



# SEGURANÇA CIBERNÉTICA

## AMEAÇAS, VULNERABILIDADES E ATAQUES II: **VULNERABILIDADES WEB E ENGENHARIA SOCIAL**

Prof. Dr. Daniel Caetano

2022 - 1

# Compreendendo o problema

- **Situação:** A internet surgiu para integrar a humanidade. Como a informação tem alto valor, pessoas mal intencionadas buscam meios de obtê-las a qualquer custo.



**Como tais dados podem ser obtidos?**

# Compreendendo o problema

- **Situação:** a interação com sites é um dos pontos centrais da atuação de crackers para obter informações sobre as pessoas. Existem vulnerabilidades intrínsecas



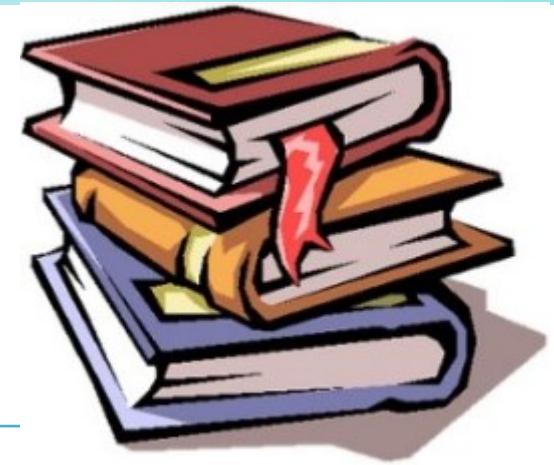
## Quais as origens dessas vulnerabilidades?

# Objetivos

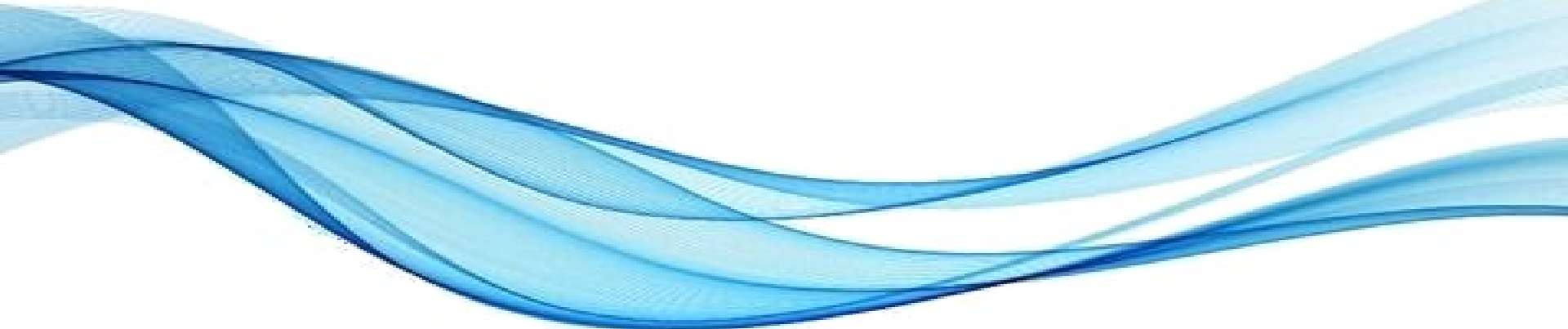
- Conhecer o funcionamento de aplicações web e algumas de suas vulnerabilidades
- Entender os principais tipos de códigos maliciosos
- Entender o processo de engenharia social e seus principais alvos na empresa



# Material de Estudo



Material	Acesso ao Material
Notas de Aula e Apresentação	<a href="https://www.caetano.eng.br/aulas/2022a/ara0076.php">https://www.caetano.eng.br/aulas/2022a/ara0076.php</a> (Segurança Cibernética – Aula 05)
Minha Biblioteca	<ul style="list-style-type: none"><li>• Hackers Expostos: Segredos e Soluções para a Segurança de Redes (ISBN: 978-0-07-178028-5), págs 555 a 568.</li><li>• Segurança de Computadores: Princípios e Práticas (ISBN: 978-85-352-6449-4), págs 501 a 507.</li></ul>
Material Adicional	<ol style="list-style-type: none"><li>1) Vulnerabilidades em aplicações web - Disponível em: <a href="https://youtu.be/oaxYwTk3AoE">https://youtu.be/oaxYwTk3AoE</a></li><li>2) Reconhecimento Web: Introdução ao PenTesting - Disponível em: <a href="https://youtu.be/wdysXk6Jsbk">https://youtu.be/wdysXk6Jsbk</a></li><li>3) Ferramentas automatizadas para identificação de vulnerabilidades web – Disponível em <a href="https://youtu.be/ElaNa_Rwck8">https://youtu.be/ElaNa_Rwck8</a></li></ol>

