RELATÓRIO 5 – CLIENT SIDE EXPLOITATION

Lucas Loscheider Reis Muniz – lucaslc01@hotmail.com

TERMOS IMPORTANTES

Cookies: Segundo o site techmundo “cookies são pequenos arquivos criados por sites visitados e que são salvos no computador do usuário, por meio do navegador. Esses arquivos contêm informações que servem para identificar o visitante.”. Com isso, itens adicionados ao carrinho de compras se mantém mesmo que a pesosa saia da página; anúncios personalizados são exibidos; termos de busca em formulários, senhas e logins são salvos para não precisar ficar logando, ou digitando toda vez que tentar entrar no site.

CSRF

Cross-Site Request Forgery (CSRF) – em português, falsificação de solicitação entre sites. Segundo o site OWASP.org, CSRF é um tipo de ataque que força o usuário final a executar ações não desejadas em uma aplicação web a qual, o usuário final, está naquele instante autenticado. Ou seja, é um ataque que herda a identidade e os privilégios da vítima para realizar ações indesejadas pela vítima, como mudar email/senha ou realizar uma compra.

Os pedidos do navegador web incluem automaticamente qualquer credencial associada com o site, como **cookies** de seção, endereço IP e por aí vai. Portanto, se o usuário está autenticado em um site, o site não sabe distinguir entre a solicitação forjada mandada pela vítima ou uma solicitação de fato verdadeira mandada pela vítima.

Exemplo de como acontece:

1. A vítima loga no seu email, usando um navegador. A credencial do usuário é validada, um cookie é enviado na resposta da requisição HTTP. A partir desse momento, o navegador tem salvo no disco o cookie que o mantém autenticado.
2. Em outra página web, por meio de engenharia social, a vítima é convencida a clickar em um link para um site arbitrário.
3. Ao entrar nesse site, no corpo dele, tem um formulário em javascript que emula exatamente como o seu email faz para mudar o login, por exemplo.
4. Em algum momento, esse site arbitrário submete um formulário (em JavaScript) solicitando a mudança de login. Foi forjada uma requisição Cross-Site, de um site para outro (daí o nome Cross-Site Request Forgery).

Como prevenir:

CSRF tokens: é criado um token único por usuário. Esse token fica salvo na sessão do usuário no servidor e, quando o formulário é postado, o token enviado pelo formulário é comparado com o que se tem na sessão, lá no servidor. Sendo iguais, a requisição é aceita. Caso contrário, é recusada. O token precisa ser único por seção, secreto, imprevisível, associado à sessão do usuário logado e sempre precisa ser solicitado para qualquer requisição deste usuário.

XSS

Cross-Site Scrypting (XSS) – Segundo o site OWASP.org, é um tipo de ataque o qual o atacante usa uma aplicação web para mandar código malicioso, geralmente na forma de script de navegador, para um usuário final. O navegador do usuário final não sabe que o script é malicioso e o executa. Por isso o script malicioso consegue acessar cookies e outras informações registradas pelo navegador.

Existem 2 tipos de ataque XSS:

- **Persistent:** O script injetado pelo atacante fica alojado de forma permanente no servidor de destino. O usuário pode acionar a carga apenas por navegar num website infectado. Esse é um tipo bastante perigoso de ataque, pois o código pode estar alojado em diversos destinos, como campos de comentário, base de dados etc.
- **Reflected ou non-persistent:** o script não estará alojado em um servidor de destino e por isso precisará ser entregue para cada vítima. Isso pode acontecer por várias formas de engenharia social, por exemplo uma mensagem de erro ou um resultado de busca. Uma forma frequente será um link distribuído por meio de esquemas de phishing. Ao acionar o servidor, por meio do link, o script será refletido e executado no navegador. Esta técnica é a mais frequente.

Exemplo de como acontece:

O atacante se aproveita de uma dada vulnerabilidade do próprio site para instalar um script que irá executar ações maliciosas como copiar cookies, tokens ou roubar dados de acesso registrados no navegador web do usuário. Quando este usuário faz uma pesquisa nesse site, as informações enviadas contém malwares capazes de roubar dado**s.**

Como prevenir:

- Fique sempre atento ao clicar em links recebidos, mesmo que o remetente seja algum conhecido próximo, antes de clicar verifique se o link não possui conteúdo malicioso, como por exemplo ter os símbolos < e > ao longo do endereço.
- Acesse sempre sites confiáveis e que conheça a reputação. Sites amplamente conhecidos tendem a ter uma equipe responsável por cuidar da segurança do site e de seus usuários.
- Para os desenvolvedores da página web: Nunca ponha links e códigos não testados e não confiáveis no documento HTML; No ponto em que os dados controláveis pelo usuário são produzidos nas respostas HTTP, codifique a saída para evitar que seja interpretada, pelo atacante, como conteúdo ativo. Dependendo do contexto de saída, isso pode exigir a aplicação de combinações de codificação HTML, URL, JavaScript e CSS; Filtrar sempre o input que se chega ao site, baseado no que é esperado no que seria esperado.

Observações dos entregáveis do laboratório

Os vídeos contendo as soluções estão com o áudio ruim por causa do forte barulho do cooler do notebook, e o áudio está baixo porque gravei a noite e meus pais estavam dormindo. Sendo assim pros próximos laboratórios tentarei resolver esses problemas.
Também não consegui resolver o laboratório de número 3: CSRF where token is not tied to user session. Eu copio e colo as soluções no exploit server, mas por algum motivo nao acontece nada nem mensagem de erro aparece.
Segue abaixo os links dos vídeos:
Primeira : https://youtu.be/9Eru3HWvaY4
segunda: https://youtu.be/UAcEHFz26BY
quarta: https://youtu.be/7ALHFBG5d6w