Tag de Web-Hacking

1. O que é o protocolo HTTP e Como ele funciona?

**HTTP** é sigla de HyperText Transfer Protocol que em português significa "**Protocolo** de Transferência de Hipertexto". É um **protocolo** de comunicação entre sistemas de informação que permite a transferência de dados entre redes de computadores, principalmente na World Wide Web (Internet)

1. O que é um Response Code? Cite um exemplo de um programa que você pode fazer com ele?

Os códigos de status das respostas HTTP indicam se uma requisição HTTP foi corretamente concluída. As respostas são agrupadas em cinco classes:

1. O que é um HEADER? Cite um uso INSEGURO desse cabeçalho.

Header é o cabeçalho do HTML O **elemento HTML** representa um grupo de suporte introdutório ou navegacional. Pode conter alguns elementos de cabeçalho mas também outros elementos como um logo, seções de cabeçalho, formulário de pesquisa, e outros.

Ouso inseguro deste cabeçalho é que deve se prestar bastante atenção para não seguir um site falso com propositos maliciosos

1. O que é um Método HTTP? Explique o funcionamento do método POST, o funcionamento do método GET. Explique qual é considerado mais seguro e por que.

O HTTP é o Protocolo de Hipertyexto usado pelos navegadores para acessar a internet

1. O que é Cache e como ele funciona? Cite os principais HEADERS de Request e Response

responsáveis pelo controle de Cache.

1. O que é Cookie? Qual é o principal ataque relacionado a ele?

Cookie consiste em um parametro (nome=> valor que representa a seção

Expires = data de valiade

Domain = dominio onde é realizado

Path = especifica o domínio onde é realizado

Segure = o cookie só sera transmitido via html

1. O que é OWASP-Top-Ten?

**OWASP**. O **OWASP** (Open Web Application Security Project), 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.

1. O que é Recon e Por que ela é importante?

9)Command Injection (SO-Injection)

a) O que é Command Injection?

Consiste em injetar comandos por meio de um parametro

b) Mostre um exemplo de Command Injection

XSS

1. SQL INJECTION
   1. O que é SQL injection?

Injeção de SQL (do inglês SQL Injection) é um tipo de ameaça de segurança que se aproveita de falhas em sistemas que interagem com bases de dados ...

* 1. O que é Union Based Attack?

1. Os ataques baseados em UNION permitem que o testador extraia facilmente informações do banco de dados. Como o operador UNION pode ser usado apenas se as duas consultas tiverem exatamente a mesma estrutura, o invasor deve criar uma instrução SELECT semelhante à consulta original. Para fazer isso, um nome de tabela válido deve ser conhecido, mas também é necessário determinar o número de colunas na primeira consulta e seu tipo de dados. Para simplificar o aprendizado, este artigo explica como isso pode ser feito quando o relatório de erros está ativado. No entanto, o mesmo princípio se aplicaria se não fosse o caso. Para mais informações, consulte a última seção do artigo.
   1. O que é Blind-SQL-I?

Descrição

A injeção cega de SQL (Structured Query Language) é um tipo de ataque de injeção de SQL que faz perguntas verdadeiras ou falsas ao banco de dados e determina a resposta com base na resposta dos aplicativos. Esse ataque geralmente é usado quando o aplicativo Web está configurado para mostrar mensagens de erro genéricas, mas não atenua o código vulnerável à injeção de SQL.

Quando um invasor explora a injeção de SQL, às vezes o aplicativo Web exibe mensagens de erro do banco de dados reclamando que a sintaxe da Consulta SQL está incorreta. A injeção cega de SQL é quase idêntica à injeção normal de SQL, a única diferença é a maneira como os dados são recuperados do banco de dados. Quando o banco de dados não gera dados para a página da web, um invasor é forçado a roubar dados, fazendo ao banco de dados uma série de perguntas verdadeiras ou falsas. Isso torna a exploração da vulnerabilidade de injeção SQL mais difícil, mas não impossível. .

d)Mostre um exemplo de um Blind

SQL-Injection

1. XSS
   1. O que é XSS?

É um tipo de command injecão

b)Quais são os tipos de XSS? Explique-os.

