PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS

CENTRO DE CIÊNCIAS EXATAS, AMBIENTAIS E DE TECNOLOGIA

Edson Lüders

SMART CITIES: SEGURANÇA EM APLICAÇÕES WEB - PREFEITURAS:
Proposta de uma metodologia simplificada para avaliação e operação de um ambiente seguro.

CAMPINAS

2019

Edson Lüders

SMART CITIES: SEGURANÇA EM APLICAÇÕES WEB - PREFEITURAS: Proposta de uma metodologia simplificada para avaliação e operação de um ambiente seguro.

Dissertação como exigência curricular para obtenção do título de Mestre em Engenharia Elétrica do Programa de Pós-Graduação em Gestão de Redes e Telecomunicações , do Centro de Ciências Exatas, Ambientais e de Tecnologia da Pontifícia Universidade Católica de Campinas.

Orientador: Prof. Dr. David Bianchini

CAMPINAS
2019

Ficha catalográfica elaborada por Vanessa da Silveira CRB 8/8423
Sistema de Bibliotecas e Informação - SBI - PUC-Campinas

005.8 Lüders, Edson.
L944s Smart cities: segurança em aplicações web - prefeituras: proposta de uma metodologia simplificada para avaliação e operação de um ambiente seguro / Edson Lüders.- Campinas: PUC-Campinas, 2019.
102 f.: il.

Orientador: David Bianchini.
Dissertação (Mestrado em Engenharia Elétrica) - Centro de Ciências Exatas, Ambientais e de Tecnologias, Pontifícia Universidade Católica de Campinas, Campinas, 2019.
Inclui anexo e bibliografia.

1. Redes de computação (Medidas de segurança). 2. Sistemas de segurança. 3. Software livre. 4. PHP (Linguagem de programação de computadores). I. Bianchini, David. II. Pontifícia Universidade Católica de Campinas. Centro de Ciências Exatas, Ambientais e de Tecnologias. Programa de Pós-Graduação em Engenharia elétrica. III. Título.

CDD - 22. ed. 005.8

EDSON LÜDERS

"SMART CITIES: SEGURANÇA EM APLICAÇÕES WEB - PREFEITURAS: Proposta de uma metodologia simplificada para avaliação e operação de um ambiente seguro"

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro de Ciências Exatas, Ambientais e de Tecnologias da Pontifícia Universidade Católica de Campinas como requisito parcial para obtenção do título de Mestre em Gestão de Redes de Telecomunicações.
Área de Concentração: Engenharia Elétrica.
Orientador: Prof. Dr. David Bianchini

Dissertação defendida e aprovada em 25 de fevereiro de 2019 pela Comissão Examinadora constituída dos seguintes professores:

Prof. Dr. David Bianchini
Orientador da Dissertação e Presidente da Comissão Examinadora
Pontifícia Universidade Católica de Campinas

Prof. Dr. Antônio Carlos Demanboro
Pontifícia Universidade Católica de Campinas

Prof. Dr. Edson Luiz Ursini
UNICAMP

A minha amada mãe, pelo exemplo de vida e luta, sendo fonte de constante inspiração em toda minha vida. Ao meu pai e avós pelo exemplo de vida e honestidade que fortemente lapidaram meu caráter e irei carregar por toda a vida. A minha esposa Lucilene, minha filha Júlia e meu filho Samuel pelo amor, carinho, suporte nos momentos de dificuldade e desafios que enfrentamos em nossa breve caminhada.

Agradecimentos

Ao Prof. Dr. David Bianchini,

Pelo sempre apoio e incentivo ao longo do desenvolvimento desse trabalho e pela motivação constante em ensinar e partilhar seus conhecimentos.

A todos os professores, mestres e doutores do curso de Pós-Graduação em Engenharia Elétrica pela amizade, comprometimento, paciência e disposição constante em compartilhar seus conhecimentos.

Á minha família pelo apoio e suporte, paciente nas ausências, e inspirando a vencer os desafios com perseverança e coragem.

A Pontifícia Universidade Católica de Campinas por me conceder a bolsa de estudos e portanto sendo determinante para a conclusão desse trabalho.

“Procure ser um homem de valor, em vez de procurar ser um homem de sucesso”.

Albert Einstein

(1879-1955)

Resumo

LÜDERS, Edson. Proposta metodológica simplificada para avaliação e operação de aplicações WEB nas Cidades Inteligentes em um ambiente seguro. 2019. Dissertação (Mestrado em Engenharia Elétrica) - Programa de Pós-Graduação em Engenharia Elétrica, Curso de Gestão de Redes de Telecomunicações, do Centro de Ciências Exatas, Ambientais e de Tecnologias. Pontifícia Universidade Católica de Campinas. Campinas. 2019.

Nos últimos anos o desenvolvimento de aplicações para Cidades Inteligentes vem aumentando, como pode ser notado, na diretiva do governo na adoção de *software* livre e parcerias pública privadas no desenvolvimento de soluções que atendam a gestão pública. Nesse contexto observa-se que a linguagem PHP é utilizada atualmente em 82,9% dos web-sites mundialmente. A gestão pública dá seus primeiros passos na informatização de seus serviços e, quando existente, ainda é, em muitos casos por meio de desenvolvimentos internos ou desenvolvimentos isolados. Contudo, existe a iniciativa do Ministério do Planejamento que desde 2005 investe no desenvolvimento e padronização, em parceria com a iniciativa privada, disponibilizando o sistema e-cidade e seu código- fontes de forma gratuita para as prefeituras. Devido ao aumento dos ataques cibernéticos e do nível de sofisticação desses ataques nota-se claramente que a segurança da informação se torna um dos elementos fundamentais e decisórios no sucesso do desenvolvimento, implantação e sustentação de *websites* voltados aos serviços a população. A presente dissertação visa apresentar as principais e atuais vulnerabilidades, destaca as três vulnerabilidades mais exploradas em linguagem PHP com soluções para corrigi-las. Efetua um teste de vulnerabilidade nos softwares e-CIDADE, e-SIC Livre e SIVAC, descrevendo os resultados obtidos com o uso da ferramenta Netsparker Desktop e, por fim, apresenta de forma simplificada os processos de segurança, fundamentado no *framework* COBIT 5 e nos processos DSS05 - Gerenciar Serviços de Segurança, para avaliação e sustentação de um ambiente robusto e seguro através da análise de três estudos de casos. Finalmente os pontos levantados servirão para tomada de decisão, objeto para futuras pesquisas e busca de novas soluções.

Termos de indexação: Segurança de Sistemas, Aplicações WEB e Cidades Inteligentes.

Abstract

LÜDERS, Edson. Simplified methodological proposal for evaluation and operation of WEB applications in Smart Cities in a secure environment. 2019. Dissertation (Master's Degree in Electrical Engineering) - Graduate Program in Electrical Engineering, Course of Management of Telecommunications Networks, Center of Exact, Environmental and Technologies Sciences. Pontifical Catholic University of Campinas. Campinas. 2019.

In recent years the development of applications for Smart Cities has been increasing, as can be noticed, in the government's directive on the adoption of free software and public-private partnerships in the development of solutions that comply with public management. In this context it is observed that the PHP language is currently used in 82.9% of web-sites worldwide. Public management takes its first steps in computerizing its services and, when it exists, is still, in many cases through internal developments or isolated developments. However, there is the initiative of the Ministry of Planning that since 2005 invests in the development and standardization, in partnership with the private initiative, making the systems and its source code available free of charge to city halls. The present dissertation aims to present the main and current vulnerabilities, highlights the three vulnerabilities most exploited in PHP language with solutions to correct them. It performs a vulnerability test in the systems: e-CIDADES, e-SIC Livre and SIVAC, describing the results obtained with the use of the Netsparker Desktop tool and, finally, presents in a simplified way the security processes, in the COBIT 5 framework and in the processes DSS05 - Manage Security Services, to evaluate and sustain a robust and secure environment through the analysis of three case studies. Finally the points raised will serve as decision-making, object for future research and search for new solutions.

Index terms: System Security, WEB applications and Smart Cities

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AB	Atenção Básica
AM	Amazonas
AST	Abstract Syntax Tree (em português, Árvore Sintática Abstrata)
BA	Bahia
BI	Business Intelligence
BSC	Balanced Scorecard
CISL	Comitê Técnico de Implementação de Software Livre
COBIT	Control Objectives for Information and related Technology
Cookie	Cookie é um arquivo de texto simples, cuja sua composição depende diretamente do conteúdo do endereço da página WEB acessada.
CPF	Cadastro de Pessoas Físicas
CSS	Cascading Style Sheets
DA	Despesas Administrativas
DF	Distrito Federal
DMZ	Desmilitarized Zone
DOM	Document Object Model
ERP	Entrerprise Resource Planning
ES	Espirito Santo
EUA	United States of America
FRAMEWORK	conceitual é um conjunto de conceitos usado para resolver um problema de um domínio específico.
GB	Gigabyte
HD	Hard Disk
HTML	HyperText Markup Language
HTTP	Hyper Text Transport Protocol

HTTPS	Hiper Text Transfer Protocol Secure
IBM	International Business Machines
IPTU	Imposto sobre a Propriedade Predial Urbana
ISACA	Information Systems Audit and Control Association
ISMS	Information Security Management System (Sistema de Gerenciamento da Segurança da Informação)
MG	Minas Gerais
MT	Mato Grosso
OAB	Ordem dos Advogados do Brasil
PE	Pernambuco
PHP	Hypertext Preprocessor
RAM	Random Access Memory - Memoria de Acesso Aleatorio
RS	Resíduos Sólidos
SC	Santa Catarina
SD	Secure Digital Card
SDK	Software Development Kit
SE	Sergipe
SMS	Short Message Service
SP	São Paulo
SQL	Structured Query Language
SSL	Secure Socket Layer
SUS	Sistema Único de Saúde
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
TLS	Transport Layer Security
TSE	Tribunal Superior Eleitoral

UF	Unidade Federativa
UFMT	Universidade Federal de Mato Grosso
URL	Uniform Resource Locator (Localizador Padrão de Recurso)
VLAN	Virtual Local Area Network
VPN	Virtual private network
WEB	World Wide Web
XSS	Cross-site scripting

Sumário

1 INTRODUÇÃO

Nos últimos anos o desenvolvimento e implantação das Cidades Inteligentes vem ao encontro de uma sociedade cada vez mais conectada pelas facilidades advindas dos avanços tecnológicos em comunicação (TICs) ocorridas nos últimos anos.

Indiscutivelmente, essas tecnologias trouxeram mudanças profundas em vários aspectos quer sociais, econômicos e políticos, surgindo novas demandas, tais como: crescimento exponencial do comércio eletrônico de produtos e serviços, rápida adoção de transações eletrônicas bancárias e financeiras, uso excessivo das redes e mídias sociais, uso de aplicativos de mensagens instantâneas, bem como muitas outras mudanças profundas na forma e na maneira como as pessoas estão interagindo, isto refletindo diretamente nas relações pessoais, com as empresas e com o setor público.

Esses avanços tecnológicos trazem inúmeros benefícios para o setor público, com uma maior proximidade das demandas de seus munícipes, conhecendo melhor suas necessidades e anseios. Outro fator importante refere-se ao acesso às informações que, de uma forma mais rápida do que se consegue atualmente, pode se tornar decisiva para uma melhor tomada de decisões táticas e estratégicas na administração pública. Neste ponto, vislumbram-se ganhos efetivos para a aprovação de projetos de investimento em áreas importantes e essenciais, tais como, na educação, na saúde, na segurança, no transporte e na mobilidade urbana, no uso dos recursos naturais e para o meio ambiente.

Por outro lado, o gerenciamento público, em especial as prefeituras do Brasil, estão dando seus primeiros passos em busca de um modelo de governança e eficiência. Tanto para os seus processos internos, quanto para seus sistemas, onde, em sua maioria existem sistemas legados e aplicações desenvolvidas internamente e, em poucos casos, sistemas adquiridos de empresas por processo de licitação.

Evidencia-se também o desenvolvimento de sistemas em parceria pública-privada, tais como o E-CIDADE, E-SUS AB, E-SIC, SIVAC e dentre outras iniciativas.

Assim, verifica-se a adoção de *software* livre por todas as esferas do governo, sendo que esta estratégia vem se consolidando como um padrão. Dentre as tecnologias adotadas, destaca-se a utilização do sistema operacional e de servidor Linux, de servidores WEB em Apache, de bancos de dados em PostgreSQL e MySQL, da linguagem de desenvolvimento PHP (*Hypertext Preprocessor*), que atualmente é utilizada em 82,9% dos *web-sites* mundialmente, bem como outras ferramentas e tecnologias livres.

Vislumbra-se que a crescente necessidade da informatização dos municípios e a implantação de cidades inteligentes, irão aumentar exponencialmente o desenvolvi-

mento de soluções e aplicações voltadas para o serviço à população.

Porém, a estes desenvolvimentos atuais e futuros, deve-se dar uma atenção especial à segurança da informação em todas as fases de construção, desde o planejamento, desenvolvimento, implantação e atualizações, pois o crescente número de ataques cibernéticos e a sofisticação desses ataques, vem crescendo dia a dia.