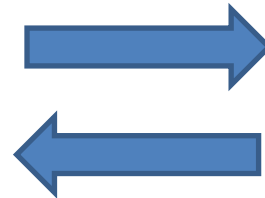


**VISÃO GERAL:**

# **FUNCIONAMENTO E DESENVOLVIMENTO WEB**

# O que é a World Wide Web?

- Chrome, Firefox, Edge, Safari, Opera...
  - São Navegadores
- Eles acessam conteúdo de um servidor
  - Apache, Nginx, IIS...





# Web Server x Navegador

- Comunicação por meio do HTTP/HTTPS
- HTTP significa: HyperText Transfer Protocol
  - Especificado por Tim Berners-Lee em 1990
  - Transmitir documentos hipertexto

`http://www.meuservidor.com/index.html`



Me dá a página  
index.html !

Requisição

Resposta

404 – A página  
não existe!



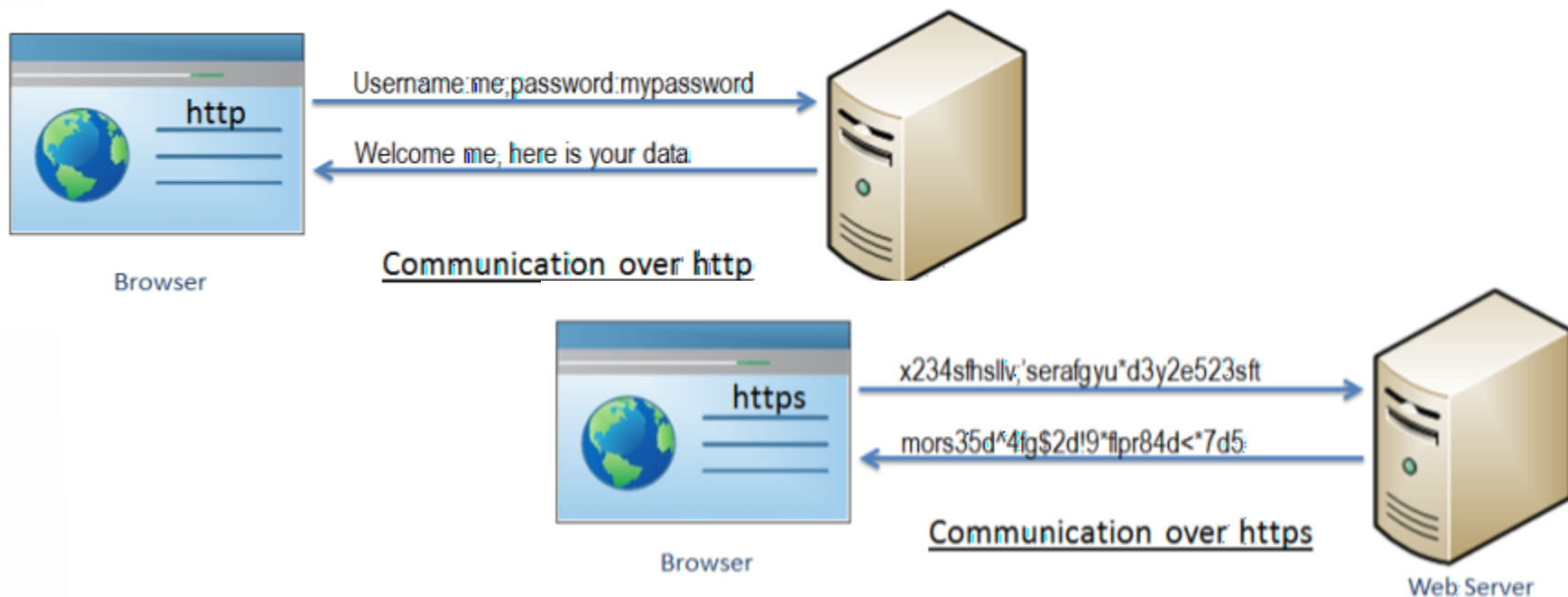
**GET**  
**POST**  
**HEAD**

...



