INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CÂMPUS VOTUPORANGA

MARIANA DE ALMEIDA RODRIGUES

BOAS PRÁTICAS DE PROGRAMAÇÃO NO DESENVOLVIMENTO DE APLICAÇÕES WEB

VOTUPORANGA
2020

Mariana de Almeida Rodrigues

BOAS PRÁTICAS DE PROGRAMAÇÃO NO DESENVOLVIMENTO DE APLICAÇÕES WEB

Trabalho de Conclusão de Curso apresentado como exigência parcial para obtenção do diploma do curso Tecnólogo em Informática no Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Câmpus Votuporanga.

Professor Orientador: Ubiratan Zakaib do Nascimento.

Votuporanga

2020

FICHA CATALOGRÁFICA

(APÓS AS CORREÇÕES DA BANCA EXAMINADORA A FICHA CATALOGRÁFICA DEVERÁ SER SOLICITADA, VIA PERGAMUM, À BIBLIOTECA PARA ELABORAÇÃO)

ERRATA

(Exemplo)

SOBRENOME, Nome. **Título do trabalho:** subtítulo do trabalho, 2017. 57 f. Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) — Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Votuporanga, 2017.

Folha	Linha	Onde se lê	Leia-se
13	2	quantificação	quantificação
20	10	computadorw	computador

Elemento Pré-Textual Opcional. Deve ser inserida logo após a folha de rosto, constituída pela referência do trabalho e pelo texto da errata. Apresentada em papel avulso ou encartado, acrescida ao trabalho depois de impresso, caso não haja outro jeito de corrigir o erro de digitação.

Mariana de Almeida Rodrigues

BOAS PRÁTICAS DE PROGRAMAÇÃO NO DESENVOLVIMENTO DE APLICAÇÕES WEB

Trabalho de Conclusão de Curso apresentado como exigência parcial para obtenção do diploma do curso Tecnólogo em Informática no Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Câmpus Votuporanga.

Professor Orientador: Ubiratan Zakaib do Nascimento.

Aprovado pela banca examinadora em xx de mês de ano.

BANCA EXAMINADORA:

_____

Prof. D.r Cicrano da Silva (para feminino use Dra.)

_____

Prof. M.e Beltrano dos Santos (para feminino use M.ª)

_____

Prof. Esp. José Luis Brasil

DEDICATÓRIA

(EXEMPLO — ELEMENTO PRÉ-TEXTUAL OPCIONAL; NÃO INFORMAR O TÍTULO DEDICATÓRIA)

Dedicamos este trabalho a todos os colegas de classe que estiveram presente nessa trajetória de alegrias e de superação.

AGRADECIMENTOS

Agradeço a todos os professores e servidores do IFSP, Câmpus Votuporanga, que contribuíram direta ou indiretamente para a conclusão deste trabalho.

À minha família, que deu todo o apoio necessário para que eu chegasse até aqui.

Ao meu orientador, que me auxiliou a solucionar as dificuldades encontradas no caminho.

EPÍGRAFE

(EXEMPLO — ELEMENTO PRÉ-TEXTUAL OPCIONAL; NÃO INFORMAR O TÍTULO EPÍGRAFE)

“Descobrir consiste em olhar para o que todo mundo está vendo e pensar uma coisa diferente”

Roger Von Oech

RESUMO

O resumo é a apresentação dos pontos relevantes de um documento. O resumo de um trabalho acadêmico deve ser do tipo informativo, para que o leitor conheça as finalidades, metodologia, resultados e conclusões do documento, de tal forma que este possa dispensar a consulta ao original. É composto por uma sequência de frases concisas, afirmativas e não pode ter enumeração de tópicos, recomendando-se parágrafo único. A primeira frase deve ser significativa, explicando o tema principal do documento. Segue-se indicando a categoria do trabalho, como memória, estudo de caso, análise de situação, dentre outros. Deve-se usar o verbo na voz ativa e na terceira pessoa do singular e evitar o uso de símbolos e contrações, fórmulas, equações e diagramas apenas quando absolutamente necessário, seguidos de sua definição na primeira vez em que forem mencionados. Quando o resumo não é inserido no próprio documento, deve ser precedido de sua referência. Sua extensão é de 150 a 500 palavras. (Arial/Times New Roman; 12; Justificado; Espaçamento 1,5).

