FEPEX
Feira de Ensino, Pesquisa e Extensão
Campus São Francisco do Sul

ADOÇÃO DE BOAS PRÁTICAS DE SEGURANÇA EM UM JOGO DE SEGURANÇA COMPUTACIONAL

Autores: Vítor Augusto Ueno Otto, Lucas Vargas, Ricardo de la Rocha Ladeira
Identificação autores: bolsista interno; bolsista interno; orientador IFC-*Campus* Blumenau
Avaliação na modalidade: Pesquisa – Ciências Exatas e da Terra Nível: Superior
Palavras-chave: segurança computacional; boas práticas de segurança; jogo de segurança computacional; TreasureHunt

Introdução

Em um mundo em que tecnologia e informação se fazem cada vez mais presentes, temas como Segurança Computacional vêm sempre à tona. No âmbito do ensino dessa área, o uso de jogos e competições se mostra instigador aos alunos, que se beneficiam da prática (Schreuders et al., 2017). Nesse aspecto, é importante que ferramentas como o TreasureHunt (Ladeira, 2018), um gerador de competições e desafios de forma automatizada para o ensino da Segurança Computacional, se mantenham atualizadas e atentas à segurança dos próprios dados, pois os casos de tentativa de invasão e sabotagem desses sistemas em competições não são incomuns (Matias, et al. 2018).

Assim sendo, o presente trabalho tem como objetivo detalhar as alterações realizadas no *back-end* e nos scripts geradores da plataforma TreasureHunt, evidenciando as boas práticas adotadas.

Material e Métodos

Para realizar as alterações do *back-end* e dos scripts do gerador de competições da ferramenta TreasureHunt, utilizou-se uma lista de atividades classificadas por prioridade definida pelos autores. As mudanças do projeto foram realizadas incrementalmente adicionando-as a um repositório *online*[1]. A seleção das atividades baseou-se nas necessidades registradas em anos anteriores do projeto e na leitura de trabalhos relacionados. Tais atividades foram classificadas nos seguintes grupos: adição de recursos, refatoração de código e melhorias de segurança. Para aferir os resultados obtidos pelas alterações do projeto, foi usado o validador de websites *Google Lighthouse*[2].

Resultados e discussão

A lista de alterações recentes do TreasureHunt está descrita na Tabela 1, que também inclui algumas das próximas tarefas, como as melhorias no gerador de desafios.

As mudanças no instalador visam a facilitar a instalação tanto dos requisitos necessários ao gerador de competições pelo organizador, quanto, opcionalmente, às ferramentas recomendadas para a resolução dos desafios pelos jogadores.

No gerador de competições, as principais mudanças foram otimizações dos scripts, o detalhamento dos *logs*, a detecção de erros verificando o número de respostas geradas e a adição da possibilidade de salvar os arquivos de competições anteriores, caso existam. Também foi adicionada a opção de ativação do ngrok[3] ao fim do script: um túnel de rede seguro com *https (Hyper Text Transfer Protocol Secure*, protocolo de transferência de hipertexto seguro*)*, de forma que o organizador não precise necessariamente configurar sua rede ou expor seu endereço IP para realizar uma competição. O *https* é preferível em relação ao *http*, pois proporciona confiabilidade e integridade aos dados ao incluir uma camada *SSL*/*TLS* (Felt, 2017).

No *back-end* substituiu-se o método de criptografia *hash* de dados *sha-256* pelo *bcrypt*. Essa alteração visa a impedir ataques do tipo força bruta, pois o *bcrypt* inclui *salts* aleatórios e um parâmetro de custo que pode ser aumentado para acompanhar a evolução de

1- https://github.com/TreasureHuntGame/TreasureHunt
2- https://developers.google.com/web/tools/lighthouse
3 - https://ngrok.com/

hardware (Malvoni, 2014). Como o TreasureHunt não armazena dados que identificam os usuários, os *hashes* são utilizados apenas para armazenar as senhas e respostas dos desafios.

Os resultados da validação do *Google Lighthouse* indicam que a plataforma web apresenta 80% de conformidade com as boas práticas, o que inclui o uso de *https*. Apesar disso, observaram-se erros que precisam de atenção, como a recomendação de atualização da biblioteca *front-end Jquery*, cuja versão utilizada apresenta vulnerabilidades conhecidas. Essas recomendações foram adicionadas à lista de atividades e têm prioridade.

Tabela 1 - Funcionalidades e tarefas do TreasureHunt

Local da alteração	Descrição	Status
script instalador	adição: suporte a principais gerenciadores de pacotes linux	cumprido
script instalador	adição: ferramentas mínimas para jogadores e guias de instalação	cumprido
gerador de competições	adição: detalhamento dos *logs*	cumprido
gerador de competições	refatoração: reestruturado o script em funções	cumprido
gerador de competições	adição: gerenciamento de arquivos de competições anteriores	cumprido
gerador de competições	adição: conferência de quantidade de respostas com n° de jogadores	cumprido
gerador de competições	refatoração: otimização de scripts de solução e correção de bugs de composição	cumprido
gerador de competições	melhoria de segurança: opção de abertura do *ngrok* ao fim do *script*	cumprido
back-end (aplicação web)	melhoria de segurança: substituição de sha-256 por *bcrypt*	cumprido
gerador de desafio	adição: incluir acessibilidade aos desafios da classe "*web*"	andamento
front-end	melhoria de segurança: atualizar bibliotecas *front-end*	agendado

Conclusão

O presente trabalho destacou as alterações recentes do *back-end* e *scripts* do TreasureHunt. As boas práticas comentadas incluem o uso do *bcrypt* como método de criptografia *hash* para armazenamento de todos os dados sensíveis de uma aplicação e o uso de *https* ao invés de *http*. Por meio dessas alterações obteve-se a pontuação de 80% em boas práticas do *Google Lighthouse.*

Como trabalhos futuros pretende-se fazer as correções sugeridas pelo *Google Lighthouse* e incluir novas ferramentas de validação automática para o ciclo de desenvolvimento da plataforma, a fim de melhorá-la continuamente.

Agradecimentos

Os autores agradecem ao IFC que tornou esse trabalho possível.

Referências

FELT, Adrienne Porter et al. Measuring {HTTPS} adoption on the web. In: **26th {USENIX} Security Symposium ({USENIX} Security 17)**. 2017. p. 1323-1338.

LADEIRA, Ricardo de la Rocha. **TreasureHunt**: geração automática de desafios aplicados no ensino de segurança computacional. 2018. 119 f. Dissertação (Mestrado) - Curso de Computação Aplicada, Udesc, Joinville, 2018.

MALVONI, Katja; KNEZOVIC, Josip. Are your passwords safe: Energy-efficient bcrypt cracking with low-cost parallel hardware. In: **{USENIX} 8th WOOT 14**. 2014.

MATIAS, Paulo et al. NIZKCTF: a noninteractive zero-knowledge capture-the-flag platform. **IEEE Security & Privacy**, v. 16, n. 6, p. 42-51, 2018.

SCHREUDERS, Z. Cliffe et al. Security Scenario Generator (SecGen): A Framework for Generating Randomly Vulnerable Rich-scenario VMs for Learning Computer Security and Hosting {CTF} Events. In: **2017 {USENIX} ASE' 17**. 2017.