# Navegação Segura

- Comunicação por meio do HTTPS
- HTTP + TLS (Transport Layer Security),
  - Criptografa as informações ponto-a-ponto



# Tecnologias para a Web

- Página/Aplicação Web
  - Conteúdo
  - Forma
  - Ações (cliente)
  - Ações (servidor)
- Cada parte...
  - Desenvolvida com tecnologias próprias
- Vulnerabilidades
  - Na interação entre esses elementos!



# Tecnologias Usuais

- Sopa de letrinhas
    - HTML x XHTML x HTML5
    - CSS1, 2, 3...
    - JS 2.x, 3.0...
    - DOM 1, 2...
    - AJAX
    - JSON/XML
    - Java, C#, PHP, NodeJS...
    - MariaDB, Postgre, Oracle...
- Conteúdo (estrutura)**
- Visual (cores e layout)**
- Processamento (ações)**
- Estrutura Interna (memória)**
- Transferência de Dados**
- Representação de Dados**
- Processamento no Servidor**
- Armazenamento de Dados**

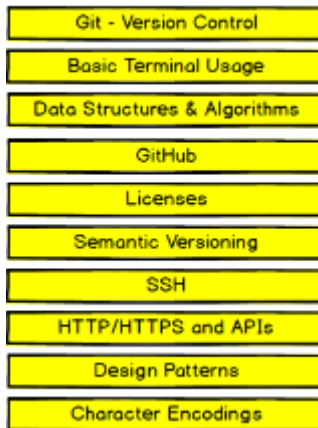
# Por que separar as ações?

- Cliente: Navegador, “Front-end”
  - Processamento da interface
- Servidor: “Back-end”
  - Armazena dados e estados
  - Gerencia processos mais sensíveis

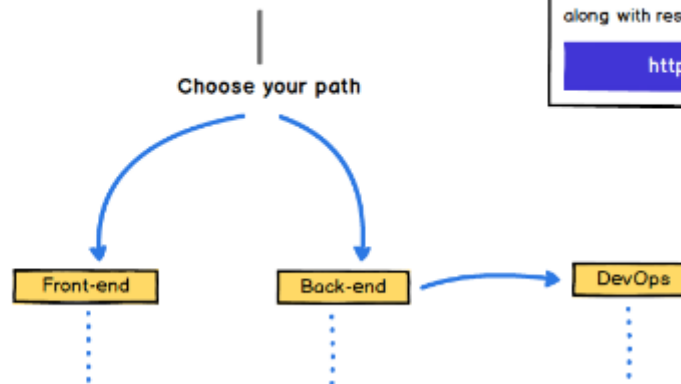
É claro quem  
faz o quê...?

Falhas!

Required for any path



## Web Developer in 2021



Find the detailed version of this roadmap  
along with resources and other roadmaps

<http://roadmap.sh>

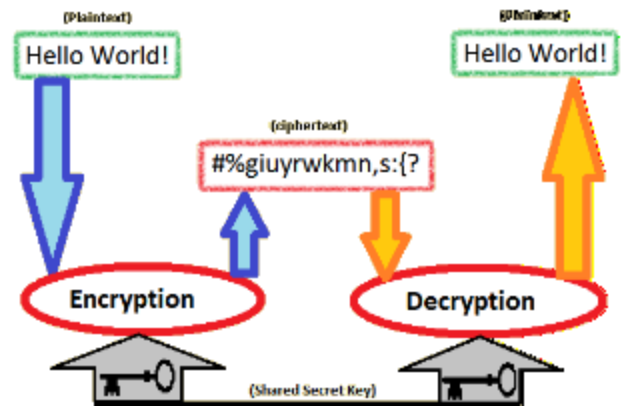


# QUESTÕES DE SEGURANÇA NA WEB



# Criptografia Salva?

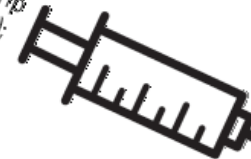
- SSL não é tudo?
  - SSL (Secure Socket Layer) X TLS (Transport Layer Security)
    - TLS v1.0, 1.1, 1.2, 1.3..
  - As coisas mudam com o tempo
- OWASP
  - Open Web Application Security Project
  - <https://owasp.org/>



# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
    - Executar código estranho à aplicação
    - <https://www.hacksplaining.com/exercises/sql-injection/>

`http://www.example.com/index.php?  
user_name=admin;phpinfo();`



### Login

  
  
  
[Lost your Password?](#)  
Don't have an account? [Sign up here!](#)



