Título: Exposição de Token JWT da API do TheMovieDB no Front-End Público

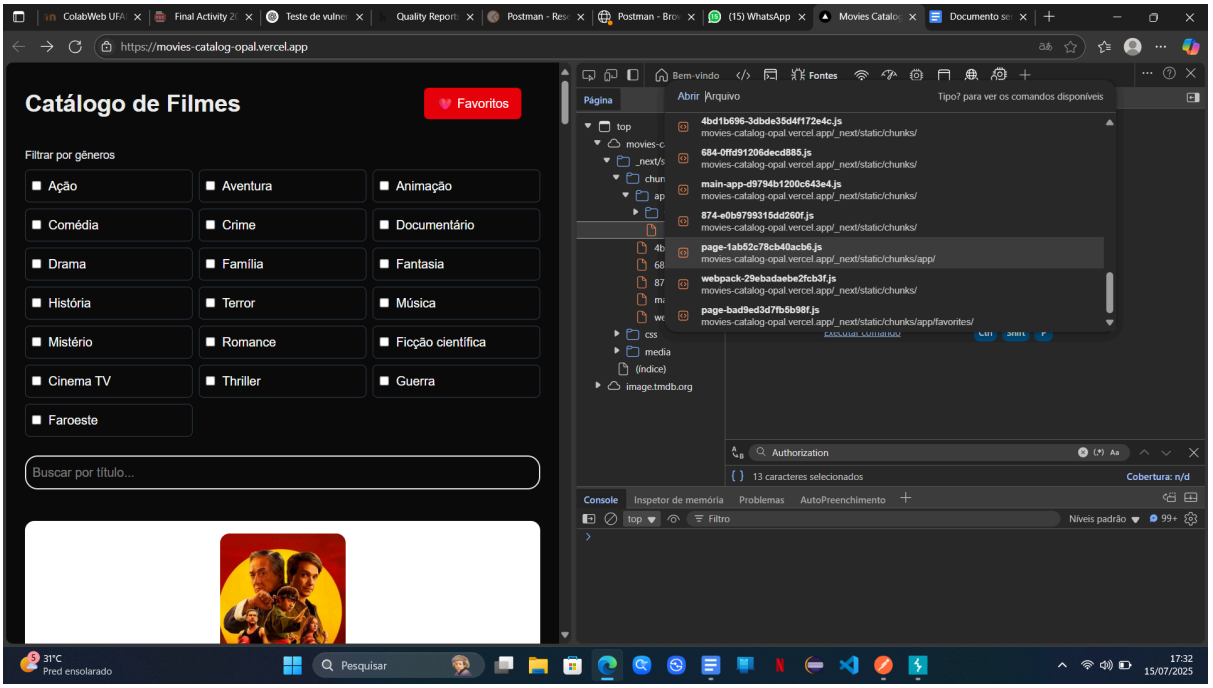
Site: [Movies Catalog]

Resumo:
A aplicação `https://movies-catalog-opal.vercel.app` incorpora diretamente um **token JWT da API do TheMovieDB** no código JavaScript acessível ao público. Isso permite que qualquer usuário reutilize esse token para fazer requisições autenticadas como se fosse a aplicação, violando os princípios de segurança de APIs públicas.