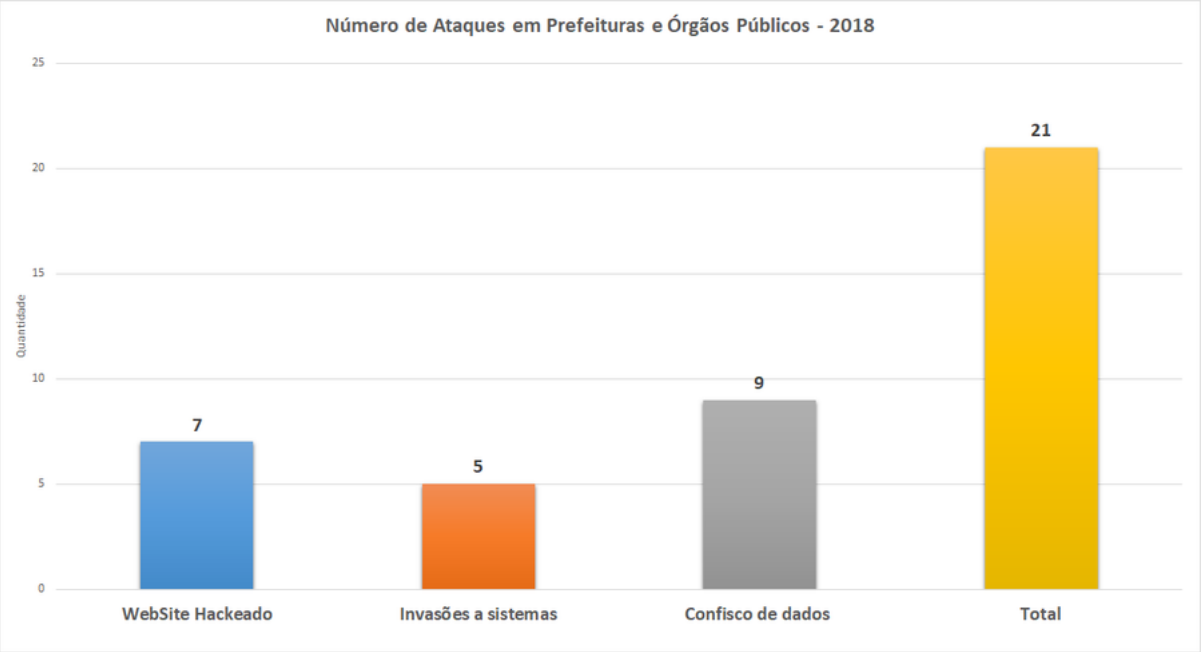
A literatura da área aponta, como os autores (KIEZUN et al., 2009), (MERLO; LATARTE; ANTINIOL, 2007) e (ALTAF et al., 2015),que nos desenvolvimentos WEB, as 3 vulnerabilidades mais comuns exploradas pelos hackers são: *SQL Injection*, *Cross-Site Scripting* e *Any File Inclusion*. Existem pesquisas e artigos mostrando como resolver ou mitigar, em sua grande maioria, esses riscos (VOITOVYCH; YUVKOVETSKI; KUPERSHTEIN, 2016), (KUMAR; REDDY, 2014) e (HAQUE; MIAH; MASUD, 2018).

Destaca-se o projeto *Open Web Application Security Project* (OWASP), uma entidade sem fins lucrativos e com a participação de desenvolvedores em diversos países, que identifica, classifica e trabalha na prevenção dos ataques em aplicações WEB, fornecendo conhecimento técnico, material, fóruns de discussão e ferramentas.

Em adição a uma solução ou mitigação dos riscos inerentes ao desenvolvimento em linguagem PHP, recomenda-se a instalação em ambientes computacionais seguros, pois são de vital importância para garantir a disponibilidade, confidencialidade e integridade das informações. . É importante ressaltar que, considerando a amplitude do COBIT 5, recomenda-se aqui uma metodologia mais simplificada, aplicando-se tão somente sete dos trinta e sete processos do COBIT 5, dando maior objetividade ao processo na construção e operação de um ambiente seguro.

Na Figura 19 constam os ataques ocorridos, que foram divulgados por meios alguns meios de comunicação, no ano de 2018, em Prefeituras e Órgãos Públicos brasileiros. A lista completa e detalhada encontra-se no Anexo B.

Figura 1 – Ataques a Prefeituras e Órgãos Públicos em 2018



Fonte: Pesquisa do autor em diversos meios de comunicação (2018)

Web-site 'Haqueado' significa que um invasor obteve acesso ao seu web-site possibilitando a alteração de conteúdos, acesso a dados sensíveis, instalar códigos maliciosos ou vúrus e contaminar outros usuários, derrubar seu site e postar mensagens obcenas.

Invasões a sistemas significa que um invasor burlou o acesso do sistema ou do ambiente TIC onde esta hospedado, tendo acesso, alterando ou excluindo informações sensíveis e, até mesmo, causar a perca dos dados e dis sistema invadido.

Confisco e Dados significa que um hacker obteve acesso ao ambiente TIC ou a um computador isoladamente, roubando ou criptografando dados sensíveis e exigindo pagamento, geralmente em moedas virtuais (*BitCoins*), para a devolução ou resgate dos dados sequestrados.

2 OBJETIVO

Este trabalho busca responder a questão: Como aumentar a segurança dos web-sites desenvolvidos em linguagem PHP em Prefeituras?

Os conceitos apresentados são temas de estudos científicos, revistas especializadas, bases de conhecimento teórico e prático e são fortemente utilizados em diversas organizações privadas na prevenção e proteção de seus sistemas e dados. Portanto o objetivo deste trabalho é apresentar uma metodologia simplificada para avaliação de web-sites em prefeituras, visando:

- Documentar as três mais exploradas vulnerabilidades em aplicações WEB e como resolvê-las;
- Propor o uso de uma ferramenta de análise de vulnerabilidades para se identificar possíveis brechas e vulnerabilidades de segurança existentes em um sistema;
- Recomendar a adoção de sete dos trinta e sete processos existentes no *framework* de governança COBIT 5, deste modo, considera-se aqui um processo de simplificação, sendo os sete processos estão diretamente relacionados com a segurança da informação em ambientes TIC, e que ao serem aplicados dão suporte a um ambiente computacional seguro.

Este trabalho está estruturado de forma a apresentar as fases da pesquisa realizada, o desenvolvimento e os resultados da metodologia simplificada para avaliação e operação de um ambiente seguro. A dissertação é composta dos seguintes capítulos:

No Capítulo 1 faz-se a introdução das expectativas, mudanças e benefícios aos gestores públicos e à população com respeito à implantação das cidades inteligentes. Apresentam-se algumas iniciativas do governo na distribuição e padronização de sistemas, tais como o e-Cidade, e-SIC e SIVAC, dentre eles, alguns desenvolvidos em parcerias públicas privadas. Contextualiza-se a adoção do *software* livre pelo governo e destaca-se algumas das tecnologias adotadas. Por outro lado, apresentam-se os riscos existentes, o crescimento e sofisticação dos ataques cibernéticos, apresenta-se as três vulnerabilidades mais exploradas pelos *hackers* e os ataques ocorridos em prefeituras e órgãos públicos em 2018. Desta forma, procura-se mostrar a importância no uso de ferramentas e processos que garantam a segurança dos sistemas e do ambiente TIC.

O aporte teórico traz o estado da arte no Capítulo 3, que aborda as Cidades Inteligentes, o uso do *Software* Livre em servidores, linguagens de programação e bancos de dados, as principais vulnerabilidades conhecidas com base na lista OWASP *Top Ten* (2017), métodos e ferramentas de análise de vulnerabilidades e, por fim,

apresenta-se o *framework* COBIT 5, seus objetivos, processos e como é a avaliação do nível de maturidade.

No Capítulo 4 apresenta-se o método proposto, que consiste no entendimento do sistema e ambiente TIC, determinação do nível de maturidade esperado e atual dos sete processos de COBIT 5 propostos, descritos na Tabela 9, aplicação de uma ferramenta para a análise de vulnerabilidades do sistema. Realiza-se a análise das vulnerabilidades e riscos encontrados, propondo correções e remediações. São apresentados três sistemas avaliados em laboratório, que são: e-Cidade versão 2018-2, e-SIC Livre versão 1.4 e o sistema SICAC versão 2.1.

No Capítulo 5, apresentam-se os resultados obtidos pela análise de vulnerabilidade, com uso do software NETSPAKER 4.9, dos sistemas e-Cidade, e-SIC e SIVAC, como foi construído o ambiente de testes em laboratório para cada sistema, bem como os recursos e *softwares* utilizados.

No Capítulo 6 são discutidas as vulnerabilidades encontradas nos sistemas avaliados e propostas correções e mitigações das vulnerabilidades e dos riscos.

No Capítulo 7 são apresentados três estudos de caso de ataques a prefeituras e órgãos públicos, ocorridos em 2018 com base na pesquisa apresentada no Anexo B, sendo proposta a implantação ou melhoria dos sete processos de COBIT 5 para a correção e melhoria do ambiente TIC dos casos avaliados. Reforça-se aqui a necessidade de uma avaliação mais detalhada e completa dos estudos de caso apresentados, pois foram avaliados como base nos fatos divulgados pela imprensa.

Por fim, no Capítulo 8, apresenta-se a conclusão deste trabalho e a importância no uso do método proposto para se construir e operar sistemas e ambientes TIC seguros, visando contribuir para a tomada de decisão dos gestores públicos, na escolha e adoção de sistemas, ferramentas e processos que garantam disponibilizar sistemas e serviços WEB seguros à população, bem como o uso adequado dos recursos TIC, reduzindo os riscos de ataques cibernéticos, invasões, sequestro de dados, protegendo a confidencialidade, integridade, confiabilidade e disponibilidade das informações e dos serviços.

3 ESTADO DA ARTE

3.1 CIDADES INTELIGENTES

O conceito de Cidades Inteligentes vem sendo amplamente discutido no meio científico. Casos práticos e de sucesso já foram implementados em algumas cidades do Brasil tais como: Barueri (SP), Santos (SP), Tubarão (SC) e Vinhedo (SP), norteando padrões e soluções, buscando trazer benefícios à população (TEIXEIRA, 2018).

As cidades procuram implementar melhorias em seus sistemas e serviços de forma a se tornarem “inteligentes”. Entretanto, são confrontadas dentro de um ambiente de ameaças, riscos e oportunidades (NEGRE; ROSENTHAL-SABROUX; GASCÓ, 2015).

Em serviços inteligentes, a privacidade ganha uma vital importância em garantir as informações dos cidadãos (ELMAGHRABY; LOSAVIO, 2014).

Os dispositivos, sensores e sistemas inteligentes, tais como: medidores de energia inteligentes, sensores de estacionamento e trânsito, sensores de clima, e entre outros, devem elevar seu nível de segurança, de forma a não causar grandes impactos à segurança dos dados e à privacidade dos cidadãos (SEN et al., 2013)

Fornecedores de soluções de tecnologia, tais como: SAP, ERICSSON, SIEMENS, IBM, entre outros, avaliam esta nova fatia promissora de mercado, onde existe um grande potencial de novos desenvolvimentos de soluções. No Brasil, já existem empresas especializadas em prestar serviço de análise, estratégia e implementação de cidades inteligentes tais como: Instituto das Cidades Inteligentes (ICI), *Smart Cities Solutions* (iCities) e outras. Feiras e congressos especializados, como a “Smart City Expo” que aconteceu em Curitiba em 2018, trazem grandes nomes na área de Cidades Inteligentes, promovem palestras, debates e apresentam projetos e soluções (SMART CITY EXPO, 2018).

No desenvolvimento e implantação de novas soluções é imprescindível que sejam, desde o início do projeto, utilizadas soluções adequadas de segurança, por meio da aplicação de tecnologias como barreiras de segurança (*firewalls*), criptografia, segregação da rede por meio do uso de zonas desmilitarizadas (DMZ), sub-redes fornecendo uma segurança adicional, onde os serviços que possuem acessos externos ficam isolados dos acessos da rede interna, bem como a utilização de sistemas de monitoramento, que alertam contra ataques e/ou intrusões (IDS - *Intrusion Detection System*). Também, na adoção de uma determinada tecnologia ou linguagem de programação, construindo as aplicações de forma segura, aplicando sistemas de análise de vulnerabilidades, de forma, a impedir que *hackers* explorem possíveis e potenciais brechas no sistema (ORTNER et al., 2015).

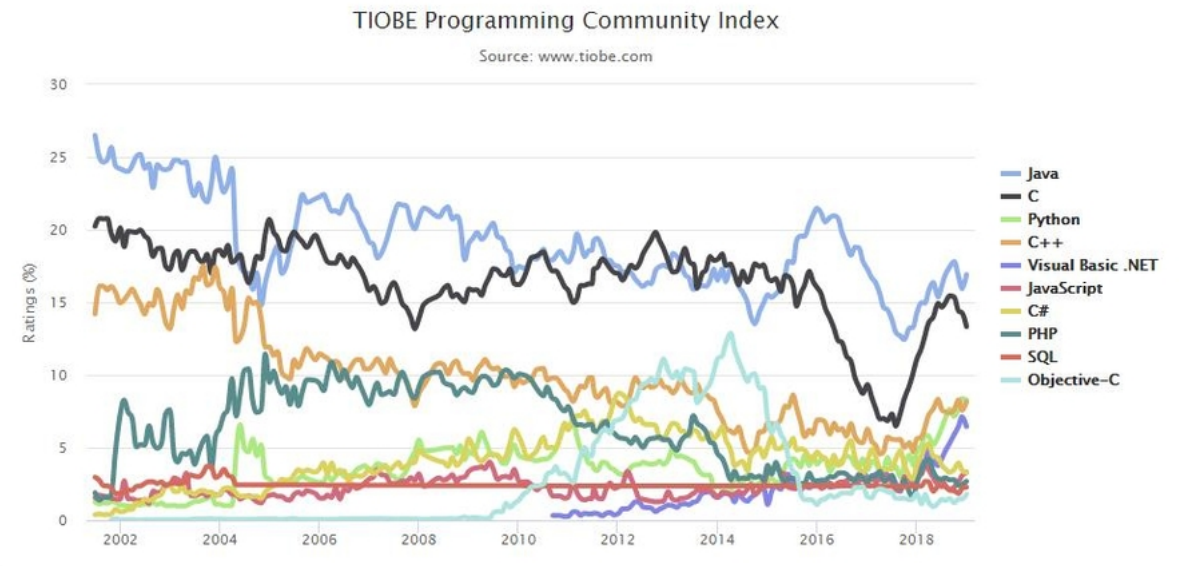
3.2 Software Livre, Linguagem de Programação e Bancos de Dados

A adoção do software livre ganhou importância e adeptos em todo o mundo, gerando inúmeras aplicações e soluções, que, no que lhe concerne, atendem desde sistemas operacionais de uso pessoal, *softwares* de colaboração e até poderosos servidores que suportam diversos serviços, tais como: servidores WEB, servidores de bancos de dados, servidores de correios eletrônicos, entre outros (MINISTÉRIO DE PLANEJAMENTO, DESENVOLVIMENTO E GESTÃO, 2015).

Esse fenômeno ocorreu, principalmente, por três fatores predominantes, a saber: pelo alto custo do investimento em licenciamento de *softwares* comerciais, pela evolução tecnológica das aplicações e soluções de *software* livre, se equiparando e, em alguns casos, superando as aplicações comerciais e, por fim, a participação ativa de comunidades, fomentando o desenvolvimento, treinamento, cooperativismo e distribuição (MILENA, 2013).

Para as aplicações WEB, hospedadas em servidores, a linguagem PHP é adotada em 83,4% dos *web-sites* desenvolvidos em todo o mundo, sendo avaliada em 2018 como a 8ª linguagem mais popular de acordo com o índice TIOBE apresentado na Figura 2 (TIOBE, 2019)[1].

