente.



Fazendo isso concluímos com sucesso o CTF do Jurassic Park.