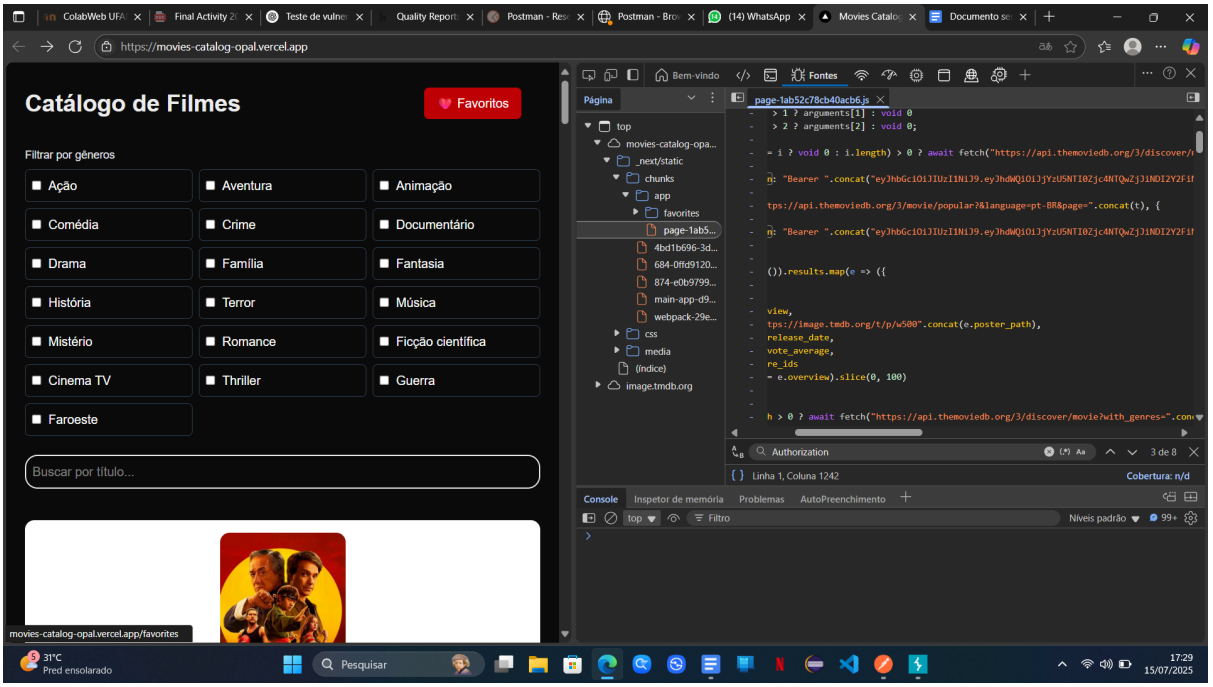
PASSO-A-PASSO:

1-Acesse o site `https://movies-catalog-opal.vercel.app`

2-Abra o DevTools com `F12`, vá em **Sources**, e abra `page-1ab52c78cb40acb6.js`

