# Segurança Cibernética

## Vulnerabilidades Comuns:
# Principais Vulnerabilidades da Web II
## (Conteúdo Digital Aura!)

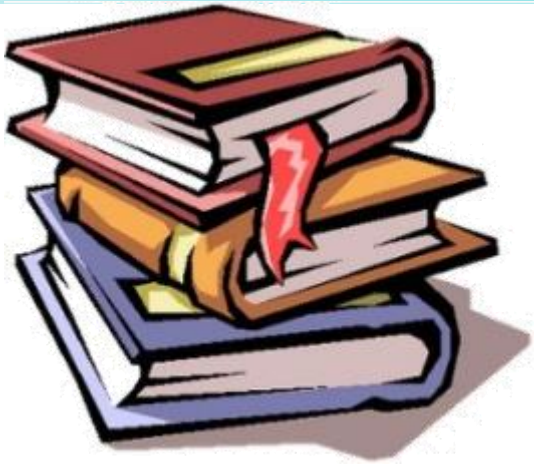
Prof. Dr. Daniel Caetano

2021 - 2

# Objetivos

- Compreender a lógica do sequestro de sessão em aplicações web
- Compreender o que é quebra do controle de acesso em aplicações web
- Compreender as falhas de cross-site scripting (XSS)
- Compreender o problema da desserialização insegura
- Compreender o problema do monitoramento insuficiente

# Material de Estudo

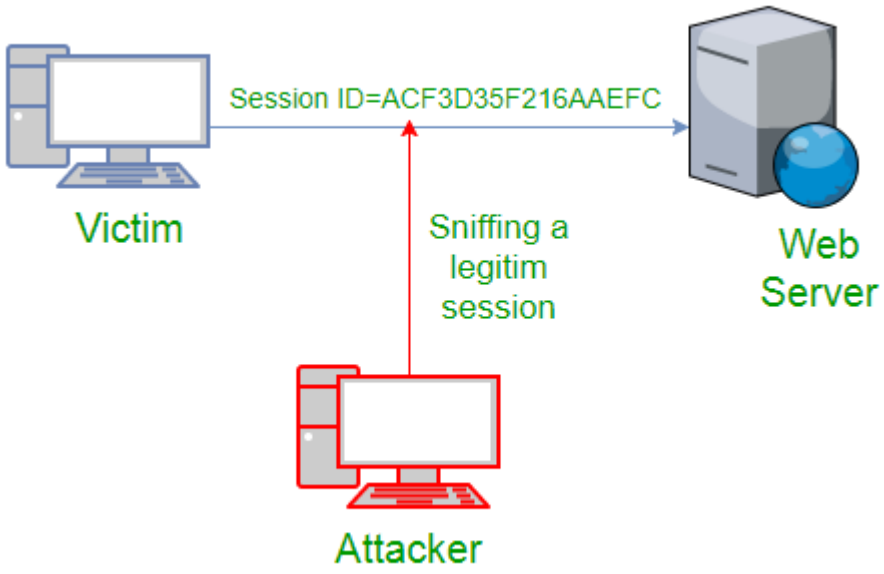


| Material | Acesso ao Material |
|---|---|
| Notas de Aula e Apresentação | https://www.caetano.eng.br/aulas/2021b/ara0076.php (Segurança Cibernética – Aula 8) |
| Módulo Digital | • Ambiente Aura: Tema 3, Assunto 2 |
| Material Adicional | 1) Cross-site scripting. Dispoinível em: https://youtu.be/brB6xFzCmCw<br>2) Desserialização insegura. Dispoinível em: https://youtu.be/-BfizfhKN3A |

# Sequestro e Quebra de Sessão

# Sequestro e Quebra de Sessão

- O que é?
  - Alguém “roubar” a sessão de outro usuário
  - Alguém forjar sessões válidas.
- Como funciona?