Palavras-chave: As palavras-chave devem figurar logo abaixo do resumo, antecedidas da expressão “Palavras-chave:” e separadas entre si por ponto e finalizadas também por ponto. Deve constar de 3 a 5 palavras no máximo e ordenadas por ordem de relevância.

ABSTRACT

The abstract is the presentation of the relevant points of a document. The abstract of an academic work must be the informative type, so that the reader knows the purposes, methodology, results and conclusions of the document, in such a way that it can dispense with the consultation to the original text. It is composed of a sequence of concise, affirmative phrases and cannot have enumeration of topics, recommending single paragraph. The first sentence must be meaningful, explaining the main theme of the document. It follows indicating the category of work, such as memory, case study, situation analysis, among others. The verb must be in the active voice and in the third person singular and avoid using symbols and contractions, formulas, equations and diagrams, which are used only when absolutely necessary, followed by their definition the first time they are mentioned. When the abstract is not inserted in the document itself, it must be preceded by its reference. Its length is 150 to 500 words. Keywords must appear just below the abstract, preceded by the phrase “keywords” and separated each other by a dot.

Keywords: Standardization. Academics Works. Documents.

LISTA DE FIGURAS

Elemento Pré-Textual Opcional. Elaborada de acordo com a ordem de apresentada no texto, com cada item designado por seu nome específico, travessão, título e respectivo número da folha ou página. Quando necessário, recomenda-se a elaboração de lista própria para cada tipo de ilustração (desenhos, esquemas, fluxogramas, fotografias, gráficos, mapas, orgonogramas, plantas, quadros, retratos e outras).

LISTA DE TABELAS

Elemento Pré-Textual Opcional. Elaborada de acordo com a ordem de apresentada no texto, com cada item designado por seu nome específico, acompanhado do respectivo número da folha ou página.

LISTA DE SIGLAS

PHP - PHP: Hypertext Preprocessor

SQL - Structured Query Language

URL - Uniform Resource Locator

SUMÁRIO

(EXEMPLO)

NOTA – Elaborado conforme ABNT NBR 6027

1 INTRODUÇÃO

O projeto tem a finalidade de mostrar formas de deixar o desenvolvimento, neste caso destinado à *web*, mais seguro, mitigando ao máximo, as possíveis falhas ou ataques. Para o norteamento dos principais problemas de segurança, foi usado o OWASP (Projeto Aberto de Segurança em Aplicações Web), que visa demonstrar as maiores falhas do ano e listá-las em tópicos, entre outras coisas como documentos, artigos, ferramentas do ramo, segundo o OWASP (2020).

Segundo o autor Carvalho et al.(2013), a indiferença em relação ao tema, talvez seja pela falta de conhecimento sobre essas vulnerabilidades. Para mostrá-las e corrigi-las, partiremos de uma ferramenta que contém diversos erros, até desenvolver uma aplicação sem falhas.

1.2 OBJETIVOS

1.2.1 OBJETIVO GERAL

Ressaltar a importância de uma programação pensada corretamente, que vai gerar mais segurança tanto para quem programa, para quem compra o produto e principalmente para quem usa. Provar também que a segurança é um assunto de extrema importância, que não deve ser feita a partir de lógicas ou retalhos de códigos que não seguem algum padrão, ou norma específica, para que seja discutida como a principal parte de um projeto em execução.