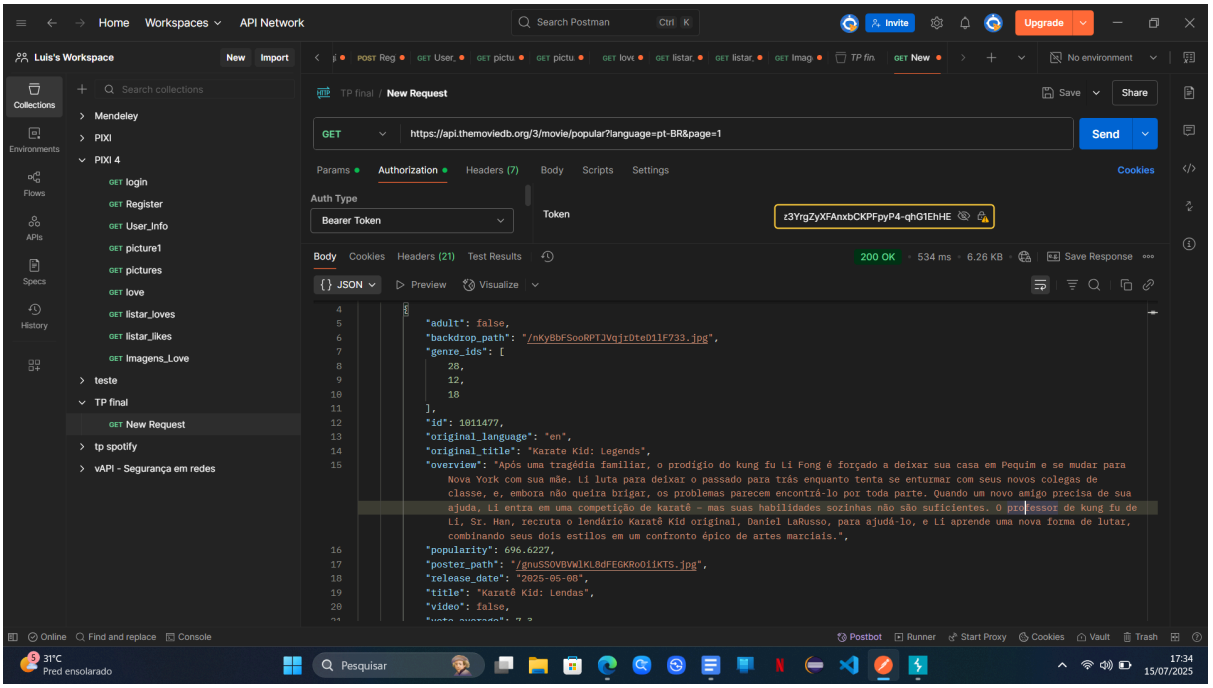


3-Procure pelo termo `Authorization`



4-Copie o token JWT presente nas chamadas `fetch()`

5-No Postman, envie a seguinte requisição:



6 - A resposta será `200 OK` com a lista de filmes — validando o uso direto do token público sem proteção.

Gravidade sugerida: Alta

Categoria OWASP: API2 / API9

Título: Ausência de Limite de Requisições na API (Lack of Rate Limiting)

Gravidade: Alta (uso indevido, custo à API de terceiros, rate limit, abuso)