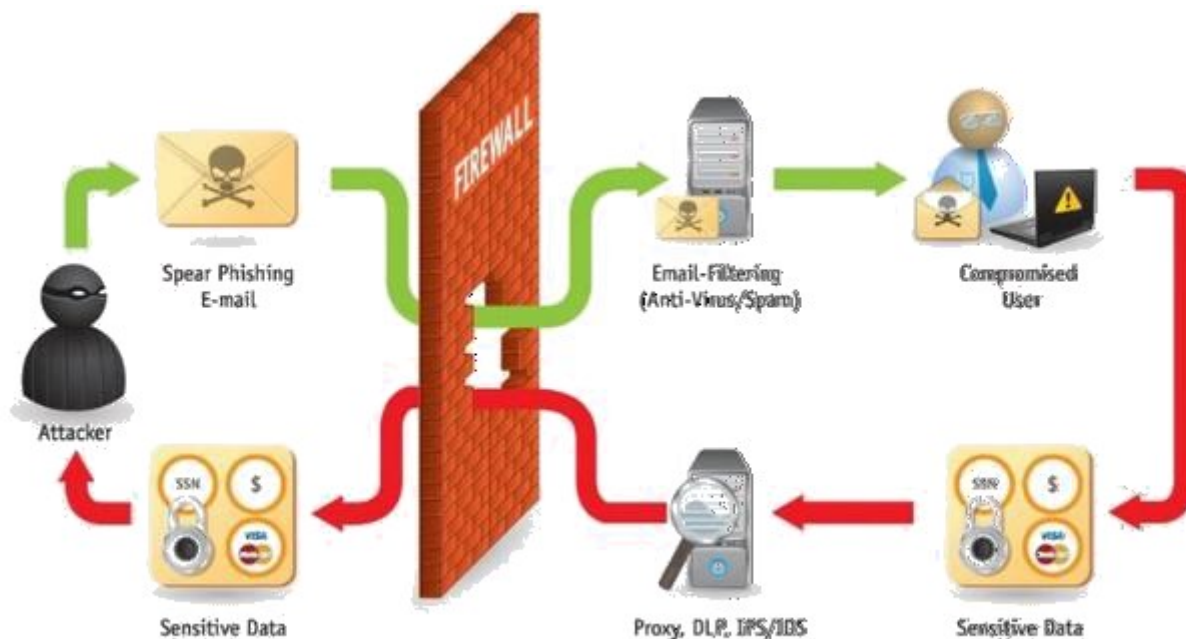
# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
    - Erros na implementação de autenticação e sessão
    - Permitem alguém a roubar senhas, sessões etc...



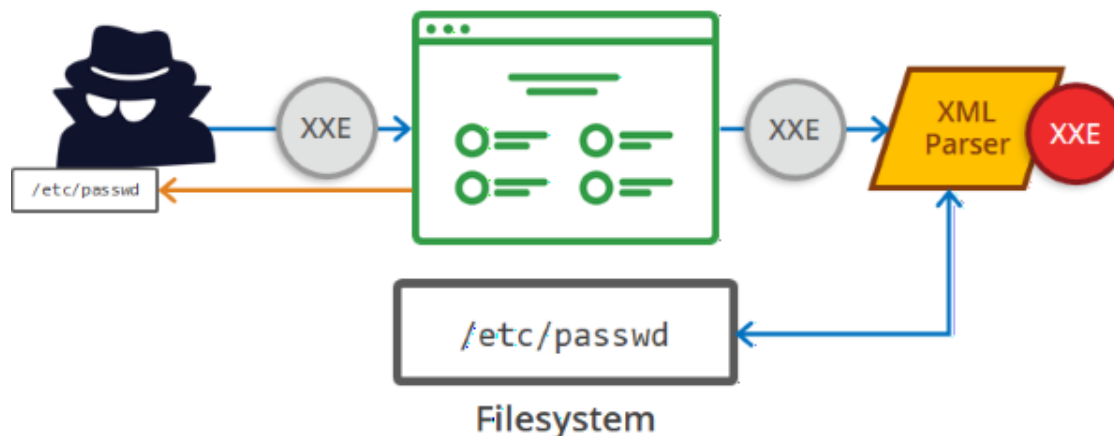
# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
    - Não mostrar o número do cartão, por exemplo!
    - Não usar dados desnecessários no front-end