1.2.1 OBJETIVOS ESPECÍFICOS

- Desenvolver uma aplicação *web* ;
- Estudar a linguagem de programação PHP orientada a objeto;
- Explorar as diversas formas de proteger o sistema contra-ataques;
- Mostrar os resultados dos ataques em uma aplicação não segura;
- Mostrar as falhas de segurança que pode acontecer se a entrada e saída de dados não forem tratadas devidamente.

1.3 JUSTIFICATIVA

A cada dia, mais pessoas estão conectadas no mundo virtual, oferecendo seus dados, muitas vezes de forma não intencional. O inconveniente é que existem diversos criminosos tentando acessá-los a todo momento. Quando esses dados são expostos por alguém ilicitamente, vazamento de fotos íntimas, contas de banco, informações pessoais e empresariais, podem se tornar um transtorno maior do que se imagina, como, por exemplo, a fraude em uma eleição devido a dados pessoais.

Segundo o autor Machado et al.(2019), as pesquisas mostram um aumento considerável tanto dos dados vazados, quanto a frequência com o que isso ocorre. Com isso, o mercado de trabalho vem procurando mais profissionais especializados em segurança, já que a maioria das empresas trabalha diretamente com dados pessoais, como os bancos, e falhas como vazamento de informações são inadmissíveis. Portanto, este trabalho ressaltará o valor de se investir nessa área, pois existem diversas vulnerabilidades já conhecidas.

1.4 FUNDAMENTO TEÓRICO

De acordo com os autores, (LÜDERS; BIANCHINI, 2017), a gestão pública vem adotando medidas que melhoram a segurança das Cidades Inteligentes. Elas formam uma rede de informações que em sua maioria, usa a linguagem de programação PHP em 82,9% e *softwares* livres. A segurança deve estar em todas as fases do desenvolvimento, pois, a cada dia, a sofisticação dos ataques vem crescendo. As principais vulnerabilidades testadas são: *SQL Injection*, consiste em adicionar caracteres ou comandos SQL nos campos de formulário através de parâmetros passados por *URLs*; *Cross-Site Scripting*, ela permite inserir códigos maliciosos JavaScript, por exemplo, para roubar informações contidas nos *cookies*; e *Any File Inclusion*, que é quando um atacante insere um arquivo em um URL, esse arquivo pode executar comandos maliciosos.

O autor (SOUZA, 2012), diz que a segurança de aplicações *Web* não recebe a notoriedade necessária pelas empresas de desenvolvimento. Tendo isso em mente, utilizou-se a metodologia proposta pela OWASP, para realizar uma pesquisa qualitativa, com a finalidade de testar se os métodos propostos são possíveis de serem aplicados valorizando os princípios de segurança de

confidencialidade, integridade, autenticidade e disponibilidade. O resultado foi que os desenvolvedores não utilizam normas de segurança durante o ciclo vida dos projetos de *software*. Logo, a solução encontrada foi adotar técnicas e recomendações de segurança no desenvolvimento de aplicações *Web* através de metodologias para tratar riscos e ameaças.

Os autores (CERON et al., [s.d.]) também notaram que mecanismos de busca estão sendo usados como uma ferramenta para localizar *sites* vulneráveis. Algumas aplicações como *webmail*, aplicativos de gerenciamento remoto, fórum de discussões entre outros, contém as mesmas vulnerabilidades, sendo elas, *Cross Site Scripting, SQL Injection, Directory Transversal, entre outras.* Atualmente, para analisar as técnicas de ataque, estão sendo projetadas ferramentas como o PHP Honeypot Project (PHP.HoP), que emulam aplicações web vulneráveis e coletam informações sobre os acessos.

> Com os dados coletados, foi possível constatar que em um período de 53 dias, foi totalizado 4902 acessos aos serviços *web,* isso sinaliza que há uma alta procura por aplicações *web* vulneráveis. Mecanismos de busca como o Google e Yahoo são ferramentas de sondagem para portas e aplicações, é uma técnica eficiente, pois mecanismos tradicionais de segurança não a detecta.