Figura 2 – Índice das linguagens de progamação mais utilizadas



Fonte: Tiobe (2019)

O *software* livre vem ganhando espaço e importância na informatização dos municípios em virtude dos desafios enfrentados pelas cidades frente à arrecadação e

[1] TIOBE - The software quality company - é reconhecida mundialmente por ser especializada em avaliar e rastrear a qualidade do software. Maiores detalhes em www. tiobe.com

à defasagem na prestação dos serviços básicos à população. Nesse sentido, a informatização não pode consumir os valores previstos para os projetos de infraestrutura, educação, saúde e segurança pública (CISL - COMITÊ DE IMPLEMENTACÃO DE SOFTWARE LIVRE, 2018).

Dessa forma, a adoção do *software* livre, se torna uma solução viável frente ao uso de *softwares* comerciais, gerando uma economia decorrente do alto custo de licenciamento, dos altos valores gastos com serviços de consultoria para a implantação e manutenção e, também, não precisa de todo o processo burocrático de licitações, definido na Lei 8666, que regulamenta o processo de compra e aquisição de serviços e produtos pelo setor público (BRASIL - SENADO FEDERAL, 2017).

Por outro lado, a crescente necessidade da informatização dos municípios e a implantação de cidades inteligentes, devem aumentar exponencialmente a busca e desenvolvimento de novas soluções, aplicações e sistemas, objetivando melhorar os serviços à população, forçando os gestores públicos a repensar e melhorar os processos internos, sistemas e serviços, dando mais transparência às informações (OSÓRIO et al., 2005).

Desde 2005, o Ministério do Planejamento vem fomentando o desenvolvimento de *software* livre em parceria com empresas privadas, dentre eles, o sistema e-CIDADE, que é um sistema denominado *Enterprise Resource Planning* (ERP) voltado para o gerenciamento dos municípios, o qual é composto pelos módulos de Educação, Saúde, Financeiro, Patrimonial, Cidadão, Gestor, Recursos Humanos, *Business Inteligence* (BI) e Geoprocessamento (MINISTÉRIO DE PLANEJAMENTO, DESENVOLVIMENTO E GESTÃO, 2018a).

Destaca-se o sistema e-SIC Livre (Sistema de informação ao Cidadão), também uma iniciativa de distribuição gratuita voltado a prefeituras e órgãos públicos, visa ser um canal de maior transparência e informações, de forma clara e direta, à toda população. Atualmente, o sistema, está em uso por diversas prefeituras e alguns órgãos públicos. Seu código-fonte, fornecido na modalidade código-aberto (*open source*), permite realizar customizações e facilita a integração com outros *web-sites*, sistemas e portais (MINISTÉRIO DE PLANEJAMENTO, DESENVOLVIMENTO E GESTÃO, 2018b).

Já o sistema SIVAC (Sistema Online de Vacinação), também com distribuição na modalidade de código aberto, permite registrar e monitorar a aplicação de vacinas, gerando uma base de dados relevante na promoção de campanhas de vacinação, controle de estoque de vacinas, analisando e monitorando os dados de diversas unidades de saúde e possibilita gerar uma caderneta da vacinação eletrônica (IPPES - INSTITUTO DE PESQUISA P. E P DA EDUCACÃO E SAÚDE, 2018).

3.3 Principais Vulnerabilidades Conhecidas

Segundo (CRESPO; CHÓEZ, 2012), as vulnerabilidades são brechas ou falhas em um sistema, permitindo que usuários mal-intencionados violem os aspectos de segurança. Nas falhas de segurança permite-se que ocorram perdas ou roubo de informações sensíveis. Define-se como uma falha de um ativo, ou grupo de ativos, uma brecha ou falha de segurança que pode ser explorada por uma ou mais ameaças (ABNT, N, 2013).

A Tabela 1 apresenta uma lista com as dez maiores vulnerabilidades presentes em aplicações WEB, que são periodicamente reclassificadas pela fundação OWASP (*Open Web Application Security Project*), desenvolvida com base em estudos compartilhados por desenvolvedores e especialistas em segurança presente em diversos países, sendo sua última edição publicada em 2017 (OWASP, 2018).

Tabela 1 – OWASP Top 10 – 2017

A1	Inserção de Códigos	A6	Configuração Incorreta de Segurança
A2	Quebra de Autenticação e Gerenciamento de Sessão	A7	Cross-Site Scripting (XSS)
A3	Exposição de Dados Sensíveis	A8	Desserialização Insegura
A4	XML Entidades Externas (XXE)	A9	Utilização de Componentes com Vulnerabilidades Conhecidas.
A5	Controle de Acesso Quebrado	A10	Monitoramento e Registros Insuficientes

Fonte: Fundação OWASP (2017)

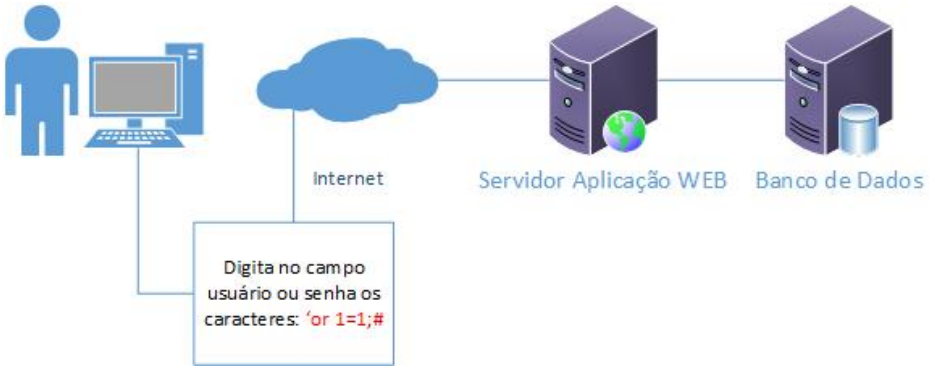
Paralelamente, a literatura aponta que, nos desenvolvimentos WEB, as 3 vulnerabilidades mais exploradas por *hackers* são: a Inserção de Códigos (*SQL Injection)* (VOITOVYCH; YUVKOVETSKI; KUPERSHTEIN, 2016) , *Cross-Site Scripting* (KUMAR; REDDY, 2014) e o *Any File Inclusion* (HAQUE; MIAH; MASUD, 2018).

A Inserção de Códigos (*SQL Injection),* um dos mais comuns de ataques, consiste em adicionar caracteres ou comandos SQL em campos de formulários da página WEB ou através de parâmetros passados nas URLs, por exemplo, em uma página de acesso com os campos de usuário e senha (VOITOVYCH; YUVKOVETSKI; KUPERSHTEIN, 2016).

Conforme (MADAN; MADAN, 2010) , uma simples sentença em linguagem SQL (*Structured Query Language*) desprotegida pode comprometer toda a segurança de uma aplicação, de seus dados ou do servidor de banco de dados. Desenvolvedores

devem ser disciplinados suficientemente para aplicar os métodos de segurança e validações em cada rotina ou função desenvolvida e em todas as páginas WEB que possuem *interface* direta com o banco de dados.

Figura 3 – Exemplo de inclusão de SQL Injection



Adaptado pelo Autor (2018)

Na Figura 3 representa-se como um ataque de Inserção de Código (*SQL Injection)* é realizado, essa vulnerabilidade consiste, basicamente, na inclusão de sentenças SQL válidas dentro de campos como, por exemplo, o usuário e a senha. Estas sentenças permitem burlar a lógica existente no código da sentença SQL da aplicação, permitindo o acesso ao sistema e suas funcionalidades. Na Figura 4, mostra-se um exemplo de código em PHP totalmente vulnerável a esse tipo de ataque, onde são inseridos códigos maliciosos, como, por exemplo, a expressão **'or 1=1;#,** no campo $sql que é recebido pelos campos GET['usuário'] e GET['senha'], resultando em uma sentença SQL válida "**select * from usuarios where usuario=" or 1=1;# senha=""**, desta forma burlando e permitindo o acesso ao sistema (KIEZUN et al., 2009).

Figura 4 – Exemplo de código em PHP com risco de SQL Injection

```
1  <?php
2  $user=$_GET['usuario'];
3  $pass=$_GET['senha'];
4  $sql="SELECT * FROM usuarios WHERE usuario ='".$user."'
5        and senha='".$pass."'";
6  $result = mysql_query($sql);
7  ?>
```

Fonte: Adaptado pelo Autor de KIEZUN (2009)

Para a correção dessa vulnerabilidade, devem ser tratados os campos de entrada, de forma, a impedir o uso de caracteres especiais. Por exemplo, na Figura 5,

se incluí duas funções, ***preg_replace*** e ***addslashes***, evitando o sucesso no uso dessa vulnerabilidade.

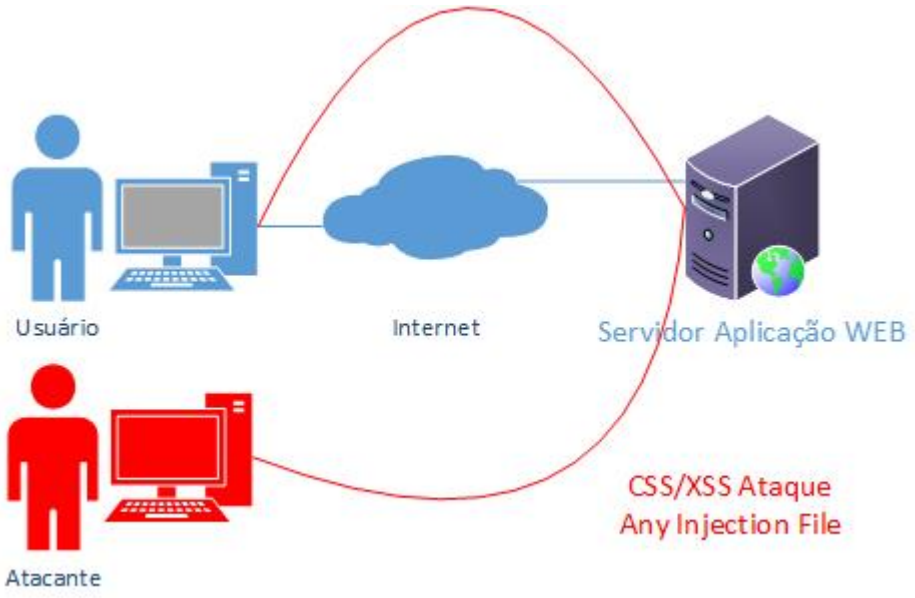
Figura 5 – Exemplo tratamento campos evitando o SQL Injection

```
1  <?php
2  $user=$_GET['usuario'];
3  $pass=$_GET['senha'];
4
5  // Tratamento dos dados de entrada
6  $user=preg_replace('/[^[:alnum:]_.-]',"",$user);
7  $pass=addslashes($pass)
8
9  $sql="SELECT * FROM usuarios WHERE usuario ='".$user."'
10       and senha='".$pass."'";
11 $result = mysql_query($sql);
12 ?>
```

Fonte: Adaptado pelo Autor de KIEZUN (2009)

O *Cross-Site Scripting,* também conhecido por CSS ou XSS, também é uma vulnerabilidade bastante explorada, pois permite inserir códigos em *JavaScript, VBScript, ActiveX* e *Flash* maliciosos através do navegador dos usuários, parecendo que são partes do *web-site*. É imperceptível para a vítima e permite o roubo de informações confidencias contidos em *cookies*. Também permite realizar *phishing*, que consiste no roubo de informações pessoais, senhas e dados bancários através de *links* e direcionamento para *web-sites* maliciosos ou se passando por pessoas ou empresas conhecidas por meio do envio de e-mails, SMS ou mensagens instantâneas, entre outras possibilidades. São classificados em 3 categorias: a) *Stored*, b) *Reflected* e c) *DOM based XSS* (DOYLE; WALDEN, 2011)

Figura 6 – Exemplo de Cross-Site Scripting (CSS/XSS) ataque Injection



Fonte: Adaptado pelo autor de DOYLE (2011)

Como pode ser observado na figura 6, o ataque CSS/XSS ocorre do lado do usuário, com a inclusão de um *web-site* com inserção de códigos maliciosos, como, por exemplo, em linguagem *JavaScript*, conforme mostrado na Figura .7

Figura 7 – Exemplo de inclusão de um código JavaScript por parâmetro

```
1  // Site com passagem de parametros/valores
2
3  http://www.exemplo.com.br/?mensagem=Você+Esta+Agora+No+Site
4
5  // É incluido outro site com códigos JavaScript Maliciosos
6  // e sendo executados pelo browse do usuário
7
8  http://www.exemplo.com.br/?mensagem=<script src=
9  "http://css.exemplo.com.br/malicioso-script.js"></script>
```

Fonte: Adaptado pelo autor de DOYGE (2011)

Para se corrigir esse tipo de ataque é necessário usar funções para validar os dados de entrada, limitar o tamanho dos campos a serem inseridos e definir um conjunto de caracteres válidos para os campos de entrada. Na linguagem *PHP* existem funções que realizam a codificação automaticamente, tais como as funções: **htmlspecialchars()** e **htmlentities()**.

O *Any File Inclusion* (ZHAO; GONG, 2015), vulnerabilidade também muito explorada, permite que um atacante insira um arquivo em uma URL de *web-site* acessado pelo usuário ou em um servidor remoto. Esse arquivo pode carregar e executar comandos maliciosos que permitem ter acesso às senhas de serviços do

usuário, roubo de dados confidenciais, ataques DoS (sigla para *Denial of Service* que consiste em enviar requisições massivas a fim derrubar ou impedir a resposta dos serviços pelo servidor) e entre outras categorias de ataque. Um exemplo de *Any File Inclusion* pode ser observado na Figura 8.

Figura 8 – Exemplo de Any File Injection em PHP

```
1) Um exemplo de Any File Injection, considere o website onde
   o valor de pagina_exemplo é invisivel para o usuário

www.website_vitima.com.br/abc.php?proxima_pagina=pagina_exemplo

2) A vulnerabilidade em PHP consite no código:

$teste = $_REQUEST["proxima_pagina"];
Include($pagina_exemplo.".php");

3) O parametro 'proxima_pagina' direciona automaticamente para
   outra página PHP
4) Um possível atague pode ser da seguinte forma:
www.website_vitima.com.br/abc.php?proxima_pagina==
http://www.ataque_website.com.br/pagina_ataque

O arquivo pagina_ataque pode conter códigos maliciosos que serão
acessados e executados pela página 'abc.php'
```

Fonte: Adaptado pelo autor de ZAHO (2015)

Neste caso, para a correção dessa vulnerabilidade recomenda-se o tratamento dos parâmetros informados nas URLs, definindo os caracteres aceitos e o tamanho máximo de caracteres passados nos parâmetros.