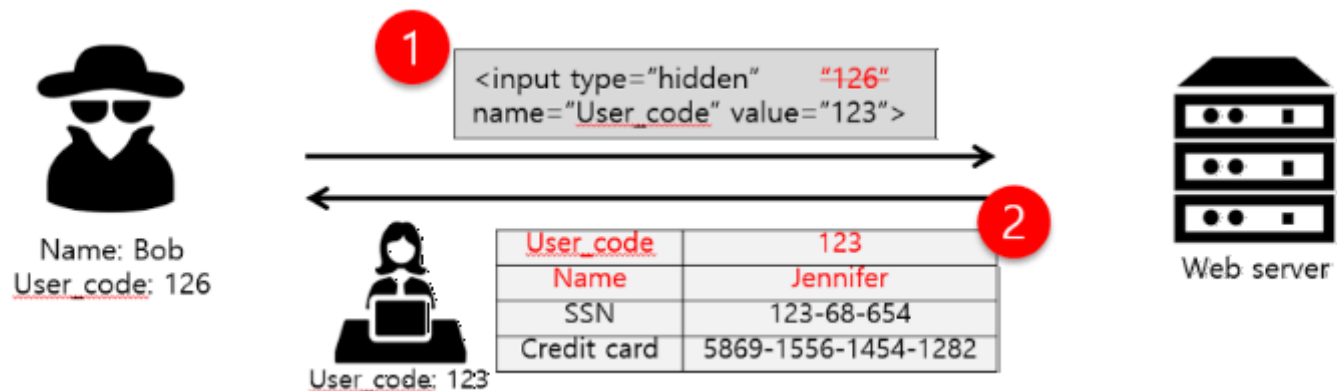
# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML (XXE)
    - Processadores de XML mal configurados avaliam dados externos
    - Abrem espaço para todo tipo de exploit!



# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML
  5. Quebra de controle de acesso
    - Falhas no controle de funcionalidades e dados
    - Some com opção do menu, mas se conhecer a URL...



# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML
  5. Quebra de controle de acesso
  6. Configuração incorreta de segurança
    - Desde identificação da versão do webserver...
    - Até acesso aberto à armazenamento em nuvem



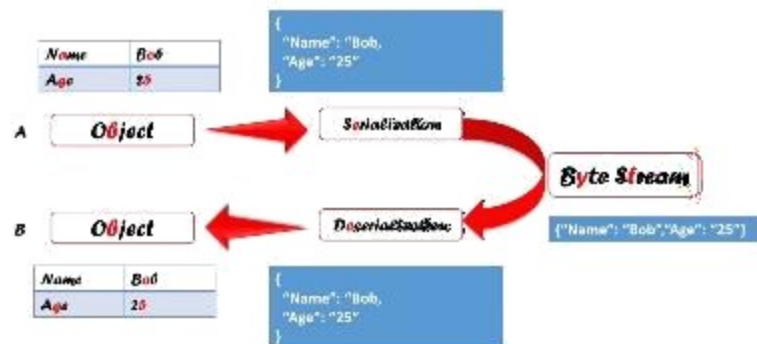
# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML
  5. Quebra de controle de acesso
  6. Configuração incorreta de segurança
  7. Cross-Site Scripting (XSS)
    - Incluir seu código na aplicação de terceiros



# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML
  5. Quebra de controle de acesso
  6. Configuração incorreta de segurança
  7. Cross-Site Scripting (XSS)
  8. Desserialização Insegura
    - Passagem de código serializado para microsserviços





# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML
  5. Quebra de controle de acesso
  6. Configuração incorreta de segurança
  7. Cross-Site Scripting (XSS)
  8. Desserialização Insegura
  9. Utilização de Componentes Vulneráveis
    - Sabe aquele plugin do WordPress que não foi atualizado?
    - CVE – Common Vulnerabilities & Exposures.



# Maiores Riscos a Aplicações Web

- Segundo o OWASP ( <https://owasp.org/www-project-top-ten/> )
  1. Injeção de Código
  2. Quebra de Autenticação
  3. Exposição de dados sensíveis
  4. Entidades externas de XML
  5. Quebra de controle de acesso
  6. Configuração incorreta de seguran
  7. Cross-Site Scripting (XSS)
  8. Desserialização Insegura
  9. Utilização de Componentes Vulneráveis
  10. Logs e monitoramento insuficientes
    - Inclui todos os dados para desvendar um incidente?

