Lorena Mamede Botelho
TAG: Web Hacking I

1) Protocolo HTTP é um conjuntos de regras para transmissão de hipermídia atuando na camada de Aplicação do modelo OSI. Estabelece métodos, headers e response codes para a comunicação cliente-servidor.

2) Response Code é um código retornado por uma requisição no cabeçalho de uma resposta. Esse código traz informações sobre a comunicação do cliente com o servidor. Por exemplo, se ocorrer tudo certo o código retornado é 200, significando sucesso, caso o servidor não consiga ofertar o serviço desejado retorna 500, Serviço Indisponível. Você pode criar uma aplicação com login, e usaria o response code como forma de verificação das atividades do usuário. Por exemplo, ao efetuar login, 200 seria a validação para autenticado e 401 (Não autorizado), para login inválido.

3) Header é o cabeçalho, podendo ser referente tanto à requisição quanto à resposta. Ele contém informações sobre essa comunicação. Na requisição, pode descrever o método, path, content-type etc. Na resposta, o response code, possivelmente cookies e uma série de outras informações. Esse header pode ser facilmente alterado para permitir por exemplo CMD injection, no qual o atacante acrescenta código em PHP no header para rodar comandos em shell script.

4) Um método HTTP é o modo pelo qual o cliente informa ao servidor o que deseja. Existem diversos métodos, GET e POST são os mais usados pelas aplicações. GET, o cliente pede uma determinada informação (na maioria das vezes, sem intenção de modificá-las) do servidor, geralmente sem parâmetro, quando com parâmetros, esses são passados via URL. POST, requisições com parâmetros passados por meio de campos da própria requisição, esses campos fornecem dados que (geralmente) modificarão informações. O último é considerado mais seguro pois omite os parâmetros (que muitas vezes podem ser dados sensíveis), enquanto o GET os deixa expostos na URL (podendo ser guardada no histórico do navegador, por exemplo) permitindo fácil acesso.

5) Cache é uma forma de guardar determinadas informações em memória, sem precisar recorrer ao servidor todas as vezes para obtê-la. Esse recurso pode ser controlado por Headers na requisições e respostas como Cache Control, Expires, Last-Modified e outros. Com eles podemos especificar por quanto tempo queremos que essa informação seja guardada, quando foi a sua última atualização, como essa informação vai ser guardada e etc. Esse cache pode ser mantido por navegadores, proxys entre outros.

6) Cookie é uma forma de identificação que contém informações acerca da sessão do usuário em determinada aplicação web. Essas informações geralmente são sensíveis e se referem dados de contas. Uma forma de ataque é a captura desse cookie por outro que não o próprio usuário, já que de posse dele qualquer um pode obter acesso à sua conta ou/e informações.

7) OWASP-Top-Ten é uma lista que rankeia as 10 vulnerabilidades mais frequentes em aplicações.

8) Recon é uma etapa de Pentest que consiste em angariar informações acerca das vulnerabilidades do sistema. É importante para decidir quais ferramentas usar e monstar um plano de ataque.

9) a) Command Injection é uma forma de ataque que consiste em rodar comandos shell script em determinados campos/header de uma aplicação de modo a conseguir manipular o seu host.

b)

10) a) SQL é uma forma de ataque que consiste em rodar comandos SQL em determinados campos/header de uma aplicação de modo a conseguir informações em seu banco de dados.

b) É um ataque de SQL Injection que faz uso do comando UNION do SQL para conseguir anexar outro comando SELECT à query original.

c) Blind SQL Injection é quando a aplicação web que está sendo atacada não transmite nenhuma resposta visual ao ataque, por exemplo, mostrar um log de erro para a query testada. Então o invasor deve ficar testando queries obtenham algum retorno visual do site que mostre que este é de fato vulnerável a SQL Injection.

d)

11) a)XSS é um ataque que consiste em acrescentar um javascript à alguma aplicação web pelo lado do cliente.

b) Reflected: a inserção do código é feita em um campo qualquer, geralmente via URL, e não é acrescentada ao código original da aplicação de forma que sua disseminação para outros usuários só é possível compartilhando a URL alterada; Stored: nesse tipo de XSS, o código é inserido na aplicação permanentemente; DOM: esse ataque não depende de comunicação com o server diferente dos outros.

c)

d)

12) a) LFI explora uma vulnerabilidade na hora de indexar arquivos há uma aplicação web. Esta vulnerabilidade é a exposição do path do arquivo na URL de forma que ao alterar esse path o atacante consegue visualizar na própria página outros arquivos do server.

b) RFI é parecido com o anterior porém o path alterado, aponta para uma outra URL, podendo redirecionar usuários para um site de escolha do atacante.

c) Path Traversal, consiste em alterar o path do do arquivo original para visualizar outros diretórios dentro do server.

d) Aliando os dois, o atacante pode usar o Path Traversal para navegar pelos diretórios do host e o LFI para visualizar os arquivos dentro destes. Podendo acessar arquivos sensíveis do sistema, como arquivos de senhas por exemplo.

e)

13) a) CSRF é uma forma de ataque que se baseia na confiança de sites frequentados por um usuário em sua autenticação persistida em dado navegador. O atacante se aproveita de o usuário já estar autenticado em determinado site (do banco por exemplo) e envia de forma oculta deste usuário por meio de um outro site comum aos dois (como uma rede social) uma requisição que é disparada do lado do próprio usuário, essa requisição sendo de transferências bancárias, por exemplo, acaba sendo validada pelo banco por vir do lado do usuário (que já estava previamente logado).

b)

c) Ataque em que o ator faz com que uma determinada aplicação faça requisição para o próprio serve acessando conteúdos do mesmo como se fosse localhost.

d)

e) Principalmente evitando habilitar a opção lembrar-me nos sites.