O autor (MONTEVERDE, 2014) defende que, o aumento do uso de aplicações *Web*, facilitou não só a consolidar-se como um dos principais meios de comunicação mas também a expansão dos ataques a segurança, por outro lado, conhecer as principais vulnerabilidades, permite estabelecer medidas preventivas para garantir a segurança das aplicações, executando, por exemplo, os pilares centrais da segurança. Mas afinal o que são vulnerabilidades? São condições que podem virar falhas de segurança quando exploradas por alguém mal intencionado. Elas são construídas ao longo do desenvolvimento a partir de uma série de fatores, por exemplo, prazos curtos para entrega do trabalho, falta de qualificação técnica dos desenvolvedores ou até mesmo uma ausência de revisão contínua e atualizações. Entender essas falhas e resolvê-las é o principal passo a ser tomado para evitar custos futuros com manutenção.

De acordo com o autor (TAHA, 2017), grande parte das aplicações armazenam dados sensíveis, tendo isso em mente, foi desenvolvido um guia de testes de segurança usando a metodologia OWASP como base, para ajudar os desenvolvedores em diversos estágios de aprendizado. O guia conta técnicas de teste, algumas delas são: a revisão e inspeção de manuais, que são documentos

que contém informações sobre a arquitetura da aplicação; modelagem de ameaças, permite que os desenvolvedores identifiquem as ameaças antes do *software* ser desenvolvido; revisão do código fonte e muitas outras. Ele realiza testes de penetração em ambiente controlado, demonstra experimentos explicando as ferramentas usadas e apresenta as vulnerabilidades encontradas através dos procedimentos realizados.

Os autores (MONTANHEIRO; CARVALHO, 2018) assumem que, as equipes de desenvolvimento de sistemas apresentam uma eminente falta de conhecimento e atenção a respeito de segurança da informação, e é claro que isso ocasiona falhas. Uma justificativa para esse erro, é que cada vez mais os *softwares* são desenvolvidos em um prazo curto de tempo e são mais complexos. Então, os gestores optam por poupar custo e tempo, deixando a segurança como um acessório e não prioridade. Porém, toda a equipe de desenvolvimento deveria receber um treinamento sobre o assunto, com a finalidade de mitigar os riscos nas aplicações e entender como as falhas podem gerar vulnerabilidades.

Isso é comprovado pelos autores (COSTA et al., 2018), que em uma entrevista com desenvolvedores *web*, visando mensurar o nível de conhecimento dos programadores em identificar, analisar e corrigir ameaças a segurança do código, recolheu dados alarmantes. Dos sessenta resultados obtidos pelo *survey, 73*% são respostas de profissionais com mais de 3 anos de experiência e 57% contém curso superior. Mesmo assim, a pesquisa revela que os motivos para não implementação de prevenções contra os ataques são: falta de conhecimento em 92%; prazos muito apertados em 70% e baixa prioridade em 47%.

Além disso 90% dos entrevistados disseram que a formação acadêmica não oferece uma preparação adequada para lidar com segurança, outros 50% considera o ensino insuficiente, enquanto que 37,5% afirmam que o assunto não foi abordado em sua formação. Mas dos que conhecem o assunto, 47% aprenderam através de livros, cursos ou documentações oficiais, enquanto 17%, aprenderam com a prática da profissão. E apenas 30% afirmaram ter aprendido durante a formação acadêmica. São números extremamente alarmantes quando esses profissionais desenvolvem diversos aplicativos, desde redes sociais até aplicativos de bancos, logo, isso pode afetar a vida de qualquer um e fazer diversos estragos.

