MINISTÉRIO DA EDUCAÇÃO SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA NORTE DE MINAS GERAIS CAMPUS JANUÁRIA.

BOAS PRATICAS PARA GARANTIR A SEGURANÇA DAS INFORMAÇÕES.

Januária-MG, 30 de Março de 2023.

Resumo.

Segurança de informações e o assunto principal quando se trata de desenvolvimento de software e aplicações web pois se o desenvolvedor criar uma rede social, por exemplo, por mais inovadora que essa rede social seja se ela não tiver o fator segurança ela não ira pra frente pois nenhum usuário vai ter vontade de usar um software que tem a fama de vazar dados. Quando uma empresa de grande porte tem seu sistema invadido as ações dessa empresa tendem a cair além do prejuízo para restaurar o sistema e consertar erros que levaram esse sistema a ser invadido. Este trabalho tem a intenção de mostrar alguns métodos que invasores usam para invadir sistemas, como injeção de código e engenharia social e alguns métodos para tentar prevenir essas invasões.

SUMARIO

Lista De Figuras.

1.INTRODUÇÃO

Com o crescimento do uso de aplicações e ferramentas web que utilizam dados pessoais para a sua utilização tem se tornado cada vez mais importante e necessário desenvolver sistemas com o máximo de segurança possível, para isso e importante o desenvolvedor ter o conhecimento das boas praticas.

Boas praticas não envolve somente a parte da programação em si ela também envolve como o desenvolvedor trata sua máquina e seu ambiente de trabalho e o treinamento de quem vai ter acesso de administrador do sistema, pois muitos ataques começam pela máquina dos administradores do sistema.

Existem diversos tipos de ataques e motivações para ataques podem ser por motivos políticos (difamação, discordâncias ideológicas), monetários (roubo e sequestro de dados e informações), ou por algum motivo pessoal. Existem ataques que exploram falhas no desenvolvimento do software como, injeção de SQL, CSRF e o XSS (Cross-Site Scripting). Também existem ataques do tipo de engenharia social que usam de pessoas para ter acesso a dados.

Para minimizar as chances de ataques as empresas e desenvolvedores devem se atentar para o uso de ferramentas que protejam seus sistemas, treinamento dos funcionários, testes e conscientização do uso.

1.1 Metodologia

Este artigo foi desenvolvido com caráter qualitativo, com a utilização de fontes de dados secundárias. Serão realizadas pesquisas bibliográficas baseadas em artigos da área de segurança da informação e outras publicações correlatas.

1.2 Objetivo Geral

Apresentar os riscos que uma aplicação Web pode sofrer devido as falhas no desenvolvimento, configuração de software e hardware ou vulnerabilidades internas.

1.2.1 Objetivos especifico:

Levantar os artigos sobre o tema;
Conhecer as principais tipos de vulnerabilidades;
Apresentar boas praticas no desenvolvimento de software;
Apresentar boas praticas no uso de rede e software da empresa ou do desenvolvedor;

2.Tipos de ataques.

Este tópico tem como intenção mostrar como os ataques funcionam e as vulnerabilidades da aplicação que estes ataques exploram, e mostrar algumas das boas praticas que podem ajudar a evitar esta situação.

2.1 Engenharia social.

Ataques do tipo de engenharia social são os mais praticados na atualidade pois são ataques tecnicamente mais fáceis de serem realizados, a engenharia social se utiliza do comportamento humano para obter acesso ao sistema.

> [...] Em vez de ficar se descabelando para encontrar uma falha no sistema, o hacker pode largar no banheiro um dispositivo de armazenamento infectado, com o logotipo da empresa e uma etiqueta bem sugestiva: 'Informações Confidenciais. Histórico Salarial 2003'. É provável que alguém o encontre e insira na máquina. (MITNICK,2003, p.273)

Existem várias técnicas de engenharia social elas se utilizam do poder de convencimento do invasor e da ingenuidade e falta de conhecimento e treinamento da vítima.

2.1.1 Phishing (Pescaria).

Essa e a técnica mais famosa de engenharia social ela pode se utilizar da curiosidade da vítima para fazer ela clica em um link de um e-mail ou anuncio de um produto que levam para um site malicioso ou fazer ela instalar ou executar um programa que contenham um vírus. Ou o invasor pode invasor pode enviar mensagens para funcionários da empresa se passando por algum administrador ate conseguir acesso a senhas de administrador e ter livre acesso ao sistema.

Um ataque danoso e que se utiliza da técnica phishing, e o malware ransowere, que e enviado através de mensagens que contenham um link ou através da execução de um programa.