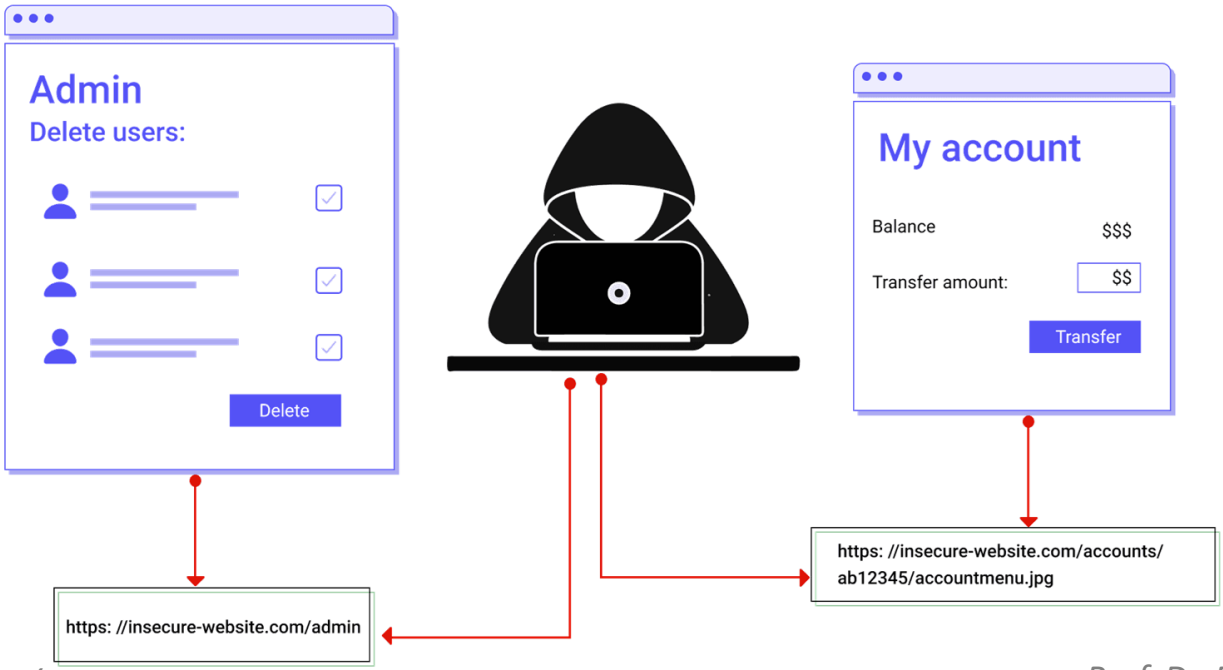
# Sequestro e Quebra de Sessão

- Como evitar?
  - Tratar adequadamente a sessão
    - Codificar dados.
  - Manter a sessão em banco de dados
    - Sessão se torna um ID, dados estão no banco
    - Associar ID da sessão ao IP do computador
    - Associar um *timeout* à sessão.
  - Colocar o identificador da sessão como httponly
    - Já veremos mais sobre isso adiante.
  - Usar linguagem que controle a sessão.

# Quebra de Controle de Acesso

# Quebra de Controle de Acesso

- O que é?
  - Usuário acessar página que não deveria
  - Em geral: falta de verificação de permissões.
- Como funciona?