2 CONCLUSÕES

Parte do texto que apresenta os resultados correspondentes aos objetivos ou hipóteses levantadas na introdução. O mesmo se aplica se o resultado é a proposta de um produto ou processo. Descreve de forma resumida o que se aprendeu sobre o tema, até mesmo propostas para outros trabalhos referentes ao assunto. Deve estar coerente com o desenvolvimento e relacionado à introdução. Pode ainda estabelecer relações com outros fatos referentes à mesma matéria. Em trabalhos acadêmicos, o termo “Conclusão” é substituído por “Considerações Finais”.

REFERÊNCIAS

CERON, J. M. et al. Vulnerabilidades em Aplicações Web: uma Análise Baseada nos Dados Coletados em Honeypots. p. 2, [s.d.].

COSTA, P. V. et al. Nível de conhecimento de desenvolvedores sobre segurança em aplicações web: Pesquisa e análise. **Anais da Escola Regional de Sistemas de Informação do Rio de Janeiro (ERSI-RJ)**, p. 92–99, 16 out. 2018.

MONTANHEIRO, L. S.; CARVALHO, A. M. M. **Primeiros passos para o Desenvolvimento Seguro de Aplicações Web**. . In: ANAIS ESTENDIDOS DO XVIII SIMPÓSIO BRASILEIRO EM SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS. SBC, 25 out. 2018Disponível em: <https://sol.sbc.org.br/index.php/sbseg_estendido/article/view/4162>. Acesso em: 2 abr. 2020

MONTEVERDE, W. A. ESTUDO E ANÁLISE DE VULNERABILIDADES WEB. p. 82, 2014.

SOUZA, L. L. D. DESENVOLVIMENTO SEGURO DE APLICAÇÕES WEB SEGUINDO A METODOLOGIA OWASP. p. 71, 2012.

TAHA, A. M. DA C. Guia de testes de segurança para aplicações web. 2017.

GLOSSÁRIO

(EXEMPLO)

ACESSO DEDICADO - forma de acesso à Internet no qual o computador fica conectado permanentemente com a rede mundial de computadores. Normalmente, o acesso dedicado é utilizado por empresas que vendem acesso e serviços aos usuários novos.

ACESSO DISCADO (DIAL-UP) - é o tipo de acesso por telefone dos usuários comuns. Para utilizá-lo, basta um computador, linha telefônica e modem. O usuário utiliza o computador (com um programa de comunicação) para fazer a ligação até o seu fornecedor de acesso (provedor pago ou gratuito). Ao ser recebido pelo computador do provedor, deve indicar seu nome de usuário e senha para poder entrar no sistema.

ATTACHMENT ("ARQUIVO ATACHADO") - envio de um arquivo associado a uma mensagem. A maioria dos programas de correio eletrônico, como o Eudora e Outlook permitem que arquivos sejam enviados junto com uma mensagem. Esses arquivos podem ser textos, fotos entre outros. Ao chegar no destinatário, os arquivos associados podem ser copiados para o computador.

CHAT - conversa em tempo real através do computador. Em alguns sistemas mais antigos de chat, a tela é dividida em duas. Cada parte contém o texto de um dos interlocutores. Novos sistemas permitem a criação de "salas" de conversa em páginas de Web. O chat na Internet ficou famoso através dos servidores de IRC (Internet Relay Chat), onde são criadas as várias "salas" ou "canais" para abrigar os usuários.

Elemento Pós-textual Opcional. Dicionário de palavras de sentido obscuro ou pouco conhecido e indicado para trabalhos que utilizam muitos termos pouco usuais.

APÊNDICE A – TÍTULO DO APÊNDICE

Elemento Pós- Textual Opcional. Texto ou documento elaborado pelo autor, a fim de complementar sua argumentação, sem prejuízo da unidade nuclear do trabalho. Exemplo: questionário aplicado para levantamento de dados.

ANEXO A – TÍTULO DO ANEXO

Elemento Pós-Textual Opcional. Texto ou documento não elaborado pelo autor, que serve de fundamentação, comprovação e ilustração. Exemplo: dados estatísticos do IBGE.