Quando o sistema e infectado, o ransomware irá criptografar em segundo plano, arquivos da máquina. Assim que concluído o processo de encriptação, emitirá um aviso em tela informando sobre o bloqueio. Em seguida, um valor será exigido para obter uma chave a fim de restabelecer acesso aos arquivos criptografados.

2.1.2 Envenenamento de DNS.

De forma resumida o DNS age como um tradutor de domínio (link) para um endereço de IP, exemplo o usuário digita o nome do site: <ifnmg.edu.br/> o DNS transforma isso para o endereço de IP que a máquina e capaz de compreender : <18.231.88.217>.

> O DNS é formado por um sistema aprimorado que determina através de um protocolo da camada de aplicação dos modelos Open Systems Interconnection (OSI) e Transmission Control Protocol (TCP), a responsabilidade de administrar nomes de máquinas e endereços IP na rede e na internet(CARISSIMI.,2009).

O envenenamento de DNS ocorre quando um invasor tem acesso à rede de internet da pessoa ou da empresa e consegue alterar o IP de origem de um site para um IP alterado que clona o site original.

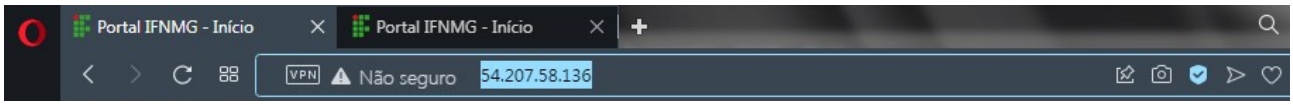

Figura 01: Endereço que não passou pelo processo de DNS. Fonte: Autor.

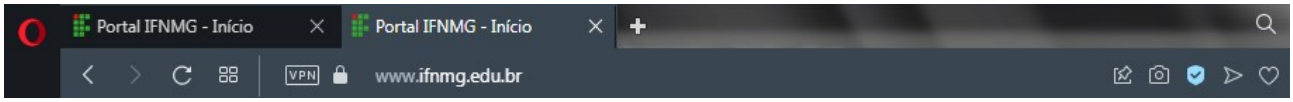

Figura 02: Endereço que passou pelo processo de DNS. Fonte: Autor.

Quando o navegador esta atualizado ele geralmente detecta quando o site não passa pelo processo de DNS ou quando o IP não tem certificado de segurança.

2.1.3 Boas praticas para evitar ataques do tipo: Engenharia Social.

Para empresas as melhores praticas para se evitar ataques e sempre treinar funcionários que tenham algum tipo de acesso a servidores e máquinas do local, Depois de um treinamento mostrando o que o funcionário deve evitar, a empresa depois de um tempo do treinamento deve testar o funcionário.

Não compartilhar a rede privada da empresa para pessoas que não são da empresa, se possível ter redes de internet e máquinas específicas para funcionários que tenham acesso a informações sigilosas e outras para o publico em geral, e caso um funcionário que tenham informações privilegiadas da empresa seja desligado trocar todas as senhas para evitar o vazamento de informações.

Para desenvolvedores e administradores a dica e não usar o computador que se usa para lazer para trabalhar e não compartilhar a rede doméstica com vizinhos ou pessoas que não são de extrema confiança. Usar navegadores atualizados e verificar se o site que você entra tem certificado de segurança.

2.2 Vulnerabilidade do software.

Quando o desenvolvedor não se atenta a detalhes do código durante o desenvolvimento ele abre brechas para possíveis invasões no seu sistema no futuro, pois o invasor se utiliza dessas vulnerabilidades para ter acesso a informações ou para derrubar um servidor.

2.2.1 XSS (Cross-Site Scripting).

Ataques XSS exploram falhas na validação de dados de um web-site. O invasor quando acha alguma falha de validação na entrada de dados que geralmente está em campos de busca ou de cadastro, ele e capaz de enviar um código malicioso que altere algo no site, assim ele pode copiar cookies, senhas, tokens ou roubar dados de acesso registrados no navegador web do usuário. Segundo a (OWASP 2022) existem três tipos de ataques XSS. Persistente, refletido e Baseados em DOM.

Os ataques Persistente são aqueles em que o script injetado é permanentemente armazenado nos servidores de destino, como em um banco de dados, em um fórum de mensagens, registro de visitantes, campo de comentários, etc. Em seguida, a vítima recupera o script malicioso do servidor quando solicita as informações armazenadas (OWASP).