Voltaremos a  
vários desses  
tópicos em aulas  
futuras



# Exemplo de “Evolução”: XSS

- Há 10 anos, XSS estava na posição nº 2
  - Hoje, é o 7º
  - O que aconteceu?
- Havia dificuldade em corrigir
  - Muitos sites dependiam para funcionar
    - Site em um servidor, scripts compartilhados em outro
  - Bas OS



# Exemplo de “Evolução”: XSS

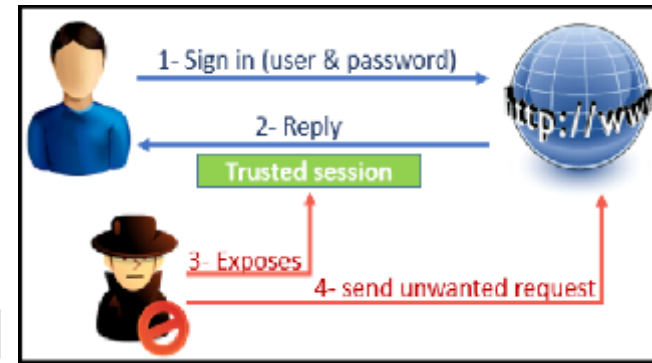
- Há 10 anos, XSS estava na posição nº 2
  - Hoje, é o 7º
  - O que aconteceu?



- Navegadores/servidores forçaram segurança
  - Por isso a queda!
  - Ao desabilitar no navegador/servidor...
    - Muitas aplicações precisaram ser redesenhadas
  - Soluções novas: cookies do tipo *httponly*
    - Só são acessíveis por scripts *server-side*

# O que há além dos “10 mais”?

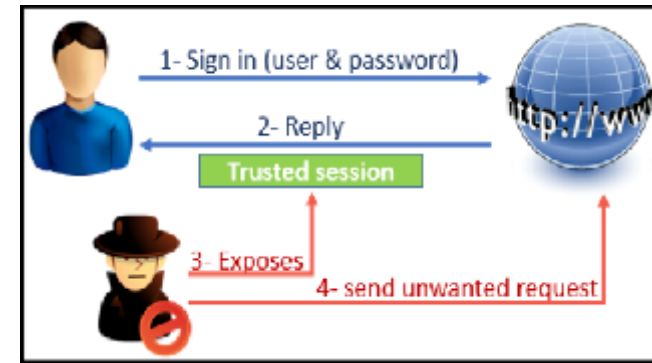
- Muita coisa está de fora!
- Cross Site Request Forgery
  - Mecanismo: Engenharia Social



- Site com áreas não autenticadas (contato, recuperar senha...)
- Copia o *form* para outro lugar, modifica e manda e-mail
  - “Por favor, atualize sua senha!”
- Usuário entra e faz operações solicitadas
- Usa campo *hidden*, envia requisição para página logada
  - » Expectativa que usuário esteja logado, por ex.!
  - » Faz downloads, baixa conteúdos etc.

# O que há além dos “10 mais”?

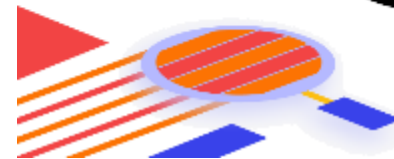
- Muita coisa está de fora!
- Cross Site Request Forgery



- Difícil identificar e poucos casos reportados
- Se aproveita de vulnerabilidade
  - Não é um “uso inteligente” de recursos
- Solução
  - Usar *hash* único para forms
  - Dificultar uso de dados de um form para outro

# Ferramentas para Prevenção

- Existem diversas ferramentas
  - Incluindo as mantidas pela OWASP
- São recursos adicionais:
  - Importante é entender os mecanismos
  - E agir preventivamente
- Luta constante e eterna
  - Client Side x Server Side
  - Em ataques, as coisas se misturam