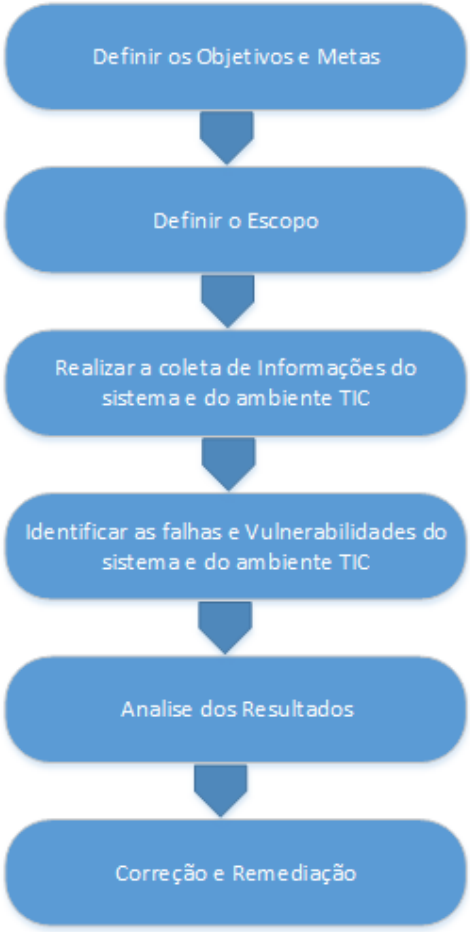
3.4 Análise de Vulnerabilidades

Define-se uma vulnerabilidade como uma falha de segurança, que quando não devidamente detectada e corrigida a tempo, pode levar à intrusão e ao comprometimento de um sistema (CERT.BR - COMITÊ GESTOR DA INTERNET NO BRASIL, 2012). De acordo com (CAMPOS, 2014), "Os ativos de informação, que suportam os processos de negócio, têm vulnerabilidades. É importante destacar que essas vulnerabilidades estão presentes nos próprios ativos, ou seja, que são inerentes a eles, e não de origem externa".

A análise de vulnerabilidades define-se por um processo de descoberta do ambiente TIC e o varrimento (i.e., *scan*) de um sistema de maneira a identificar características, propriedades e possíveis falhas e fraquezas, conhecidas ou até então não (BINGCHANG et al., 2012) (Figura 9). Trata-se de um teste que utiliza ferramentas auto-

máticas, denominadas *scanners*, que apresentam como resultado as vulnerabilidades identificadas e informações dos serviços disponíveis, aplicações, banco de dados e do sistema operacional. A análise de vulnerabilidades é uma das primeiras ações para avaliar o nível de segurança de um sistema e normalmente precede todos as outras categorias de teste de segurança (YAQOOB, 2017).

Figura 9 – Processo de Análise de Vulnerabilidades



Fonte: Adaptado pelo autor (2018)

O processo de análise das vulnerabilidades de um *web-site* pode ser realizada de forma manual, pela análise do código- fonte, mineração de dados, das sentenças em linguagem SQL e por meio do uso de ferramentas, podendo estas serem *freeware*, que são programas de computador cuja utilização não implica em pagamento de licenças de uso ou *royalties,* de código aberto ou de mercado. A aplicação de testes manuais de penetração e vulnerabilidades não são efetivos em termos de tempo e dinheiro. Desta forma, o uso de ferramentas automatizadas, *webscanners*, para a execução de testes de penetração e vulnerabilidades são melhores recomendados e utilizados (MIRJALILI; NOWROOZI; ALIDOOSTI, 2014).

Ressalta-se a importância que deve ser dada, pela equipe de desenvolvimento,

no conhecimento das vulnerabilidades existentes e como escrever seus códigos de forma segura, de maneira a eliminar possíveis brechas de segurança, que podem ser exploradas pelos *hackers* e usuários mal-intencionados. Recomenda-se a aplicação da metodologia de **Ciclo de Vida do Desenvolvimento Seguro** (SDLC - *Security Development Lifecycle*), ilustrada na Figura 10, que é largamente utilizada e possui certificação dos profissionais (PLC, 2018).

Figura 10 – Desenvolvimento de software seguro

Fonte: CALADO, Rui Pedro Cascalheira - Desenvolvimento de software seguro (2018)

As ocorrências de vulnerabilidades em sistemas WEB podem ser diminuídas através do melhor conhecimento e de treinamento dos desenvolvedores e, também, por meio da aplicação de ferramentas que realizam testes de segurança (DOUPÉ et al., 2012).

Atualmente existem dois métodos de análise de vulnerabilidades aplicados para análise de códigos-fonte e *scripts* de linguagem, que são a **análise estática** e a **análise dinâmica**. A análise estática permite a auditoria dos códigos-fonte e a detecção de erros durante o processo de compilação (HAUZAR; KOFRON, 2012) . Muitas linguagens, como a PHP, ASP, entre outras, são interpretadas, desta forma, pode-se utilizar de ferramentas de análise gramatical (*Parsers*) tais como: Zend Engine, PhpParser, Phc, HHVM e entre outras (MEDEIROS; NEVES; CORREIA, 2016) .

Pelo processo de análise estática, os códigos-fontes são re-codificados em AST (*Abstract Syntaxe Tree*), onde é possível a análise da estrutura de dados, possibilitando uma visão em formulários de seus dados de saída (ZHAO; GONG, 2015).

O processo de análise dinâmica, geralmente, é usado em ferramentas de testes instaladas, localmente ou *online,* que simulam o comportamento dos *hackers*, em um grande número de vetores de ataques. Muitos questionam sua confiabilidade e eficiência, pois na grande maioria das ferramentas, os testes são realizados internamente, apresentam-se somente os resultados, não dando visibilidade de quais métodos foram utilizados nos testes (MONGA; PALEARI; PASSERINI, 2009).

Existem diversas ferramentas comerciais e versões gratuitas para otimizar o processo de análise de vulnerabilidades, aplicadas para linguagens de desenvolvimento WEB, dentre elas destacam-se: phpBB (PHPBB GROUP, 2018), Ardilla (KEIZUN et al., 2018), Metasploit (LLC, 2018), Nessus Vulnerability Scanner (TENABLE, 2018), Burp Suite (PORTSWIGGER, 2018), OWASP Zed Attack Proxy (OWASP, 2018), SQL Map (GUIMARAES, 2018), Kali Linux (OFFENSIVE SECURITY, 2018).

Uma etapa importante é definir corretamente a ferramenta a ser adotada para a fase de análise de vulnerabilidades, ou seja:

> ... existem hoje no mercado dezenas de ferramentas. As grandes problemáticas envolvendo o uso dessas ferramentas são a forma de acesso, licenciamento e quantidade de funcionalidades presentes nas aplicações (ASSUNÇÃO, 2015).

As ferramentas gratuitas, em sua grande maioria, não realizam uma análise completa das vulnerabilidades, apresentam muitos falsos positivos, que são vulnerabilidades apresentadas, porém quando analisadas manualmente não são confirmadas, e falsos negativos, que são vulnerabilidades existentes, mas não são apontadas pela ferramenta (ASSUNÇÃO, 2008). Nota-se que, algumas ferramentas, não possuem uma *interface* amigável e pouco detalhamento dos resultados apresentados. Porém, se utilizada por especialistas de forma conjunta, podem trazer um melhor resultado. Desde 2010, existe o projeto *Web Application Vulnerability Scanner Evaluation Project* (WAVSEP), avaliando as ferramentas de análise de vulnerabilidade, gratuitas e comerciais, quanto aos recursos e a qualidade, aplicando casos de testes de forma a se aferir a precisão dessas ferramentas (CHEN, 2018).

Dentre as ferramentas de análise de vulnerabilidades (*webscanners*), neste trabalho, optou-se pelo uso do *software* NetSparker 4.9 Desktop, na versão comercial (NETSPARKER, 2018), conforme testes apresentados por (QASAIMEH; SHAMLAWI; KHAIRALLAH, 2018), o Netspaker é uma das ferramentas que apresenta uma precisão de detecção de mais de 90%, reavalia automaticamente a vulnerabilidade

encontrada, apresenta baixas quantidades de falsos positivos e falsos negativos, resultados também evidenciados nos testes por (JNENA, 2013).

3.5 COBIT 5

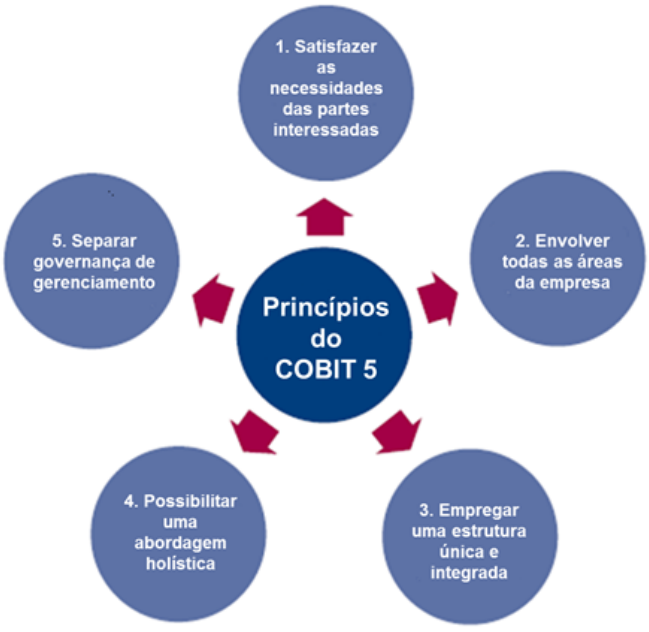
O modelo de governança COBIT 5 é aplicável ao gerenciamento da área de tecnologia e comunicação, desde o nível estratégico, tático, até o operacional, apresentando um padrão que é aplicado por inúmeras organizações, de diferentes segmentos e tamanhos, podendo ser implementado parcial, total ou em conjunto com outras metodologias e ferramentas de forma integrada (ISACA, 2012).

No setor público existem iniciativas, ainda modestas se comparadas com empresas privadas, para a definição e implementação de ferramentas de governança, isto deve-se à falta de recursos financeiros, profissionais qualificados, auditorias internas e externas especializadas em *complaince*, termo em inglês usado para definir se as políticas, processos e procedimentos estão de acordo com a metodologia ou ferramenta adotada (SOUZA, 2013) (BRASIL, TRIBUNAL DE CONTAS DO ESTADO DO RIO DE JANEIRO, 2013)

O COBIT 5 permite a avaliação do nível de maturidade dos processos de TIC, visando à melhoria contínua e garantindo um ambiente seguro, robusto e eficiente, dessa forma, ajudando a atingir os objetivos estratégicos e atender as partes interessadas (*stakeholders*).

A Figura 11 mostra os cinco princípios do COBIT 5.

Figura 11 – Princípios do COBIT 5



Fonte: ISACA (2012)

Na aplicação dos princípios do COBIT 5, também referencia-se ao modelo de negócio aplicado pelo BSC (*Balanced Scorecard*), metodologia desenvolvida em 1992 pelos professores Robert Kaplan e David Norton da *Harvard Business School* e que é utilizada mundialmente por diversos segmentos de negócio. O BSC tem como componentes, a definição do mapa estratégico, objetivos estratégicos, indicadores, metas e planos de ação. A estratégia é decomposta de uma maneira lógica, baseando-se em relações de causa e efeito, vetores de desempenho e de fatores financeiros. Os objetivos, metas, indicadores e iniciativas são divididos em quatro dimensões: financeira, cliente, interna, treinamento e crescimento conforme, mostra a Figura 12 (BALANCED SCORECARD INSTITUTE, 2018).

Figura 12 – Dimensões do Balanced Scorecard

Fonte: (SITEWARE, 2018)

Ressalta-se o trabalho desenvolvido por (HUANG; LEE; KAO, 2006), que consiste em um modelo em BSC nas quatro dimensões: financeira, cliente, interna, treinamento e crescimento para mensurar o nível de atendimento dos objetivos corporativos, de maneira a, atender todos as partes interessadas do negócio, ajudando a organização a entregar mais valor e atingir as estratégias do plano de negócio.

A Tabela 2 apresenta como os processos do COBIT 5 atendem os objetivos corporativos, a realização de benefícios, a otimização de riscos e recursos, agrupados nas dimensões do BSC e classificados por atendimento **Primário** ou **Secundário**.

Desta forma, para cada objetivo há uma indicação para cada uma das perguntas descritas na Tabela 2, por exemplo: “1 - Valor dos investimentos percebidos pelas partes interessadas (*Stakeholders*)”, atende principalmente à necessidade dos stakeholders na “Realização de Benefícios”, mas também atende à “Otimização dos Recursos”, só

que de forma secundária ou adicional.

Tabela 2 – Objetivos do COBIT 5, os objetivos corporativos e as dimensões do BSC

BSC	Objetivos Corporativos	Objetivos de Governança		
		Realização de Benefícios	Otimização de Riscos	Otimização de Recursos
Financeira	1 - Valor dos investimentos percebido pelas Partes Interessadas (*Stakeholders*).	Primário		Secundário
	2 - Portfólio de produtos e serviços competitivos.	Primário	Primário	Primário
	3 - Gestão do Risco do Negócio (Preservação dos ativos).		Primário	Secundário
	4 - Conformidade com leis vigentes e regulamentações externas.		Primário	
	5 - Transparência financeira.	Primário	Secundário	Secundário
Cliente	6 - Cultura de serviços orientado ao cliente.	Primário		Secundário
	7 - Continuidade e Disponibilidade dos serviços.		Primário	
	8 - Resposta rápida para ambiente de negócio em mudança.	Primário		Secundário
	9 - Tomada de decisão estratégica com base em informação.	Primário	Primário	Primário
	10 - Otimização dos custos de prestação de serviço.	Primário		Primário
Interna	11 - Otimização da funcionalidade dos processos de negócio.	Primário		Primário
	12 - Otimização dos custos dos processos de negócio.	Primário		Primário

	13 - Gestão de programas de mudança de negócios.	Primário		Secundário
	14 - Produtividade operacional e da equipe.	Primário		Primário
	15 - Conformidade com as políticas internas.		Primário	
Treinamento	16 - Pessoas qualificadas e motivadas.	Secundário	Primário	Primário
	17 - Cultura de inovação.	Primário		

Fonte: ISACA - adaptado de Figueredos (2014)

A Tabela 3 relaciona o atendimento dos objetivos de Tecnologia da Informação e Comunicação nas quatro dimensões do BSC (*Balanced Scored Card*)

Tabela 3 – Objetivos de TIC nas dimensões do BSC

BSC	Objetivos de Tecnologia da Informação e Comunicação
Financeira	1 - Alinhamento com as estratégias do negócio.
	2 - Conformidade com as leis e regulamentos externos.
	3 - Compromisso da gerência executiva com a tomada de decisão de TIC.
	4 - Gestão de risco organizacional de TIC.
	5 - Benefício pelo portfólio e pelos investimentos de TIC.
	6 - Transparência dos custos, benefícios e riscos de TIC.
Cliente	7 - Prestação de serviços de TIC em consonância com os requisitos de negócio.
	8 - Uso adequado dos recursos de TIC, informações e soluções tecnológicas.
Interna	9 - Agilidade nos projetos e serviços.
	10 - Segurança da informação, infraestrutura e aplicativos.
	11 - Otimização de ativos e recursos de TIC.