Ataques refletidos são aqueles em que o script injetado é refletido fora do servidor web, como em uma mensagem de erro, resultado de pesquisa ou qualquer outra resposta que inclua alguma ou toda a entrada enviada ao servidor como parte da solicitação. Ataques refletidos são entregues às vítimas Usando tecnicas de Engenharia social, por outra rota, como em uma mensagem de e-mail ou em algum outro site. Quando um usuário é enganado para clicar em um link malicioso, enviar um formulário especialmente criado ou mesmo apenas navegar para um site malicioso, o código injetado viaja para o site vulnerável, o que reflete o ataque de volta ao navegador do usuário. O navegador então executa o código porque ele veio de um servidor "confiável" (OWASP).

Ataques baseado em DOM é um ataque XSS no qual a carga de ataque é executada como resultado da modificação do “ambiente” do DOM no navegador da vítima usado pelo script do lado do cliente original, de modo que o código lateral do cliente seja executado de forma “inesperada”. Ou seja, a página em si não muda, mas o código lateral do cliente contido na página é executado de forma diferente devido às modificações maliciosas que ocorreram no ambiente DOM. Ele e diferente dos outros ataques XSS no qual a carga de ataque é colocada na página de resposta (devido a uma falha no lado do servidor) (OWASP).

Para tentar evitar ataques XSS e nessesario tratar entradas do usuario antes de exbilas ou antes de armazenalas algumas linguagem tem alguns metodos prontos. No php existe o **htmlspecialchars()** que trata a entrada de dados.

```php
<?php
$user = $_GET['input'];
echo "Olá, " . $user. "!";
?>
```

No codigo demonstrado não existe nenhum tratamento e se o usuario escrever um script como por exemplo “<script>alert(“XSS”)</script>” a função alert() sera executada na aplicação.Você pode tratar isso da seguinte maneira usando o **htmlspecialchars()**.

```php
<?php
$user= $_GET['input'];
echo "Olá, " . htmlspecialchars($user) . "!";
?>
```

2.2.2 CSRF (Cross-Site Request Forgery).

O CSRF ataca a funcionalidade de alvo que causa uma alteração de estado no servidor,ataques de CSFR necessitam de uma sessão autenticada para alcançar o seu objetivo. Portanto, o atacante explora apenas sessões em que o usuário está logado em serviços web. Com a ajuda da engenharia social,o atacante envia um email contendo um link para a vítima. ao ser clicado

realiza uma requisição forjada para a aplicação web alvo. Como a vítima provavelmente estará autenticada e logada na aplicação alvo na hora do ataque, é impossível que a aplicação web alvo consiga distinguir entre uma requisição legítima de uma requisição forjada.

Para tentar previnir este tipo de ataque e necessario sempre verificar se a sessão do usuario e valida você pode fazer isso da seguinte maneira em PHP:

```
<?php
session_start(); // Inicia a sessão
$token = md5(uniqid(rand(), true)); // Gera um token CSRF
$_SESSION['csrf_token'] = $token; // Armazena o token na sessão
?>

<form action="delete.php" method="post">

  <input type="hidden" name="id" value="123">
  <input type="hidden" name="token" value="<?php echo $token; ?>">
<!-- Inclui o token  no formulário -->
  <input type="submit" value="Delete">

</form>
```

Neste exemplo, estamos gerando um token exclusivo e armazenando-o na sessão do usuário. Em seguida,inclui o token como um campo oculto no formulário. Quando o formulário é enviado, o token será enviado junto com a solicitação POST e pode ser verificado da seguinte maneira:

```
session_start(); // Inicia a sessão

if ($_POST['token'] !== $_SESSION['csrf_token']) {
        die("Erro: token  inválido");
}// Processa a solicitação de exclusão

[...]
```

Neste exemplo a condicional verifica se a diferença entre a session e o post caso aja ele irar matar a execução da aplicação.

2.2.3 Injeção de SQL.

Injeção de SQL, ou SQL, é um tipo de ataque a uma aplicação web que permite a um atacante inserir instruções SQL maliciosas na aplicação web, potencialmente ganhando acesso a dados sensíveis no banco de dados ou destruindo esses dados (Daityari 2022).