# CONHECENDO ALGUMAS FERRAMENTAS E TÉCNICAS



# Console de Desenvolvimento

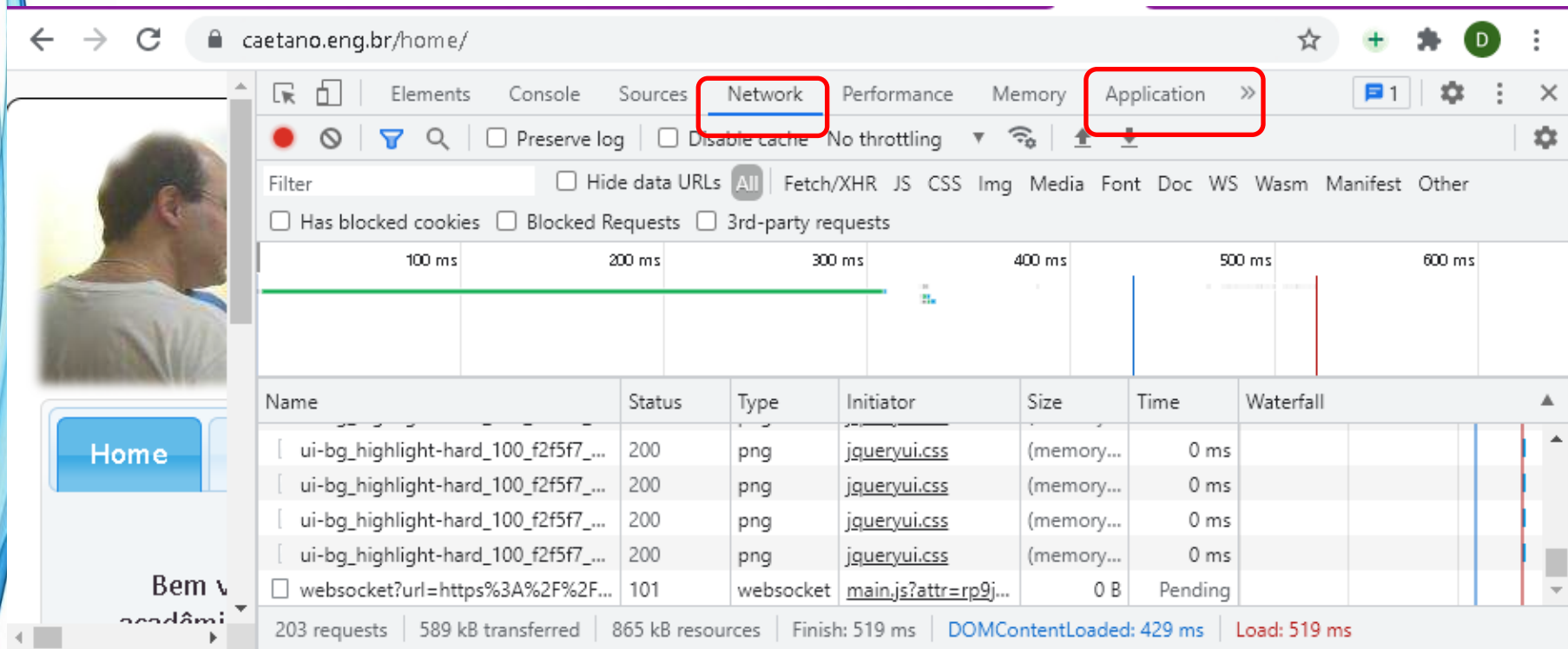
- Navegador possui ferramentas poderosas
  - Exemplo: Firefox Tecla F12

The screenshot shows a web browser window with the URL <https://www.caetano.eng.br/home/>. The page content includes a photo of a man, the text "Prof. Caetano", and a date "11/09/2021, 08:45". The developer console is open, showing the "Rede" (Network) tab selected. The console displays a request for the file "ui-bg\_glass\_100\_e4f1fb\_1x400.png" with a status of "GET" and a transfer size of "158 B (raced)".

Status	Método	Domínio	Arquivo	Iniciador	Tipo	Transferido	Tamanho	0 ms	80 m
	GET	www.caetano.eng.br	ui-bg_glass_100_e4f1fb_1x400.png	img	png	158 B (raced)	158 B	0 ms	

# Console de Desenvolvimento

- Navegador possui ferramentas poderosas
  - Exemplo: Chrome Tecla F12



The screenshot shows the Chrome DevTools Network tab. The 'Network' and 'Application' tabs are highlighted with red boxes. The 'Network' tab is active, displaying a list of requests and a waterfall chart. The waterfall chart shows a green bar for the first request, which is highlighted in blue. The list of requests is as follows:

Name	Status	Type	Initiator	Size	Time	Waterfall
[ ui-bg_highlight-hard_100_f2f5f7_...	200	png	jqueryui.css	(memory...	0 ms	
[ ui-bg_highlight-hard_100_f2f5f7_...	200	png	jqueryui.css	(memory...	0 ms	
[ ui-bg_highlight-hard_100_f2f5f7_...	200	png	jqueryui.css	(memory...	0 ms	
[ ui-bg_highlight-hard_100_f2f5f7_...	200	png	jqueryui.css	(memory...	0 ms	
[ websocket?url=https%3A%2F%2F...	101	websocket	main.js?attr=rp9j...	0 B	Pending	

Summary: 203 requests | 589 kB transferred | 865 kB resources | Finish: 519 ms | DOMContentLoaded: 429 ms | Load: 519 ms

# NetCat

- Se comunica com web server por prompt
  - Ncat é instalado no Windows com o Zenmap
  - Nc pode ser instalado no Linux separadamente



# Procurando Vulnerabilidades

- O que tem no server?
  - Já vimos o Nmap...