BSC	Objetivos de Tecnologia da Informação e Comunicação
	12 - Integração de aplicativos e tecnologias em apoio e capacitação aos processos de negócio.
	13 - Entrega de soluções, atendendo os requisitos de negócio, dentro dos prazos e custos estabelecidos.
	14 - Disponibilidade de informações relevantes e confiáveis para a tomada de decisão.
	15 - Conformidade de TIC com as políticas internas.
Treinamento	16 - Equipes de TIC qualificadas e motivadas.
	17 - Conhecimento, expertise e iniciativas para inovação dos negócios.

Fonte: ISACA - adaptado de Figueredos (2014)

O COBIT é fundamentado em 5 princípios, que são: satisfazer as expectativas das partes interessadas (*stakeholders*), separar governança de gestão, habilitar uma visão holística, ser uma ferramenta integradora e cobrir o negócio na totalidade, dando direcionamento estratégico, tático e operacional em TIC.

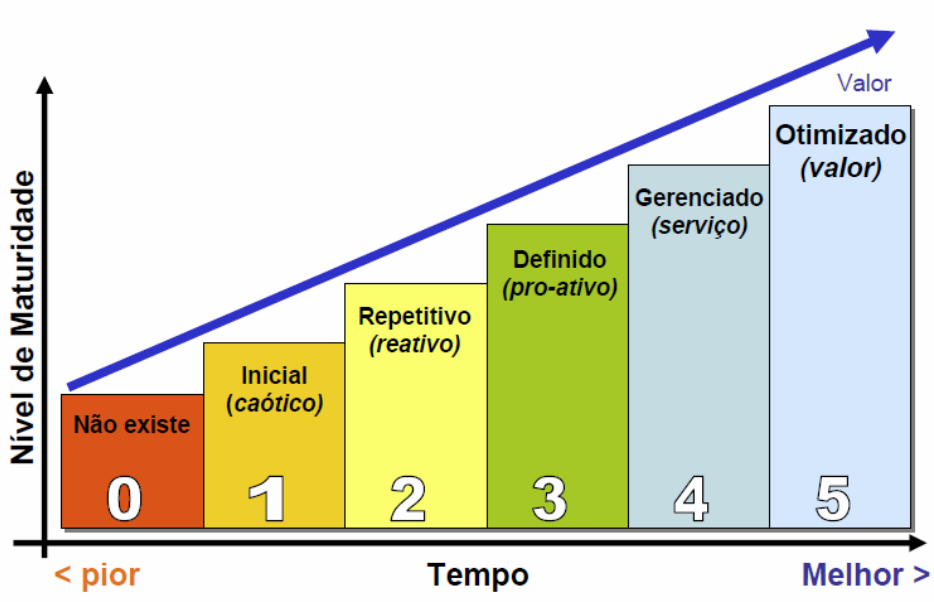
Para aferir e avaliar o nível de maturidade dos processos são utilizados 6 níveis conforme a tabela 4 e também pela Figura 13.

Tabela 4 – Nível de maturidade do processo - Modelo COBIT 5

0 - Inexistente	Gerenciamento de processos não é aplicado.
1 - Inicial	Processo é informal e desorganizado.
2 - Repetitivo	Processo é intuitivo e segue um padrão.
3 - Definido	Processo é formal, documentado, comunicado e aplicado.
4 - Gerenciado	Processo é monitorado e medido.
5 - Otimizado	Melhores práticas são seguidas, o processo é automatizado e é aplicado ciclo de melhora continua.

Fonte: ISACA - COBIT 5 (2012) - adaptado pelo autor

Figura 13 – Nível de Maturidade COBIT



Fonte: Santos (2006)

Conforme a ISACA (2012) a implementação do modelo de governança COBIT 5 traz inúmeras vantagens para a organização, dentre elas, podem-se destacar:

- Fornece um modelo integrado e abrangente que permite dar visibilidade e otimizar o valor gerado pela TIC;
- Permite uma gerência e governança de TIC de forma holística pela alta direção;
- Estabelece uma linguagem comum entre a TIC e o negócio;
- Alinha as iniciativas e projetos de TIC ao plano estratégico do negócio;
- Contribui na construção de um ambiente robusto, reduzindo os riscos de TIC, a níveis aceitáveis, para a sustentação do atual e futuros negócios.

Na tabela 5 são apresentados os trinta e sete processos do modelo COBIT 5:

Tabela 5 – Processos - Modelo COBIT 5

Avaliar, Dirigir e Monitorar - EDM (*Evaluate, Direct and Monitor*)

EDM01	Garantir a Definição e Manutenção do Modelo de Governança
EDM02	Garantir a Realização de Benefícios
EDM03	Garantir a Otimização do Risco
EDM04	Garantir a Otimização de Recursos
EDM05	Garantir a Transparência às Partes Interessadas
Alinhar, Planejar e Organizar - APO (*Align, Plan and Organise*)	
APO01	Gerenciar a Estrutura de Gestão de TIC
APO02	Gerenciar a Estratégia
APO03	Gerenciar Arquitetura da Organização
APO04	Gerenciar Inovação
APO05	Gerenciar Portfólio
APO06	Gerenciar Orçamento e Custos
APO07	Gerenciar Recursos Humanos
APO08	Gerenciar Relacionamentos
APO09	Gerenciar Contratos de Prestação de Serviços
APO10	Gerenciar Fornecedores
APO11	Gerenciar Qualidade
APO12	Gerenciar Riscos
APO13	Gerenciar Segurança
Construir, Adquirir e Implementar - BAI (*Build, Acquire and Implement*)	
BAI01	Gerenciar Programas e Projetos
BAI02	Gerenciar Definição de Requisitos
BAI03	Gerenciar Identificação e Desenvolvimento de Soluções
BAI04	Gerenciar Disponibilidade e Capacidade

BAI05	Gerenciar Capacidade de Mudança Organizacional
BAI06	Gerenciar Mudanças
BAI07	Gerenciar Aceitação e Transição da Mudança
BAI08	Gerencia Conhecimento
BAI09	Gerenciar Ativos
BAI10	Gerenciar Configuração
Entregar, Atender e Apoiar - DSS (*Deliver, Service and Support*)	
DSS01	Gerenciar Operações
DSS02	Gerenciar Solicitações e Incidentes de Serviços
DSS03	Gerenciar Problemas
DSS04	Gerenciar Continuidade
DSS05	Gerenciar Serviços de Segurança
DSS06	Gerenciar Controles do Processo de Negócio.
Monitorar, Avaliar e Analisar - MEA (*Monitor, Evaluate and Assess*)	
MEA01	Monitorar, Avaliar e Analisar Desempenho e Conformidade.
MEA02	Monitorar, Avaliar e Analisar o Sistema de Controle Interno.
MEA03	Monitorar, Avaliar e Analisar Conformidade com Requisitos Externos

Fonte: ISACA - adaptado de Figueredos (2014)

Cada processo possui seu objetivo de controle, como, por exemplo, o processo DSS05 - Gerenciar Serviços de Segurança: Proteger a informação da prefeitura para manter um nível aceitável de risco de segurança da informação, em linha com a política de segurança. Estabelecer e manter papéis e privilégios de acesso para segurança da informação e realizar monitoramento de segurança.

O processo, desse modo, é definido, quanto aos requisitos do negócio a serem atendidos, seu foco, como são atingidos e medidos, e contendo objetivos controle e divididos em processos específicos, conforme se mostra na Tabela 6.

Tabela 6 – Processo DSS05 - Gerenciar Serviços de Segurança

DSS05.01	Proteger contra *software* malicioso.
DSS05.02	Gerenciar a segurança das conexões de rede e conectividade.
DSS05.03	Gerenciar a segurança dos *endpoints*.
DSS05.04	Gerenciar identidade de usuários e acessos lógicos.
DSS05.05	Gerenciar acesso físicos aos ativos de TIC.
DSS05.06	Gerenciar documentos sensíveis e dispositivos de saída.
DSS05.07	Monitorar a infraestrutura por eventos de segurança.

Fonte: ISACA - COBIT 5 (2012)

Para cada processo específico, como, por exemplo, o processo DSS05.05 – Gerenciar acesso físicos aos ativos de TIC, aplicado a Infraestrutura, define-se a importância de possuir um ambiente de TIC controlado visando a proteção dos equipamentos computacionais e das pessoas, objetivando reduzir as interrupções do negócio por danos a equipamentos ou às pessoas, conforme mostrado na Tabela 7.

Tabela 7 – Processo DSS05.05 - Gerenciar acesso físicos aos ativos de TIC

Objetivo	Descrição
Requisito do Negócio	Proteger ativos de TIC e dados de forma a minimizar o risco de segurança e interrupções no negócio.
Foco	Prover e manter um ambiente físico apropriado para proteger ativos de TIC de acessos, danos ou furtos.
Como Atingir	Implementar medidas de segurança física.
	Gerenciar as instalações.
Como Medir	Tempo de parada advinda de incidentes no ambiente físico.
	Número de incidentes de quebra ou falha da segurança física.
	Frequência de levantamento e revisões de riscos físicos.

Fonte: ISACA (2012) - adaptado pelo autor

Os objetivos de controle detalhados, definidos neste processo, são apresentados na Tabela 8.

Tabela 8 – Processo DSS05.05 - Gerenciar acesso físicos aos ativos de TIC - Detalhado.

#	Controle	Atividades detalhadas
01	Controle de Solicitação e Concessão de Acessos	Gerenciar as solicitações e concessões de acesso aos recursos de computação. Pedidos formais de acesso devem ser preenchidos, e autorizados pela gestão do ambiente TIC, e os registros mantidos. Os formulários devem identificar especificamente as áreas em que o indivíduo é concedido acesso.
02	Revisão Periódica de Acessos	Garantir que os perfis de acesso continuem atualizados. Associe o acesso às instalações de TIC (salas de servidores, edifícios, áreas ou zonas) a funções de trabalho e responsabilidades.
03	Controle de Identificação	Instruir todos os funcionários a apresentar uma identificação visível em todos os momentos. Impedir a emissão de identificações ou crachás sem a devida autorização.
04	Segurança com visitantes	Exigir que visitantes sejam acompanhados todo o tempo, enquanto no local. Se um indivíduo desacompanhado, desconhecido ou sem identificação pessoal é identificado, alertar o pessoal de segurança.
05	Segurança Física Perimetral e de Acessos	Restringir o acesso a ambiente TIC sensíveis, estabelecendo restrições de perímetro, tais como cercas, paredes e dispositivos de segurança nas portas interiores e exteriores. Certifique-se de que os dispositivos registrem a entrada e disparem um alarme em caso de acesso não autorizado. Exemplos de tais dispositivos incluem crachás ou cartões-chave, teclados para senha, circuito fechado de televisão e *scanners* biométricos.

#	Controle	Atividades detalhadas
06	Treinamentos Regulares	Conduzir treinamentos de conscientização sobre segurança física regularmente.

Fonte: ISACA (2012) - adaptado pelo autor

4 METODOLOGIA

O trabalho apresentado tem objetivo qualitativo e utiliza os métodos bibliográfico e experimental.

Define-se pesquisa como um processo de investigação que busca descobrir relações existentes entre os aspectos que envolvem os fatos, fenômenos, situações ou coisas, resultando em maior conhecimento e busca de soluções. De acordo com (MARCONI; LAKATOS, 2003), p.221, “metodologia de pesquisa é aquela que abrange o maior número de itens, pois responde, a um só tempo, às questões: Como? Com quê? Onde? Quanto?”

O trabalho aplicou uma abordagem de pesquisa qualitativa, método de investigação científica que foca no caráter subjetivo, não faz uso de fórmulas matemáticas ou estatísticas, como aplicado na pesquisa quantitativa, mas lança-se dos resultados obtidos, para discutir e sugerir propostas na solução de um problema. De acordo com (RICHARDSON, 1999), a principal diferença entre uma abordagem qualitativa e quantitativa, explica-se no fato de que esta abordagem não faz uso de instrumento estatístico, como base para a análise de um problema, desta forma não objetivando numerar ou medir unidades ou categorias homogêneas.

De acordo com (MARCONI; LAKATOS, 2001), um método é composto por um conjunto de atividades possibilitando o atingimento de um objetivo, podendo este ser por meio da explicação de um fato através de teorias e hipóteses, ou mesmo, solucionando certo problema.

Para a elaboração da fundamentação teórica do trabalho, utilizou-se do método de pesquisa bibliográfica em livros, publicados por especialistas da área e artigos científicos, presentes em periódicos Qualis da CAPES, conforme apresentado no Capítulo 3, item 2. Para (FONSECA, 2002), “a pesquisa bibliográfica é feita a partir do levantamento de referências teóricas já analisadas, e publicadas por meios escritos e eletrônicos”.

E por fim, o método de pesquisa experimental, que consiste em estudos exploratórios com propósito de demonstrar a viabilidade do uso da metodologia proposta.

4.1 Cenário da Pesquisa

Conforme abordado no Capítulo 3, item 2, com a implantação de novos serviços a população em cidades inteligentes, eleva-se a necessidade do uso e operação de sistemas e ambiente TIC mais seguro, portanto as prefeituras precisam aplicar mais recursos financeiros e tecnológicos, adotar e implantar metodologias e possuir o pessoal adequado para as tarefas de desenvolvimento e de suporte de seu ambiente e

sistemas.