O invasor envia dados para o formulário e caso o desenvolvedor não tenha feito nenhum processo de validação ou de tratamento de dados o invasor consegue ter acesso à sessão de administradores e acesso total ao banco de dados. Um exemplo de falha de tratamento no back-end de uma aplicação em php e a seguinte:

```php
<?php
//obtém o valor da variável de consulta GET
$email = $_GET['email’];

//conexão com o banco de dados
$conn = mysqli_connect('localhost', 'root', 'senha', 'nome_bd’);

//consulta SQL para recuperar dados do banco de dados com base no valor da variável $email
$sql = "SELECT * FROM users WHERE email = $email";
$result = mysqli_query($conn, $sql);
?>
```

Neste codigo não a nenhum tratamento para a variavel “$email” que ira receber o “$_GET” uma variavel superglobal do php que é usada para coletar dados enviados por meio de um formulário HTML que se utiliza do método GET.Como não a nenhum tipo de tratamento um invasor pode manipular a consulta adicionando seu próprio código SQL. Por exemplo, se um invasor adicionar o código ' OR 1=1 -- ao final do valor da variável “$email” , que seria executado no SQL da seguinte maneira:

```
SELECT * FROM users WHERE email = '' OR 1=1 --
```

O que retornaria todos registros da tabela users pois o “WHERE” será sempre uma condição verdadeira devido à adição do “OR 1=1”.Uma das diversas formas de evitar isso no PHP e se utilizando desse exemplo:

```php
<?php
//obtém o valor da variável de consulta GET
$email = $_GET['email'];

//conexão com o banco de dados
$conn = mysqli_connect('localhost', 'root', 'senha', 'nome_bd');

//consulta SQL preparada para recuperar dados do banco de dados com base no valor da variável $email
$sql = "SELECT * FROM users WHERE email = ?";
$stmt = mysqli_prepare($conn,$sql);

//vincula o valor da variável $email à instrução preparada
mysqli_stmt_bind_param($stmt, 'i', $email);

//executa a consulta preparada
mysqli_stmt_execute($stmt);

//obtém o resultado da consulta
$result = mysqli_stmt_get_result($stmt);
?>
```

Neste exemplo de codigo, a consulta SQL é preparada usando uma instrução SELECT e um ponto de interrogação ? que é usado para indicar onde os valores dos parâmetros devem ser

inseridos. A função **mysqli_prepare** é usada para preparar a consulta e a função **mysqli_stmt_bind_param** é usada para vincular o valor da variável $email na instrução preparada. A consulta preparada é executada usando a função **mysqli_stmt_execute** e o resultado é obtido usando a função **mysqli_stmt_get_result**. Isso ajuda a prevenir a injeção SQL, pois os dados do usuário são separados da consulta SQL e são tratados corretamente.

2.2.4 Boas praticas para evitar a vulnerabilidade de software.

Para minimizar ataques que exploram as vulnerabilidades e necessário que o desenvolvedor conheça as principais falhas para saber como evitá-las. O desenvolvedor deve saber tratar dados de entrada e de saída durante a sessão o desenvolvedor deve entender que todos dados que entra no website são potencialmente perigosos.

Diversas recomendações para a prevenção de ataques podem ser extraídas do OWASP.

> O OWASP, ou Projeto Aberto de Segurança em Aplicações Web, é uma comunidade online que cria e disponibiliza de forma gratuita artigos, metodologias, documentação, ferramentas e tecnologias no campo da segurança de aplicações web.(Wikipédia).

OWASP recomenda utilizar biblioteca de codificação para facilitar o desenvolvimento e garantir que não haja falhas nas codificações dos dados não confiáveis.

