*** Kauê Soares Dos Santos - 824117267**
*** Leonardo Macedo Camargo - 82422817**
*** Luiz Washington de Jesus Muraro - 824148694**
*** Lucas Felipe Monteiro Suarez - 824138683**
*** George Geronimo Menezes Ferreira - 824148488**

Vulnerabilidades: Os funcionários que utilizam dos próprios aparelhos para se conectar a redes fora da empresa se apresentam como um risco para vulnerabilidade da empresa, falta de redes segmentadas por setores ou outros critérios pode facilitar que invasores acessem áreas da empresa que deveriam ser acessadas apenas por funcionários autorizados, pouca preocupação na verificação de dispositivos secundários (como no vídeo que a backdoor do cracker foi o termostato da empresa que não foi verificado, sendo assim um meio para retornar), senhas padrões de dispositivos de fábricas sendo facilmente descobertas por uma simples pesquisa na internet (senhas fáceis também facilitam a entrada do invasor).

Possíveis vulnerabilidades do sistema

Falhas de controle de origem: A política de mesma origem (Same-Origin Policy) é uma medida de segurança dos navegadores que impede que scripts de um domínio acessem dados de outro. No entanto, se essa política for burlada ou mal configurada, um iframe malicioso pode ser usado para acessar ou manipular dados de uma página da web legítima.

Inserção de conteúdo externo inseguro: Algumas aplicações web permitem que administradores ou usuários incorporem conteúdo externo (como vídeos ou outros recursos de sites terceiros) sem verificarem a origem ou o conteúdo desses recursos. Isso pode ser explorado se o conteúdo externo for controlado ou comprometido por um atacante.

Permissões de conteúdo flexíveis: Um site que permite a inclusão de HTML em comentários, perfis ou outras seções dinâmicas pode, inadvertidamente, abrir espaço para a injeção de iframes maliciosos.

Tipos e técnicas de ataques utilizados

Ataque de injeção I-frame: Um cracker injeta um código <iframe> malicioso em um site vulnerável, muitas vezes aproveitando falhas como XSS (Cross-Site Scripting) ou fraquezas em plugins desatualizados (como exemplificado no vídeo, onde o cracker usa um site de boliche para roubar as informações dos funcionários que acessam o site).

Motivação do Cracker

A motivação do cracker no começo foi a curiosidade, mas logo depois de ver os arquivos da empresa ele percebeu que poderia vender aquelas informações para a concorrência por 75 bitcoins.

1. Falta de sanitização e validação de entrada (Cross-Site Scripting - XSS)

- **Como ocorre**: Quando uma aplicação web permite que dados do usuário (como em campos de formulários, comentários ou URLs) sejam inseridos e exibidos sem uma sanitização adequada, atacantes podem injetar código HTML ou JavaScript, incluindo <iframe>, no conteúdo da página.
- **Exemplo**: Um site permite que os usuários postem comentários e não valida ou escapa os caracteres HTML no comentário. Um atacante pode postar um comentário que contém um código <iframe> malicioso, que será exibido a outros usuários, carregando conteúdo malicioso.

2. Falhas de Cross-Site Scripting (XSS) refletido

- **Como ocorre**: Em ataques XSS refletidos, o script malicioso é incluído em um link enviado a um usuário. Quando o usuário clica nesse link, o site legítimo reflete o conteúdo malicioso (incluindo um iframe) de volta para o navegador do usuário.
- **Exemplo**: O atacante cria um link para o site legítimo que contém parâmetros com um código <iframe> malicioso. Quando o usuário clica nesse link, o site reflete o código sem sanitização, permitindo que o iframe seja injetado e exibido ao usuário.

3. Falhas de XSS armazenado

- **Como ocorre**: Em XSS armazenado, o código malicioso (como um iframe) é permanentemente armazenado em um banco de dados ou em outra parte persistente da aplicação web. Sempre que outro usuário acessar essa parte do site, o código malicioso será executado.
- **Exemplo**: Um atacante insere um iframe malicioso em um campo de perfil ou comentário que é salvo no banco de dados da aplicação. Quando outros usuários acessam esse perfil ou comentário, o iframe é carregado.

4. Inserção de conteúdo externo inseguro

- **Como ocorre**: Algumas aplicações web permitem que administradores ou usuários incorporem conteúdo externo (como vídeos ou outros recursos de sites terceiros) sem verificarem a origem ou o conteúdo desses recursos. Isso pode ser explorado se o conteúdo externo for controlado ou comprometido por um atacante.

- **Exemplo**: Um site permite que os administradores ou usuários incorporem vídeos de sites de terceiros via iframe. Se a origem do iframe não for verificada ou se o site externo for comprometido, o iframe pode carregar conteúdo malicioso.

5. Falhas de controle de origem (Same-Origin Policy)

- **Como ocorre**: A política de mesma origem (Same-Origin Policy) é uma medida de segurança dos navegadores que impede que scripts de um domínio acessem dados de outro. No entanto, se essa política for burlada ou mal configurada, um iframe malicioso pode ser usado para acessar ou manipular dados de uma página da web legítima.
- **Exemplo**: Um atacante pode tentar inserir um iframe malicioso que manipula dados de uma sessão de usuário em um domínio vulnerável.

6. Permissões de conteúdo flexíveis

- **Como ocorre**: Um site que permite a inclusão de HTML em comentários, perfis ou outras seções dinâmicas pode, inadvertidamente, abrir espaço para a injeção de iframes maliciosos.
- **Exemplo**: Um blog que permite que os usuários insiram HTML em suas postagens ou comentários, mas não limita o uso de iframes, pode ser explorado para carregar conteúdo malicioso.