Categoria OWASP:

- API9: Improper Assets Management
- API2: Broken Authentication (se reutilizável indevidamente)

Descrição:
A aplicação Movies Catalog embute diretamente no frontend um token JWT da API do TheMovieDB. Isso permite que qualquer usuário reutilize esse token e envie requisições autenticadas em nome da aplicação, comprometendo a conta da API, podendo atingir limites ou realizar abusos.

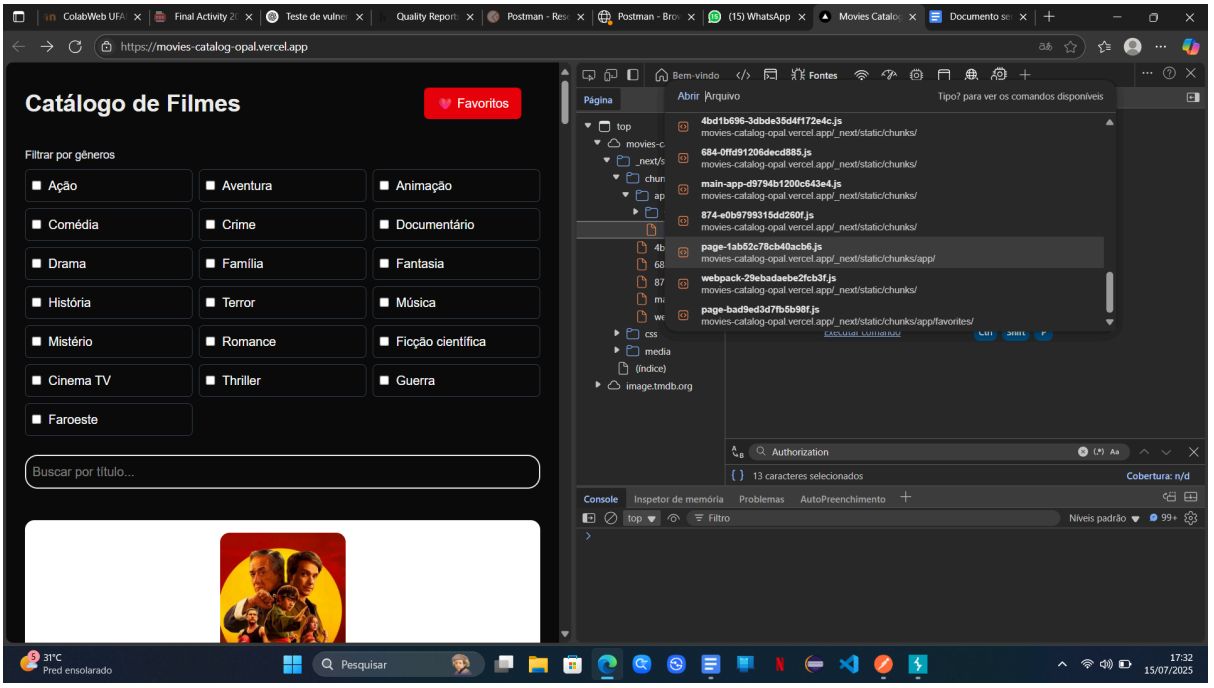
API afetada: TheMovieDB (TMDb)