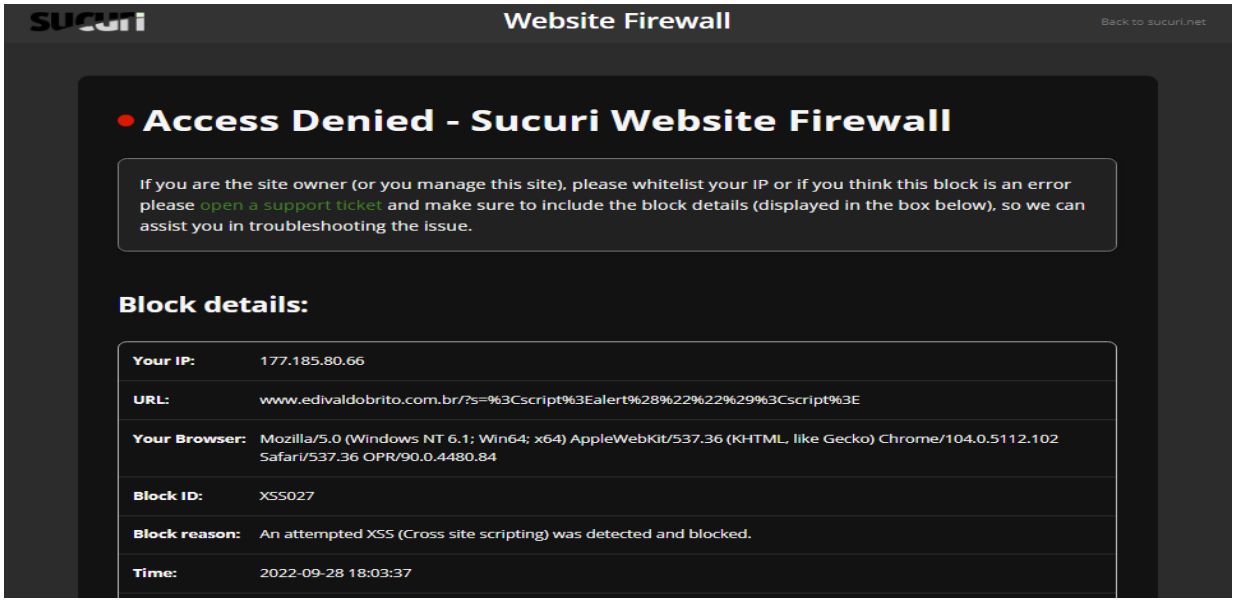

Figura 03 Exemplo de ferramenta(sucuri) que bloqueia tentativas de injeção de script Fonte: Autor

Existem também ferramentas que verificam vulnerabilidades (Figura 03) e elas podem ser usadas para escanear aplicações e obter informações atualizadas sobre o status de segurança das aplicações utilizadas pelos seus usuários.

3.Conclusão.

Se tratando da segurança da informação aprender todas as regras de segurança e fundamental isso vale para usuários empresas e desenvolvedor, pois um simples erro pode causar grandes prejuízos e no pior dos casos levar uma pessoa ou empresa a falência.

Para tentar prevenir esses prejuízos por falhas de segurança o desenvolvedor deve se atentar a regras de segurança durante o desenvolvimento e testar sua ferramenta ao máximo antes de vendê-la ou disponibilizar ela para o publico. E empresas deve manter sua equipe de segurança com o máximo de recursos possíveis e com ferramentas atualizadas e importante também a empresa ou desenvolvedor manter backups atualizados de dados separado de todas as outras máquinas pois caso seu sistema seja invadido e o invasor roube ou sequestre seus dados se a empresa tiver backups atualizados o processo de recuperação ficara mais fácil.

4.Referencias.

GRANA.H. **SEGURANÇA EM APLICAÇÕES WEB.** Disponível em : <http://ric.cps.sp.gov.br/bitstream/123456789/3732/1/20191S_GRANAHenriqueOD0663.pdf> Acesso em : 29 set. 2022.

Viana .S,Silva .R.F,Centro.P.J.,Laine .J.M. SEGURANÇA NO DESENVOLVIMENTO DE APLICAÇÕES WEB COM A QUALIDADE DOS DADOS **Revista de Sistemas e Computação,** Salvador, v. 3, n. 2, p. 93-104,jul./dez. 2013.

MENEZES.H, **Aspectos de Segurança para o desenvolvimento de aplicações web.** Disponível em : <https://www.webartigos.com/storage/app/uploads/public/588/508/0ba/5885080baf67b406967279.pdf>Acesso em : 29 set 2022.

OWASP Cross Site Scripting (XSS) Disponivel em: <https://owasp.org/www-community/attacks/xss/> Acesso em : 29 set 2022.

Daityari .S,**Injeção de SQL: Um Guia para Principiantes para Usuários do WordPress.** Disponível em : <https://kinsta.com/pt/blog/injecao-sql/> Acesso em : 29 set 2022.

Silva.T,Rosa .R.R. SEGURANÇA EM BANCO DE DADOS. **Colloquium Exactarum**, vol. 9, n. Especial, p. 61- 67.Jul–Dez, 2017.

OWASP **DOM Based XSS** Disponivel em : <https://owasp.org/www-community/attacks/DOM_Based_XSS> Acesso em : 29 set 2022.

Costa.S.G.L.,Cruz.S, **A ENGENHARIA SOCIAL E OS DESAFIOS DA SEGURANÇA DA INFORMAÇÃO NAS EMPRESAS.** Disponivel em: <http://repositorio.unis.edu.br/bitstream/prefix/431/1/A%20ENGENHARIA%20SOCIAL%20E%20OS%20DESAFIOS%20DA%20SEGURANÇA%20DA%20INFORMAÇÃO%20NAS%20EMPRESAS.pdf>Acesso em : 29 set 2022.