Por outro lado, observa-se a dificuldade de recursos financeiros e equipes treinadas e adequados, principalmente, enfrentado por pequenas e médias prefeituras (SOUZA, 2013).

Verifica-se a pouca adoção de metodologias aplicadas à governança e gestão de TIC na administração pública, desta forma, não existindo uma avaliação do nível de maturidade de seus processos e exposição a riscos (BRASIL, TRIBUNAL DE CONTAS DO ESTADO DO RIO DE JANEIRO, 2013).

Como abordado no Capítulo 3, item 3.2, a administração pública, prefeituras, órgãos públicos e universidades, adotaram o uso de *software* livre como sistemas operacionais, sistemas de servidores e de bancos de dados. Trazendo ganhos financeiros devido ao não pagamento de licenciamentos desses *softwares* (PINA, 08/05/2014) e reduzindo o processo de licitação estabelecidos na Lei no 8.666/1993 - Licitações e Contratos (BRASIL - SENADO FEDERAL, 2017).

Com isto, se viabilizou o desenvolvimento próprio ou em parceria público-privada, de sistemas WEB para atender diversas demandas dos departamentos ou processos e aplicações específicas para a administração pública. Algumas, dessas iniciativas foram abordadas no capítulo 3, item 3.2, e serão avaliadas neste trabalho.

Nos últimos anos, evidencia-se o crescimento de ataques de de invasores mal intencionados ou c*rackers*[1], e sua sofisticação, cada vez maior, nos métodos e nas ferramentas utilizadas, explorando vulnerabilidades novas e conhecidas em sistemas WEB, conforme apresentado no capítulo 3, item 3.3, tornando as pequenas e médias prefeituras, geralmente sem recursos financeiros e ambientes seguros, como vítimas de sucessos nestes ataques.

Para sistemas desenvolvidos internamente ou customizados, deve-se aplicar, segundo recomendação da OWASP, o uso da metodologia de desenvolvimento de *softwares* seguros (SDLC - *Security Development Lifecycle)*, metodologia de gerenciamento do ambiente TIC e contratação de pessoal ou prestadores de serviço adequadamente capacitados, o que não são evidenciados em pequenas e médias prefeituras, resultando em sistemas vulneráveis e expondo a riscos os processos e dados sensíveis.

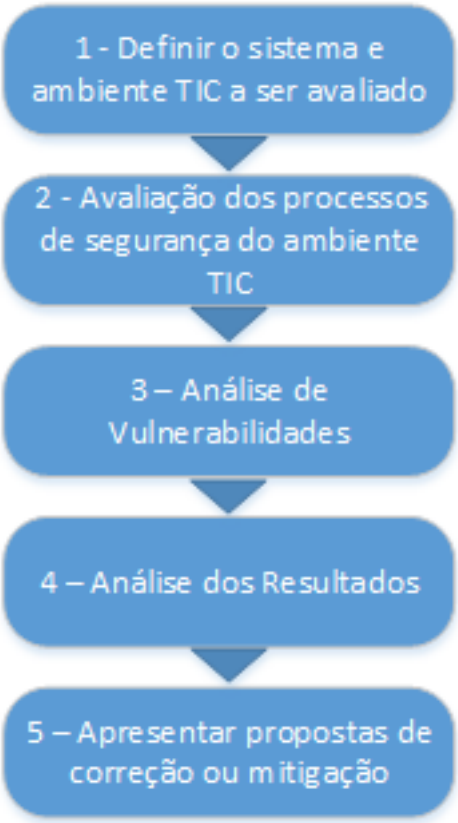
4.2 Método de Trabalho

A proposta metodológica proposta por este trabalho consiste, primeiramente, em definir e entender o sistema WEB e o ambiente TIC, avaliando os processos de segurança do ambiente, aferindo o nível de maturidade dos processos de segurança

[1] *Crackes*: São incividuos aficionadas por tecnologia da informação que utilizam seu grande conhecimento na área para quebrar códigos de segurança.

existentes, conforme apresentados na Tabela 9, executar a análise de vulnerabilidades, analisar os resultados e apresentar recomendações para correção e mitigação dos riscos encontrados, apresentados na Figura 14.

Figura 14 – Proposta para avaliação de segurança do sistema e ambiente TIC



Fonte: Autor (2018)

4.2.1 Sistemas Avaliados

O delineamento da pesquisa tem caráter experimental, sendo avaliados três sistemas WEB (e-Cidade, e-SIC e SIVAC), desenvolvidos em parceria público-privada, são inciativas do Ministério de Planejamento, Desenvolvimento e Gestão e distribuídos gratuitamente às prefeituras e órgãos públicos através do Portal Software Livre Brasileiro.

Os três sistemas foram escolhidos, seguindo os critérios a seguir:

- Sistemas desenvolvidos para ambiente WEB;
- Distribuídos na modalidade de *software* livre;
- Com o objetivo de prestação de serviços à população, prefeituras e órgãos públicos;

- Instalados em ambiente de *software* livre, sistema operacional, servidor da aplicação e banco de dados;
- Desenvolvidos em linguagem PHP (*Hypertext Preprocessor*);
- Desenvolvidos por equipes e empresas diferentes;
- Dentro da área de concentração de Cidades Inteligentes.

4.2.2 Processo de Análise de Vulnerabilidades

Neste trabalho, a adoção da ferramenta NetSpaker teve como base os experimentos realizados por (QASAIMEH; SHAMLAWI; KHAIRALLAH, 2018), (JNENA, 2013) e (BELARMINO, 2014), onde os testes realizados apresentaram um nível de acerto superior á 90%, com baixa incidência de falsos positivos e falsos negativos.

Com a aplicação da análise de vulnerabilidade, busca-se identificar as vulnerabilidades geradas na fase de desenvolvimento do sistema e no ambiente TIC com foco nos sistemas operativos, serviços executados e o bancos de dados que estão sendo executados. Optou-se por realizar uma análise estática no código-fonte, verificando se a vulnerabilidade realmente existe. Se positivo buscou-se corrigí-la ou remediá-la. Do mesmo modo, se aplica ao ambiente TIC, pelo resultado apresentado, podendo constar vulnerabilidade de segurança, em sistemas operacionais ou em *softwares* instalados com base em versões desatualizadas e protocolos não seguros. Da mesma forma, foi realizado uma análise dos resultados apresentados, buscando corrigir as vulnerabilidades, aplicando as atualizações de segurança e uso de protocolos seguros. Ressalta-se que em atualizações de versões do sistema operacional ou dos *softwares* específicos e que suportam o sistema, realizou-se um estudo para identificar se não ocorrerá impactos funcionais ao sistema, aplicando as atualizações em ambientes de testes diferentes do ambiente de produção.

Para o processo de análise de vulnerabilidades dos sistemas foi utilizado o *software* NETSPARKER 4.9 Desktop, pelas vantagens apresentadas e por sua boa avaliação entre outras ferramentas de *webscanner*. As vulnerabilidades encontradas foram apresentadas conforme classificação OWASP (OWASP, 2018). Desta forma, facilitando a busca por soluções das vulnerabilidades conhecidas.

4.2.3 Análise do Ambiente de Tecnologia da Informação e Comunicação

Foi feita a avaliação do ambiente TIC, validando os processos e sub-processos, considerando os processos do COBIT 5 descritos na Tabela 9. Esta avaliação não se volta para todos os processos do ambiente TIC, apenas serão considerados os

processos que dão sustentação à operação e manutenção do sistema a ser avaliado. Desta forma, avaliou-se uma metodologia simplificada, onde são considerados, além da análise de vulnerabilidades citado no Capítulo 3, item 3.4, sete processos do total de trinta e sete processos existentes no modelo COBIT 5, que são diretamente ligados à segurança, gerenciamento de riscos e controle de mudanças.

Para o levantamento do ambiente computacional, desenvolveu-se um *check-list* que pode ser aplicado, no intuito de auxiliar no entendimento do ambiente TIC, dos processos e do sistema WEB avaliado, podendo ser utilizado totalmente ou em parte, na fase de entrevistas e visitas. O material completo encontra-se no Anexo A.

Com base no levantamento do ambiente e processos existentes, foi classificado no nível de maturidade de cada processo e sub-processo de segurança, de acordo com classificação definida no modelo, sendo: 0 - Inexistente , 1 - Inicial , 2 - Repetitivo , 3 - Definido, 4 - Gerenciado e 5 - Otimizado conforme definidos na tabela 4. Os processos de avaliação do ambiente WEB são propostos e descritos na Tabela 9.

Tabela 9 – Processos Propostos para a Avaliação do Ambiente WEB - com base no modelo COBIT 5

Avaliar, Dirigir e Monitorar - EDM (*Evaluate, Direct and Monitor*)		
EDM03 Assegurar a Otimização de Riscos	Certificar-se que o risco ao valor agregado da organização relacionado com o uso da TIC, é identificado, comunicado e controlado.	EDM03.01 - Avaliar a gestão de risco.
		EDM03.02 - Direcionar a gestão de risco.
		EDM03.03 - Monitorar a gestão dos riscos.
Alinhar, Planejar e Organizar - APO (*Align, Plan and Organise*)		
APO12 Gerenciar os Riscos	Identificar, avaliar e reduzir continuamente o risco relacionado a TIC dentro dos níveis de tolerância definidos pela gestão executiva da organização.	APO12.01 - Coletar dados.
		APO12.02 - Analisar risco.
		APO12.03 - Manter um perfil de risco.
		APO12.04 - Articular risco.

		APO12.05 - Definir um portfolio de ações de gerenciamento de risco.
		APO12.06 - Responder ao risco.
APO13 Gerenciar a Segurança	Definir, operar e monitorar um sistema para gerenciamento de segurança da informação.	APO13.01 - Estabelecer e manter um Sistema de Gerenciamento de Seguança da Informação (ISMS).
		APO13.02 - Definir e gerenciar um plano de tratamento de risco de segurança da informação.
		APO13.03 - Monitorar e revisar o ISMS
Construir, Adquirir e Implementar - BAI (*Build, Acquire and Implement*)		
BAI06 Gerenciar Mudanças	Gerencia todas as mudanças de uma maneira controlada, incluindo mudanças de padrão e de manutenção de emergência relacionada com os processos de negócio, aplicações e infraestrutura.	BAI06.01 - Avaliar, Priorizar e Autorizar as requisições de mudança.
		BAI06.02 - Gerenciar mudanças emergenciais.
		BAI06.03 - Acompanhar e reportar o status.
		BAI06.04 - Fechar e documentar mudanças.
BAI07 - Gerenciar Aceite e Transição de Mudança	Aceitar formalmente e fazer novas soluções operacionais, incluindo planejamento da implementação, sistema e conversão de dados, testes de aceitação, comunicação, preparação para liberação, promoção para a produção de processos de negócio novos ou alterados e serviços de TIC, suporte inicial de produção e a revisão pós implementação.	BAI 07.01 - Estabelecer um plano de implementação
		BAI07.02 - Planejar processos de negócios, sistema e conversão de dados.

		BAI07.03 - Planejar testes de aceitação.
		BAI07.04 - Estabelecer um ambiente de teste.
		BAI07.05 - Realizar testes de aceitação.
		BAI07.06 - Promover para a produção e gerir os lançamentos.
		BAI 07.07 - Fornecer suporte de produção inicial.
		BAI07.08 - Realizar uma revisão pós- implementação.
Entregar, Serviços e Suporte - DSS (*Deliver, Service and Support*)		
DSS04 - Gerenciar Continuidade	Estabelecer e manter um plano para permitir que o negócio e a TIC possam responder a incidentes e interrupções, a fim de continuar a operação de processos de negócios críticos e os serviços de TIC necessários, e manter a disponibilidade de informações em um nível aceitável para a prefeitura.	DSS04.01 - Definir a política de continuidade de negócios, seus objetivos e escopo.
		DSS04.02 - Manter uma estratégia de continuidade.
		DSS04.03 - Desenvolver e implementar uma resposta de continuidade de negócio.
		DSS04.04 - Exercitar, testar e revisar o Plano de Continuidade de Negócio (BCP-*Business Continuity Plan*).
		DSS04.05 - Revisar, manter e melhorar o plano de continuidade.

		DSS04.06 - Conduzir treinamento no plano de continuidade.
		DSS04.07 - Gerenciar mecanismos para *backup*.
		DSS04.08 - Conduzir revisões pós-retomada de operações.
DSS05 - Gerenciar Serviços de Segurança	Proteger a informação da prefeitura para manter um nível aceitável de risco de segurança da informação, em linha com a política de segurança, manter papéis e privilégios de acesso e monitorar.	DSS05.01 - Proteger contra software malicioso.
		DSS05.02 - Gerenciar a segurança das conexões de rede e conectividade.
		DSS05.03 - Gerenciar a segurança dos *endpoints*.
		DSS05.04 - Gerenciar identidade de usuários e acessos lógicos.
		DSS05.05 - Gerenciar acesso físicos aos ativos de TIC.
		DSS05.06 - Gerenciar documentos sensíveis e dispositivos de saída.
		DSS05.07 - Monitorar a infraestrutura por eventos de segurança.

Fonte: ISACA (2012) - adaptado pelo autor

Os processos propostos na Tabela 9 são prioritários e complementam o processo de análise de vulnerabilidades, de forma a garantir um sistema WEB seguro, sendo executados em um ambiente TIC seguro.

Os processos de Segurança da Informação foram classificados, conforme seu nível atual de maturidade, entre os valores de 0 - Inexistente e 5 - Otimizado, conforme a Tabela 10 .

Tabela 10 – Processos de Segurança Propostos - com base no COBIT 5

Processo	Descrição	Nível Atual	Nível Desejado
EDM03	Assegurar a Otimização de Riscos.	x	y
APO12	Gerenciar os Riscos	x	y
APO13	Gerenciar a Segurança	x	y
BAI06	Gerenciar Mudanças	x	y
BAI07	Gerenciar Aceite e Transição de Mudança	x	y
DSS04	Garantir Continuidade do Serviço	x	y
DSS05	Gerenciar Serviços de Segurança	x	y
Média do Nível Atual		m	

Fonte: ISACA (2012) - adaptado pelo autor

A variável **x** representa o Nível Atual avaliado durante os processos de avaliação, visitas, entrevistas e auditorias. A variável **y** representa o Nível Desejado, pela prefeitura, com base na criticidade da aplicação e seu plano estratégico, isto foi obtido através na criticidade da aplicação informada pelos gestores, risco exposto e sua dependência dos serviços TIC para a operação.