Tipo de vulnerabilidade:

- Exposição de credenciais (API9 – Improper Assets Management)
- Falta de controle de requisições (API4 – Lack of Rate Limiting)

1. Obtenção do Token

- **Acesse o site: `https://movies-catalog-opal.vercel.app`**
- **Pressione F12 → aba Sources**



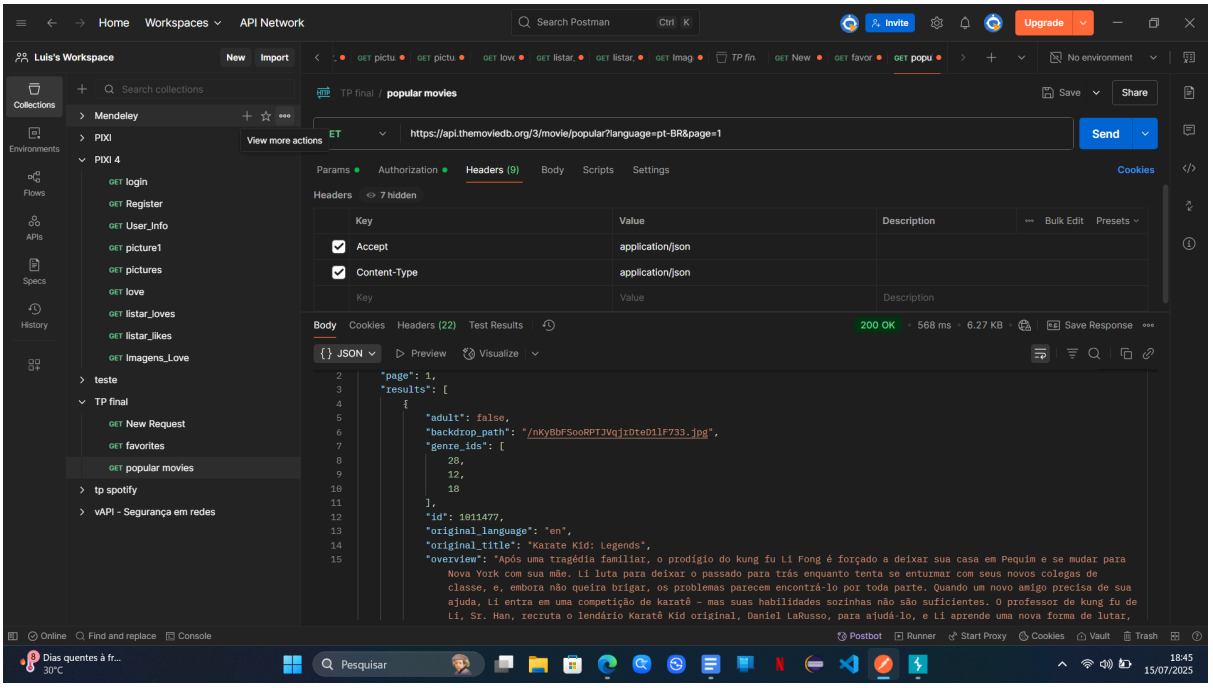
- **Localize o script `page-xxx.js`**

- **Busque por `"Authorization"` e copie o token JWT exposto:**

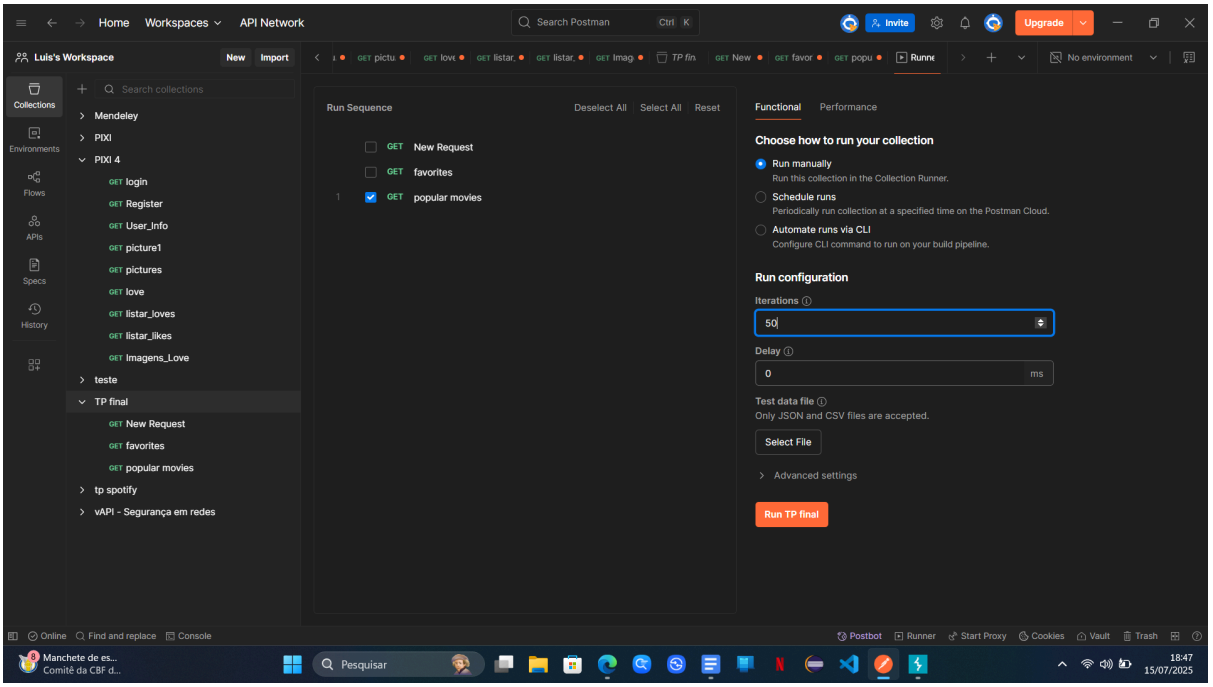
Token encontrado:

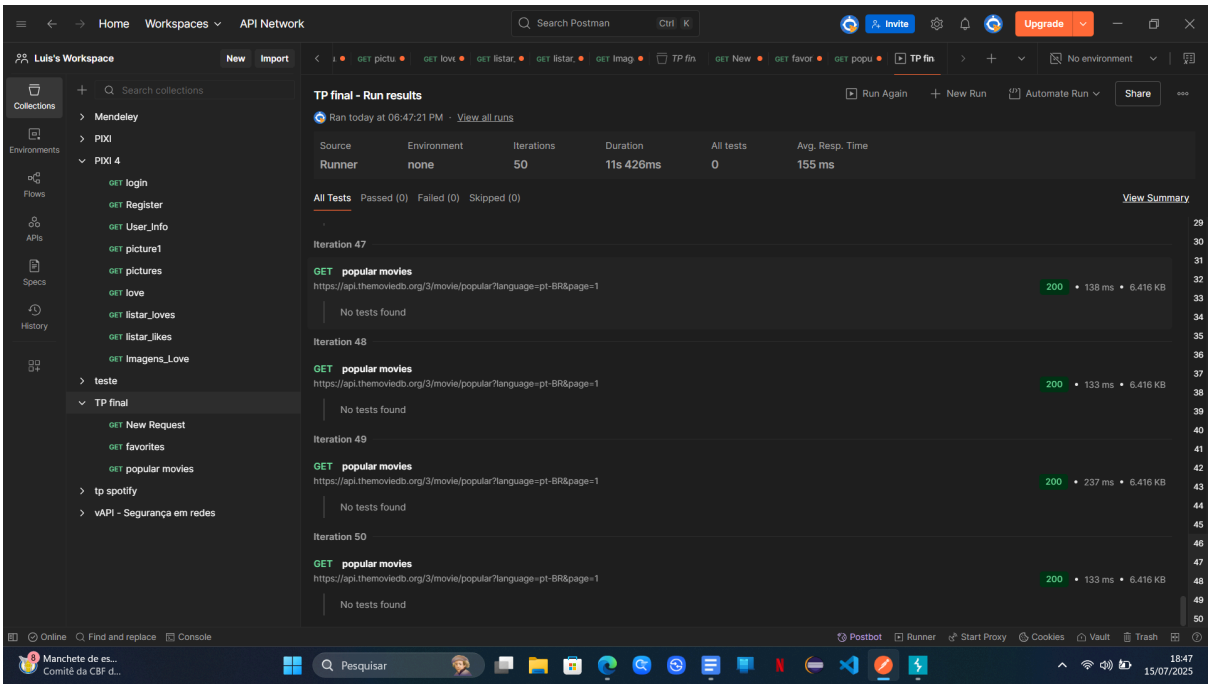
```
eyJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJjYzU5NTI0Zjc4NTQwZjJiNDI2Y2FiNTA0NmJkYmVlMiIsIm5iZiI6MTc0NjY0ODAzNi43NjksInN1YiI6IjY4MWJiYmU0MTk2MDBlZjcyYTAxZTU5MCIsInNjb3BlcyI6WyJhcGlfcmVhZCJdLCJ2ZXJzaW9uIjoxfQ.Vbf5hzWVsReuUz3YrgZyXFAnxbCKPFpyP4-qhG1EhHE
```

2. Teste direto via Postman



3. Simulação de abuso com Runner (ataque DoS leve)





Resultado:

- **100% das respostas foram `200 OK`**
- **Nenhum `429 Too Many Requests`**
- **Tempo médio: ~150ms por requisição**

Conclusão

1. **Exposição de Token no Front-End – A presença do token JWT diretamente no código JavaScript permite que qualquer usuário reutilize esse token, o que viola a confidencialidade da autenticação. Essa prática configura a vulnerabilidade API2 – Broken Authentication.**
2. **Má Gestão de Ativos Sensíveis – O token está disponível publicamente sem controle, indicando falta de segregação entre componentes seguros e inseguros, caracterizando API9 – Improper Assets Management.**
3. **Ausência de Rate Limiting – Durante o teste com múltiplas requisições, nenhum erro de limite foi retornado (ex: 429), o que expõe a API a possíveis abusos, como ataques de negação de serviço leves (DoS). Isso também se relaciona com a API4 – Lack of Resources & Rate Limiting.**
4. **Risco para o Terceiro (TMDb) – Como o token pertence a uma conta da API do TheMovieDB, o uso indevido pode gerar custos, bloqueios ou suspensão da conta, afetando o proprietário da chave.**
5. **Impacto Geral – A combinação dessas falhas compromete a segurança, confiabilidade e controle de acesso da aplicação, evidenciando a necessidade de melhores práticas na proteção de APIs públicas.**