**Apache**



Microsoft  
IIS



©sanckit0496

# Procurando Vulnerabilidades

- Algumas técnicas/ferramentas
  - Google Hacking
  - Ferramentas de varredura
    - Dirb, DirBuster, DirStalk, Scout...
  - Ferramentas de varredura de vulnerabilidade
    - wapiti

Google Hack





# ENGENHARIA SOCIAL DENTRO DA EMPRESA





# Engenharia Social

- Pessoas são predispostas a serem úteis
  - Ou são motivadas a colaborar
- Técnicas comuns:
  - Personificação (individual / funcional)
  - Suborno
  - Fraude
  - Afinidade
  - Engenharia Social Reversa.



# Engenharia Social

- Principal técnica de combate...



**Informação**

# Treinamento

- É importante?
  - Evitar “desconhecimento”
  - Melhoria no comportamento dos empregados
  - Com relação à lei, desconhecer não é desculpa
    - Responsabilizar empregados / atenuar empresa



# Qual Treinamento?

Tipo	Quem
Conscientização	Todos
Conhecimentos básicos de segurança	Todos que lidem com TI
Treinamento	Papeis e responsabilidades funcionais relativas a sistemas de TI
Educação de Segurança	Especialistas/Profissionais de Segurança em TI

	Conscientização	Treinamento	Educação
<b>Atributo</b>	"O quê"	"Como"	"Por quê"
<b>Nível</b>	Informação	Conhecimento	Percepção
<b>Objetivo</b>	Reconhecimento	Habilidade	Entendimento
<b>Método de ensino</b>	<b>Mídia</b> – Vídeos – Boletins informativos – Pôsteres etc.	<b>Instrução prática</b> – Palestra – Seminário de estudo de caso – Prática	<b>Instrução teórica</b> – Seminário de discussão – Leitura sobre o assunto
<b>Tipo de teste</b>	Verdadeiro/falso Múltipla escolha (identifica aprendizado)	Solução de problemas (aplica aprendizado)	Ensaio (interpreta aprendizado)
<b>Impacto</b>	Curto prazo	Prazo intermediário	Longo prazo

# Conscientização e Básico de Seg.

- Alguns elementos que não podem faltar:
  - Regras para os Recursos Disponibilizados
    - O que pode ou não ser feito com os mesmos!
    - Atividades ilegais são de responsabilidade dos autores!.
  - Sistemas de Monitoramento
    - Funcionários precisam estar cientes!
    - E-mail, sites visitados, ligações feitas e recebidas....





# Conscientização e Básico de Seg.

- Alguns elementos que não podem faltar:
  - Inspeção de Conteúdos
    - Empresa pode inspecionar qualquer dado
  - Instalação de Software
    - Que não podem instalar sem autorização formal e expressa da empresa



# Conscientização e Básico de Seg.

- Alguns elementos que não podem faltar:
  - Regras de Firewall/Proxy
    - Quais são as regras e que elas não podem ser alteradas
  - Divulgação de informações
    - Quais são as regras, classificações, penalidades...
    - Cuidado com todos, mas especial com desconhecidos!





# Conscientização e Básico de Seg.

- Alguns elementos que não podem faltar:
  - Uso recreativo da Internet
    - Se for possível, apenas no almoço ou fora de expediente
    - Apenas para atividades legais.
  - Preservação dos dados de acesso
    - Cuidados com as senhas...





# **ATIVIDADE**

# Atividade

- Em grupo!
- Procure sites com falhas e tente descobrir o que você conseguir sobre eles.
- Escolha um deles e, com as informações encontradas, imagine algum tipo de situação em que seria possível usá-las para obter mais informações por meio de Eng. Social.



# ENCERRAMENTO

# Resumo e Próximos Passos

- Principais tipos de vulnerabilidades web
  - Uso do navegador para analisar o lado cliente
  - Uso de ferramentas
    - Identificar vulnerabilidades!
  - Noções de treinamento
  - **Pós Aula:** Saiba Mais, A Seguir... e Desafio!
    - No mural: <https://padlet.com/djcaetano/segciber>
- 
- Segurança em Wireless
    - Como funciona? Como nos proteger?



# PERGUNTAS?