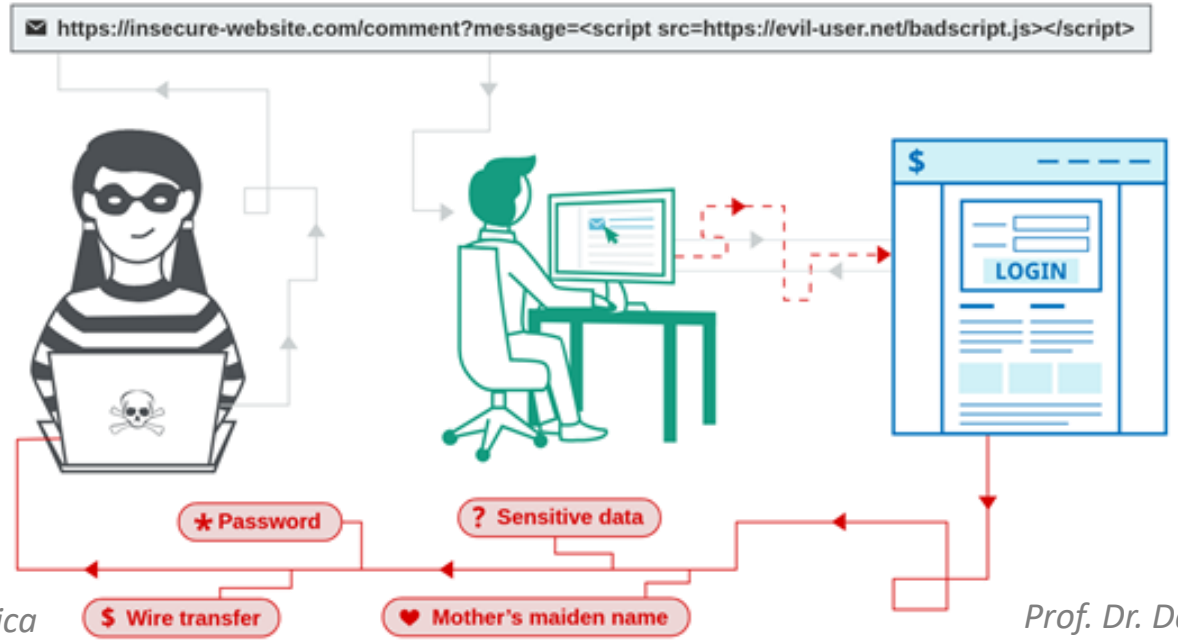
# Quebra de Controle de Acesso

- Como evitar?
  - Verificar permissões em toda página “protegida”
    - Não se limitar ao menu!
  - Verificar permissões na execução de ações
    - Front e Backend
  - Manter permissões em BD, se possível
    - Facilitar a manutenção e revogação.

# Cross-Site Scripting (XSS)

# Cross-Site Scripting (XSS)

- O que é?
  - Um tipo de injeção de código...
  - Quando o código é executado do lado do cliente
    - Em geral... JavaScript.
- Como funciona?



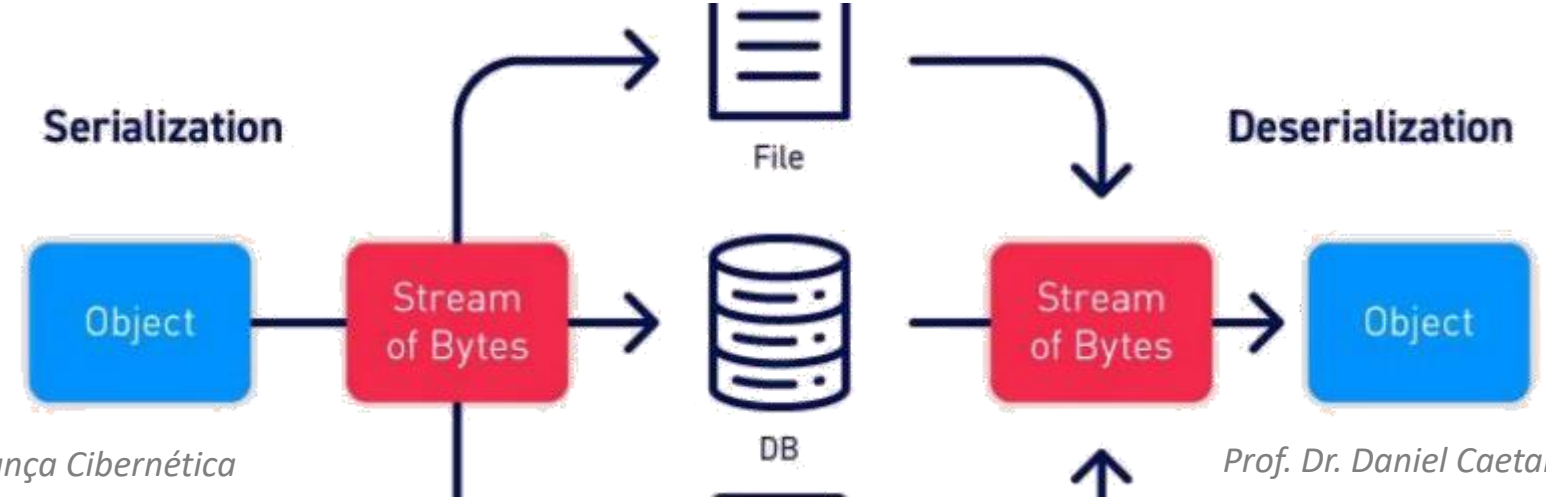
# Cross-Site Scripting (XSS)

- Como evitar?
  - Configurar web server para não aceitar
    - No httpd.conf
    - Header always set X-XSS-Protection "1;  mode=block"
  - Configurar navegador para não aceitar
    - No cabeçalho
    - Content-Security-Policy: script-src 'self’
  - Trate corretamente as entradas de usuário
  - Nunca apresentar/usar diretamente dados digitados pelo usuário
    - Ser o mais restrito possível

# DESSERIALIZAÇÃO INSEGURA

# Desserialização Insegura

- O que é?
  - Serializar: objeto (dados+código) → String
  - Desserializar: String → objeto (dados+código)
  - Objetivo: armazenar ou transferir objeto pela rede
  - Ataque: alterar o “texto” transmitido...
    - Enviando um código malicioso no lugar.