A exploração da vulnerabilidade XSS faz com que um atacante seja capaz de inserir scripts maliciosos em páginas que seriam confiáveis e usá-los para sequestrar o acesso de usuários e administradores.

* 1. Mostre um exemplo de um XSS Stored

O ataque consiste em utilizar uma página acessada para que nela se executute um script malicioso

D )Mostre um exemplo de um DOM-XSS .

Isto acontece quando um ajvascript processa informação de uma fonte não confiável

1. LFI , RFI e Path Traversal
   1. O que é LFI?

Local file inclusion

* 1. O que é RFI?

Remote file inclusion

* 1. O que é Path Traversal?

Consiste em escalar diretórios por meios de parametros mal sinalizados

d)Como aliar Path Traversal e LFI

e)Mostre um exemplo de LFI utilizando a

contaminação de LOGS

1. CSRF e SSRF

a) que e CSRF?

O cross-site request forgery (**CSRF** ou XSRF), em português falsificação de solicitação entre sites, também conhecido como ataque de um clique (one-click attack) ou montagem de sessão (session riding), é um tipo de exploit malicioso de um website, no qual comandos não autorizados são transmitidos a partir de um usuário

B) Mostre um exemplo de CSRF

MALWERE

c) O que e SSRF?

Server Side Request Forgery em uma tradução literal seria uma “Falsificação de solicitação via o lado do servidor”. O SSRF é uma vulnerabilidade na qual um atacante força um servidor a executar solicitações em seu nome. Aplicativos Web podem acionar solicitações entre servidores, que normalmente são usadas para buscar recursos remotos como, atualizações de software, importar dados de uma URL, etc. Embora essas solicitações entre servidores normalmente sejam seguras, elas podem tornar o servidor vulnerável à falsificação de solicitações do lado do servidor caso sejam implementadas incorretamente.

d)Mostre um exemplo de SSRF

Uma vez que o invasor não pode enviar solicitações diretas ao servidor da vítima, porque elas são bloqueadas por um firewall, para verificar uma rede interna o atacante precisa:

1. Envia uma solicitação ao servidor web vulnerável que se utiliza da vulnerabilidade SSRF.
2. O servidor web faz uma solicitação para o “servidor vítima” que fica atrás do firewall.
3. O “servidor vítima” responde com os dados.
4. Se a vulnerabilidade específica do SSRF permitir, os dados são enviados de volta ao atacante.

O ataque de SSRF pode gerar vários impactos para a vítima, porém, o resultado mais comum é a divulgação de informações, como:

* Digitalizar portas e endereços IP.
* Interagir com alguns protocolos como o Gopher, que permitem que o atacante faça descobertas adicionais sobre a rede interna da vítima.
* Descobrir os endereços IP dos servidores em execução atrás de um proxy reverso.
* Execução remota de código dentro do servidor utilizando shellcode.

Além dos exemplos acima, há várias outras coisas que os invasores podem fazer quando exploram uma vulnerabilidade SSRF, algumas das quais podem ter consequências mais graves, mas depende principalmente de como a aplicação web usa as respostas do recurso remoto.

e) Como evitar ataques de CSRF?

Para evitar vulnerabilidades do SSRF em suas aplicações Web, é altamente recomendável usar uma lista de permissões de domínios e protocolos de onde o servidor da Web pode buscar recursos remotos.

Além disso, como regra geral, você deve evitar usar a entrada do usuário diretamente em funções que podem fazer solicitações em nome do servidor. Você também deve desinfetar e filtrar a entrada do usuário, mas normalmente é muito difícil de configurar, principalmente porque é praticamente impossível cobrir todos os cenários diferentes.