Para melhor eficiência, é necessário realizar a avaliação com os gestores de cada área ou departamento, envolvendo as pessoas chaves de cada processo, a área de desenvolvimento e de operação do ambiente TIC, o que não foi possível neste trabalho.

Para a avaliação do nível de maturidade dos processos, apresentados na Tabela 10, foram analisados seus processos e sub-processos como, por exemplo, o processo **DSS05 - Garantir a Segurança do Sistema**, com seus sub-processos, apresentados na Tabela 11.

Tabela 11 – Processo DSS05 - Garantir a Segurança do Sistema.

Processo	Descrição	Nível Atual	Nível Desejado

Processo	Descrição	Nível Atual	Nível Desejado
DSS05.01	Proteger contra *software* malicioso.	x	y
DSS05.02	Gerenciar a segurança das conexões de rede e conectividade.	x	y
DSS05.03	Gerenciar a segurança dos *endpoints*.	x	y
DSS05.04	Gerenciar identidade de usuários e acessos lógicos.	x	y
DSS05.05	Gerenciar acesso físicos aos ativos de TIC.	x	y
DSS05.06	Gerenciar documentos sensíveis e dispositivos de saída.	x	y
DSS05.07	Monitorar a infraestrutura por eventos de segurança.	x	y
Média do Nível Atual dos sub-processos		m	

Fonte: ISACA (2012) - adaptado pelo autor

Para cada sub-processo descrito, avaliou-se a execução de suas atividades de forma a aferir o nível de maturidade existente, como, por exemplo, no sub-processo **DSS05.01 - Proteger contra *software* malicioso,** descrito na Tabela 12.

DSS05.01 - Objetivo: Implementar e manter medidas preventivas, detentivas e corretivas (especialmente atualizações de pacotes de segurança e de antivírus) em toda a prefeitura para proteger os sistemas de informação e tecnologia de *software* malicioso (por exemplo, vírus, *worms*, *spyware*, *spam*).

Tabela 12 – DSS05.01 - Proteger contra *software* malicioso - Atividades - COBIT 5

Atividade	Descrição	Nível Atual
01	Realizar conscientização sobre *software* malicioso e aplicar procedimentos para prevenção e responsabilidades.	x
02	Instalar e ativar ferramentas de proteção contra *software* malicioso em todas as instalações de processamento, com arquivos de definição atualizados conforme exigido (de forma automática ou semi-automática).	x

Atividade	Descrição	Nível Atual
03	Distribuir todos os *softwares* de proteção a partir de um ponto central (versão e nível de patch) usando gerenciamento de configuração e mudanças centralizados.	x
04	Rever e avaliar novas potenciais ameaças regularmente (ex.: avaliando boletins de fabricantes e assessorias de segurança).	x
05	Filtrar táfego de entrada, como e-mail e *downloads*, para proteção contra informações não solicitadas (ex.: *spyware*, e-mails de *phishing*)	x
06	Conduzir periodicamente, treinamento sobre *malware* em e-mails e uso da *internet*. Treinar os usuários a não instalar *software* compartilhado ou não aprovado.	x
Média do Nível Atual das Atividades		m

Fonte: ISACA (2012) - adaptado pelo autor

4.2.4 Laboratório de Teste

Para a execução do processo de análise de vulnerabilidades foi construído um laboratório de testes, para cada sistema avaliado. Foi necessário realizar a virtualização de um servidor Linux nas distribuições Ubuntu e CentOs e instalando cada sistema seguindo as instruções e procedimentos existentes no procedimento e documento de instalação.

A máquina física utilizada foi um Ultrabook Dell E7440, Processador Intel I7, 16 GB memória RAM e HD SSD 256 GB, com sistema operacional Microsoft Windows 10 64 bits e foi utilizado o *software* de virtualização VMWare Desktop 12 Pro.

Para a reprodução dos testes, recomenda-se utilizar um equipamento com recursos semelhantes, com sistema operacional *Windows* e podendo utilizar outro *software* de virtualização, tais como: Virtual Box, Virtual PC, Hyper V e entre outros.

O *software* NetSpaker 4.9 Desktop somente esta disponível na versão *Windows*, desta forma, deve-se utilizar um equipamento com sistema operacional *Windows* para a execução do processo de análise de vulnerabilidades.

No laboratório construído foi utilizado o mesmo computador físico, descrito anteriormente, para a criação dos ambientes virtualizados necessários para a instalação de cada sistema avaliado.

O processo de teste deve ser realizado sistema a sistema, recomenda-se a utilização da opção de *scan incremental*. Os testes devem ser salvos para análise futura e podem ser executados repetidas vezes, em alguns dos testes realizados, estes

demoraram mais de vinte e quatro horas para serem totalmente concluídos.

Para realizar os testes deve-se: a) instalar o sistema operacional recomendado pelo sistema a ser testado, b) aplicar as últimas atualizações de software e segurança do sistema operativo, c) instalar os serviços e banco de dados recomendados, d) aplicar as rotinas de criação de tabelas e configurações, e) realizar o acesso ao sistema.

Após a execução do processo de instalação, deve-se executar o *software* NetSparker 4.9 Desktop a partir do endereço URL do sistema, desta forma, o sistema irá realizar o escaneamento de todas as páginas PHP do sistema WEB avaliado a partir da página principal.

Recomenda-se a instalação de cada sistema, seguindo rigorosamente o procedimento de instalação, nas versões recomendadas dos *softwares*, de maneira, a reproduzir fielmente a instalação como seria em um ambiente real.

5 RESULTADOS

Para a validação do método proposto, foi aplicada a análise de vulnerabilidade em três sistemas que são distribuídos pelo governo brasileiro e desenvolvidos em parceria pública privada. Os sistemas avaliados e versões foram: **e-CIDADES versão 2018-2**, **e-SIC Livre 1.4** e **SIVAC 2.1**. Para a análise foram consideradas as últimas versões disponibilizadas, identificando suas vulnerabilidades, seus riscos ou melhorias que devem ser consideradas para sua adoração e sua utilização em um ambiente seguro pelas prefeituras.

5.1 Análise de Vulnerabilidades

Para a execução do processo de análise das vulnerabilidades dos sistemas **e-CIDADE**, **e-SIC Livre** e **SIVAC** foi adotado o *software* NETSPARKET 4.9 Desktop, conforme mencionado anteriormente, realizando os testes de forma rápida, otimizada e sem a necessidade de análise de centenas de códigos-fontes e vulnerabilidades nas tecnologias e linguagens utilizadas por cada aplicação. A proposta consiste em validar o sistema sempre que houver alterações realizadas nos códigos-fontes, e também alterações realizadas no ambiente de TIC, tais como: atualização de *software* e segurança, mudanças ocorridas nos servidores de aplicação, banco de dados e sistemas de segurança (*firewalls*), conforme apresentado no Capítulo 5, item 5.1 com o uso da metodologia de **Ciclo de Vida do Desenvolvimento Seguro** (SDLC - *Security Development Lifecycle*). Este processo, se realizado de forma sistêmica e contínua, garantirá que todas as vulnerabilidades conhecidas sejam identificadas durante a fase de desenvolvimento e de testes, possibilitando a correção e resultando em sistemas WEB mais seguros e com baixo risco de ataques.

Nas seções seguintes serão apresentados a avaliação realizada nos três sistemas, descritos anteriormente, com base nos testes de vulnerabilidade realizados em ambiente de laboratório.

5.1.1 Sistema e-CIDADE 2018-2

Para a aplicação dos testes de vulnerabilidade no sistema e-CIDADE 2018-2 foi utilizado um ambiente em laboratório conforme detalhado na Tabela 13.

Tabela 13 – Ambiente Físico e Virtualizado utilizado para a avaliação do sistema E-CIDADE 2018-2

Ambiente	Descrição
Físico	Ultrabook Dell E7440, Processador Intel I7, 16 GB memória RAM e HD SSD 256 GB
	Microsoft Windows 10 64 Bits
	VMWare Desktop 12 Pro
	Netsparker Desktop Pro 4.9
	Firefox
Virtualizado	Máquina virtual com 1 processador, 1 GB memória RAM e HD 20 GB
	Linux Ubuntu 16.10.2 LTS Server
	Apache 2.4
	PostgreSQL 9.5
	PHP 5.6
	CVS
	Java

Fonte: Autor (2018)

5.1.1.1 Instalação do sistema

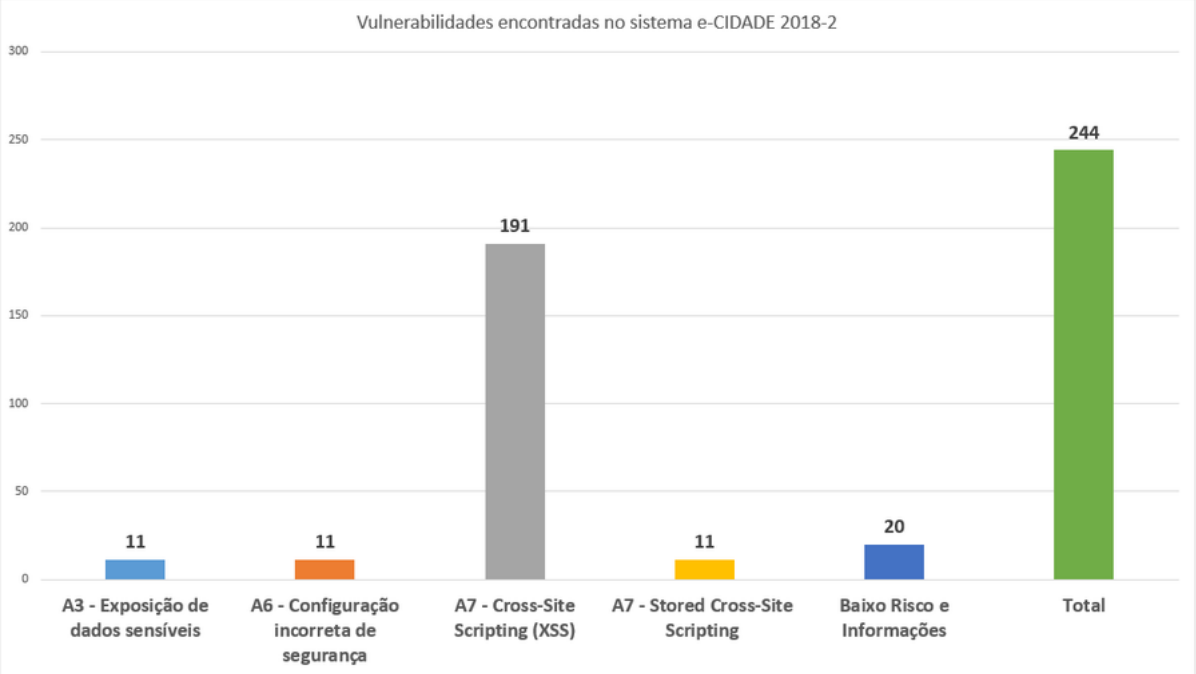
O sistema e ambiente foi instalado conforme as atividades a seguir:

- Transferência do sistema **e-CIDADE versão 2018-2** disponível por meio do endereço eletrônico: https://softwarepublico.gov.br/social/e-cidade;
- Instalação conforme a documentação técnica disponibilizada no arquivo de instalação;
- Após o processo de instalação, realizado na máquina virtual conforme ambiente especificado na Tabela 13, o acesso ao sistema é realizado através do *software* Firefox, ou outro *software* navegador de preferência do usuário, através da URL http://IP-Servidor/e-cidade;
- O usuário padrão definido na instalação é *dbseller* e a senha *dbseller.*

5.1.1.2 Resultado da análise de vulnerabilidade

O resultado do teste encontrou as vulnerabilidades apresentadas na Figura 15, classificadas com base no OWASP *Top Ten* 2017.

Figura 15 – Resultado - Sistema e-Cidade 2018-2



Fonte: NETSKAPER 4.9 - e-Cidade 2018-2 (2018)

5.1.2 Avaliação do sistema e-SIC Livre 1.4

Para a aplicação dos testes foi utilizado o ambiente em laboratório conforme destalhado na Tabela 14.

Tabela 14 – Ambiente Físico e Virtualizado utilizado para a avaliação do sistema e-SIC Livre 1.4

Ambiente	Descrição
Físico	Ultrabook Dell E7440, Processador Intel I7, 16GB memória RAM e HD SSD 256GB
	Microsoft Windows 10 64 Bits
	VMWare Desktop 12 Pro
	Netsparker Desktop Pro 4.9
	Firefox
Virtualizado	Máquina virtual com 4 processadores, 1 GB RAM e HD 60 GB
	Linux CentOS 7 64 Bits
	Apache 2.4.6
	MySql 5.5.60 MariaDB

Fonte: Autor (2018)

5.1.2.1 Instalação do sistema

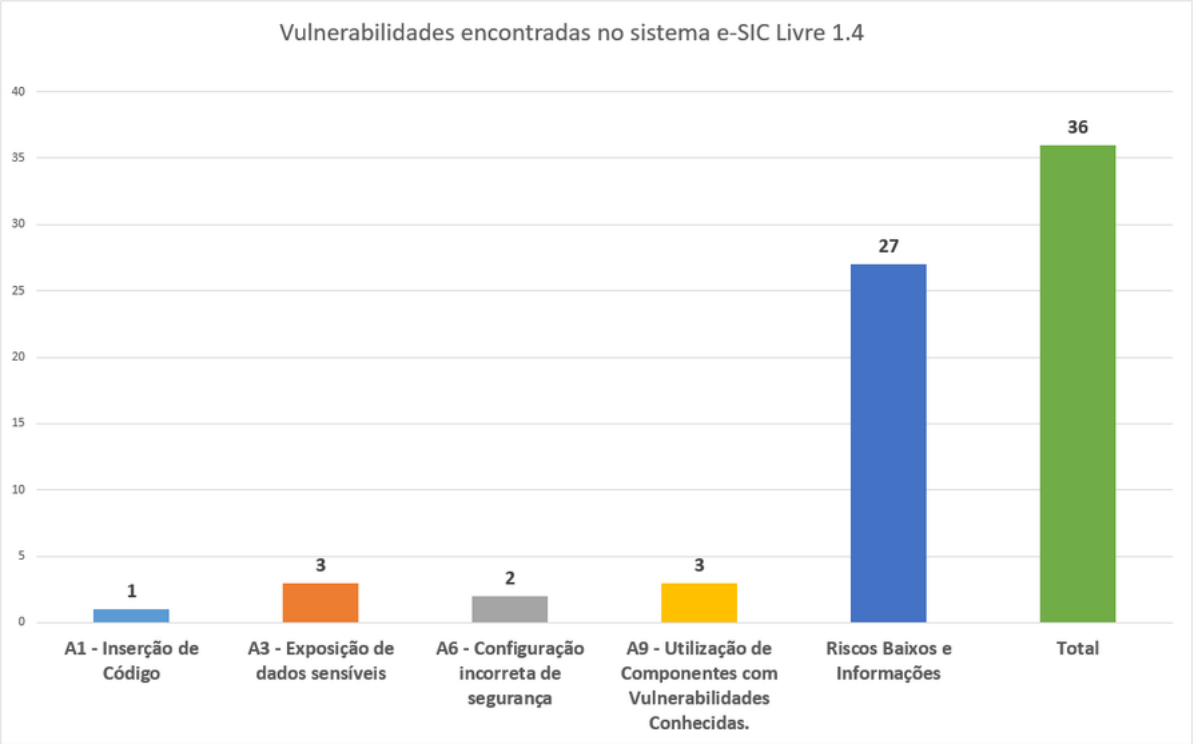
Para a instalação do sistema e do ambiente foram seguidas as seguintes atividades:

- Transferência do sistema **e-SIC Livre** versão 1.4 pelo endereço eletrônico: https ://softwarepublico.gov.br/social/e-sic-livre;
- Realizada a instalação conforme a documentação técnica disponibilizada no arquivo de instalação para o ambiente Linux;
- Após a instalação na máquina virtual o acesso ao sistema é realizado pelo *software* navegador Firefox, ou outro *software* navegador de preferência do usuário, pelo endereço: http://IP-Servidor/e-sic
- O usuário padrão definido no processo de instalação: usuário=*admin* e a senha=*admin.*

5.1.2.2 Resultado do teste de análise de vulnerabilidades

O resultado do teste encontrou vulnerabilidades conforme a Figura 16 e classificadas com base no OWASP *Top Ten* 2017.