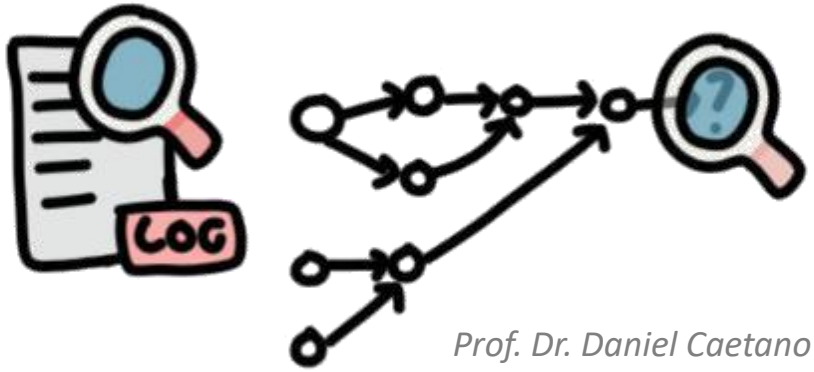
# Desserialização Insegura

- Como evitar?
  - Assinar digitalmente os dados
    - Rejeitando dados cuja assinatura não seja válida
  - Quando não for possível assinar?
    - Servidor intermediário para checar os dados
  - Monitorar processos de desserialização
    - Alerta caso algum usuário ocasione muitas.

# Monitoramento Insuficiente

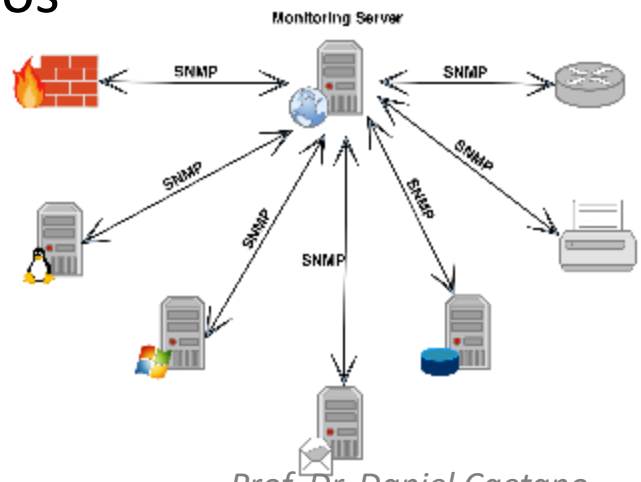
# Monitoramento Insuficiente

- O que é?
  - Registro incompleto de operações...
  - Ou acompanhamento insuficiente dos registros...
  - Ou relógios dessincronizados (ambiente de rede)
- Consequências?
  - Impossibilidade de rastrear ocorrências...
    - E de apurar os responsáveis



# Monitoramento Insuficiente

- Como é feito?
  - Monitoramento ativo
    - Zabbix, Pulseway, Cacti, Prometheus etc.
    - Usam protocolo próprio + SNMP
      - Simple Network Management Protocol
  - Monitoramento passivo
    - Windows: visualizador de eventos
    - Linux: vários arquivos de log
    - Exemplos!



# Monitoramento Insuficiente

- Como evitar problemas?
  - Período agendado para monitoramento passivo
    - Frequente!
  - Rastrear operações normais
    - Para verificar rastreabilidade
  - Proteger os arquivos de log
    - Evitar que sejam apagados
  - Simular ataques e tentar rastreá-los
  - Usar sistemas de monitoramento ativo

# ENCERRAMENTO

# Resumo e Próximos Passos

- Vários ataques via Web
  - Sequestro e quebra de sessão
  - Quebra de controle de acesso
  - Cross-Site Scripting (XSS)
  - Desserialização Insergura
  - Monitoramento Insuficiente
- **Pós Aula**: Aprenda Mais, Pós Aula e Desafio!
  - No padlet: https://padlet.com/djcaetano/segciber

---

- Contramedidas...
  - Como nos proteger?

# PERGUNTAS?