Figura 16 – Resultado - Sistema e-SIC Livre 1.4



Fonte: NETSKAPER 4.9 - e-SIC Livre 1.4 (2018)

5.1.3 Avaliação do sistema SIVAC 2.1

Para a aplicação dos testes foi utilizado um ambiente em laboratório conforme tabela 13

5.1.3.1 Instalação do sistema

Instalado o sistema SIVAC 2.1 conforme descrito nas etapas abaixo:

- Transferência do sistema **SIVAC 2.1** por meio do endereço eletrônico https://softwarepublico.gov.br/social/sivac;
- Realizada a instalação conforme a documentação técnica disponibilizada no arquivo de instalação para ambiente Linux - Ubuntu;
- Após a instalação na máquina virtual o acesso ao sistema é realizado pelo *software* navegador Firefox, ou outro *software* navegador de preferência do usuário, pelo endereço http://IP-Servidor/ippes/Sistema;
- O usuário padrão criado na instalação: usuário=*ippes* e senha=*ippes.*

Tabela 15 – Ambiente Físico e Virtualizado utilizado para a avaliação do sistema SIVAC 2.1

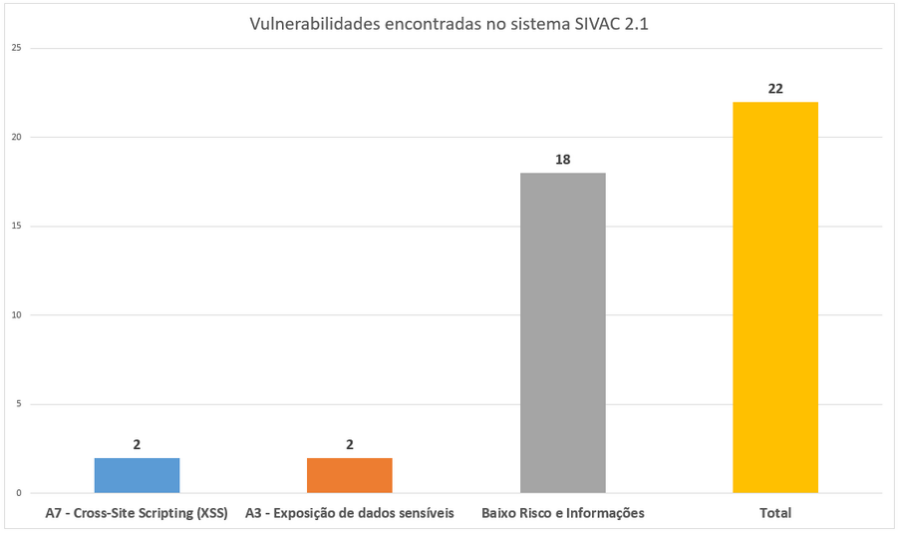
Ambiente	Descrição
Físico	Ultrabook Dell E7440, Processador Intel I7, 16 GB memória RAM e HD SSD 256 GB
	Microsoft Windows 10 64 Bits
	VMWare Desktop 12 Pro
	Netsparker Desktop Pro 4.9
	Firefox
Virtualizado	Máquina virtual com 1 processador, 1 GB memória RAM e HD 20 GB
	Linux Ubuntu 16.10.2 LTS Server
	Apache 2.4
	MySql 5.1
	PHP 5.6

Fonte: Autor (2018)

5.1.3.2 Resultado do teste de análise de vulnerabilidades

O resultado dos testes encontraram as vulnerabilidades conforme apresentados na Figura 17 e classificadas com base no OWASP *Top Ten* 2017.

Figura 17 – Resultado - Sistema SIVAC 2.1

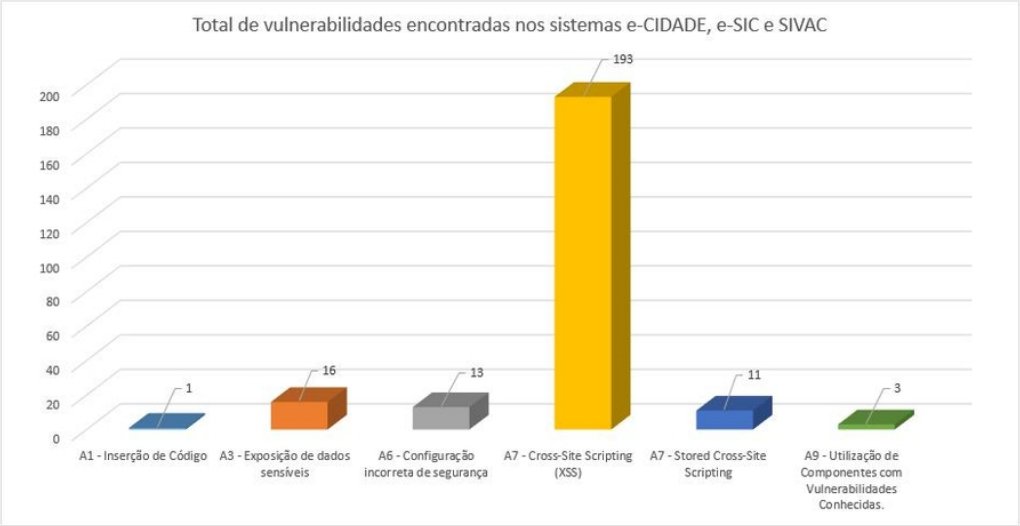


Fonte: NETSKAPER 4.9 - SIVAC 2.1 (2018)

5.2 Resultado dos sistemas avaliados

Na Figura 18 apresenta-se o total de vulnerabilidades de médio e alto risco encontradas nos sistemas.

Figura 18 – Total de Vulnerabilidades encontradas



Fonte: NETSKAPER 4.9 (2018)

6 DISCUSSÕES

6.1 Sistema e-CIDADE 2018-2

Na Tabela 16 apresentam-se as páginas (em linguagem PHP) e os diretórios classificados pelas vulnerabilidades de médio e alto risco. Com base nos resultados apresentados, possibilita-se analisar cada página e vulnerabilidade, buscando corrigi-las de forma a resolver ou mitigar o risco identificado.

Tabela 16 – Resultado das vulnerabilidades encontradas e confirmadas no sistema e-CIDADE

Vulnerabilidade	Classificação OWASP	Quantidade
Senha transmitida sobre protocolo HTTP (Confirmado)	A3 - Exposição de dados sensíveis	11
Cross-Site Scripting (Confirmado)	A7 - Cross-Site Scripting (XSS)	191
Stored Cross-site Scripting (Confirmado)	A7 - Cross-Site Scripting (XSS)	11
Auto Preenchimento Habilitado (Confirmado)	A6 - Configuração incorreta de segurança	11
Cookie não marcado apenas para acesso em HTTP (Confirmado)	A6 - Configuração incorreta de segurança	11

Fonte: NETSKAPER 4.9 - e-CIDADE versão 2018-2 (2018)

6.1.1 Vulnerabilidade: A3 - Exposição de dados sensíveis.

Na Tabela 17 apresentam-se as páginas (em linguagem PHP) com a vulnerabilidade classificada como A3 - Exposição de dados sensíveis.

Tabela 17 – A3 - Exposição de dados sensíveis - e-CIDADE

Vulnerabilidade	Página
Senha transmitida sobre protocolo HTTP	/e-cidade/login.php
	/e-cidade/w/1/login.php

Vulnerabilidade	Página
	/e-cidade/w/2/login.php
	/e-cidade/w/3/login.php
	/e-cidade/w/4/login.php
	/e-cidade/w/5/login.php

Fonte - NETSKAPER 4.9 - e-CIDADE versão 2018-2 (2018)

Para corrigir esta vulnerabilidade, recomenda-se habilitar o uso de HTTPS (Protocolo de transferência de hipertexto seguro), que basicamente consiste em uma implementação do protocolo HTTP (Protocolo de transferência de hipertexto) sobre uma camada adicional de segurança, utilizando os protocolos SSL(*Secure Sockets Layer*) / TLS(*Transport Layer Security*) permitindo a comunicação criptografada entre o sistema e o navegador do usuário.

6.1.2 Vulnerabilidade: A7 - *Cross-Site Scripting*

Na Tabela 18 apresentam-se as páginas em linguagem PHP com vulnerabilidades confirmadas e classificadas com A7 - Cross-Site Scripting (XSS).

Tabela 18 – A7 - Cross-Site Scripting (XSS) - e-CIDADE

Vulnerabilidade	Página
	abrir.php
	BuscaBase.RPC.php
	carregaCaptcha.php
	index.php
	login.php
	primeiroAcesso.php

Vulnerabilidade	Página
	cad3_conscadastro_002.php
	com3_conssolic002.php
	con1_help001.php
	con1_usuariosistema.RPC.php
	con3_versao001.php
	con4_dbhelp.RPC.php
	con4_dbtutorial.RPC.php
	con4_mensagems.RPC.php
	db_download.php
	db_listaarquivos.php
	func_agendamedica.php
	func_calendario.php
	func_consbens001.php
	func_empempenho001.php
	func_empempenhoaut001.php
	func_saldoorcdotacao.php
	func_saldoorcreceita.php
	instit.php
	iss3_consinscr003.php
	mat3_consultaatendrequi001.php
	mat3_consultadevolucao001.php
	mat3_consultarequi001.php
	mat3_matconsultaiframe003.php

Vulnerabilidade	Página
	pes2_aleemiteblvbanrisul001.php
	prot3_conscgm002.php
	estilos.php
	sys4_itensmenus.RPC.php
	img.php

Fonte - NETSKAPER 4.9 - e-CIDADE versão 2018-2 (2018)

Algumas das páginas anteriores se repetem em outros diretórios do sistema, como, por exemplo, as páginas **login.php** e **abrir.php** que estão nos diretórios w, w/1, w/2, w/3, w/4 e w/5. A correção da vulnerabilidade é a mesma, portanto foram apresentadas de forma sintética, recomenda-se a área de desenvolvimento avaliar as páginas, acima citadas, se são iguais ou diferentes.

A correção da vulnerabilidade *Cross-site Scripting* foi apresentada no capítulo 3, item 3.3 deste trabalho, como mencionado anteriormente é uma vulnerabilidade fortemente exploradas por *hackers* , de alto grau de risco e exposição de dados sensíveis, *phishing* (forma utilizada por *hackers* para enganar os usuários e obter dados pessoais tais como: CPF, números de contas bancárias, cartões de crédito e senhas) e sequestro da sessão do usuário e desta forma tendo acesso à aplicação e os dados.

A vulnerabilidade *Cross-site Scripting* deve ser corrigida rapidamente para evitar exposição dos riscos citados anteriormente.

6.1.3 Vulnerabilidade: A7 - *Stored Cross-site Scripting*

Apresenta-se o diretório com a classificação A7 - *Cross-Site Scripting (XSS)* onde confirmada a vulnerabilidade *Stored Cross-site Scripting*:

Tabela 19 – A7 - *Cross-Site Scripting*(XSS) -*Stored Cross-site scripting* - E-CIDADE

Vulnerabilidade	Página
Stored Cross-site Scripting	/e-cidade/skins/default/estilos/

Fonte - NETSKAPER 4.9 - E-CIDADE versão 2018-2 (2018)

Confirmada a vulnerabilidade através da introdução de códigos e execução de códigos em java-script, permite obter o acesso administrativo da aplicação, acesso á informações sensíveis e a possibilidade de alteração de dados dentro da própria aplicação.

6.1.4 Vulnerabilidade: A6 - Configuração incorreta de segurança

Outro risco evidenciado, porém de risco baixo, que deve ser tratado pela equipe de desenvolvimento é o auto preenchimento habilitado (*Autocomplete Enabled*),que permite ao navegador guardar informações tais como o usuário e senha, sendo um risco de segurança e possibilitando o uso indevido por terceiros, na Tabela 17 contem as páginas com auto preenchimento:

Tabela 20 – A6 - Configuração incorreta de segurança - e-CIDADE

Vulnerabilidade	Página
Autopreenchimento Habilitado	/e-cidade/login.php
	/e-cidade/w/1/login.php
	/e-cidade/w/2/login.php
	/e-cidade/w/3/login.php
	/e-cidade/w/4/login.php
	/e-cidade/w/5/login.php

Fonte - NETSKAPER 4.9 - E-CIDADE versão 2018-2 (2018)

Como recomendação, foi identificada a versão do Apache 2.4, porém, a última versão esta na versão 2.5, *hackers* exploram falhas e vulnerabilidades conhecidas em versões mais antigas. Recomenda-se que o ambiente e os pacotes de segurança sejam mantidos atualizados para evitar exploração de falhas e vulnerabilidades conhecidas em versões antigas.

6.2 Sistema e-SIC Livre 1.4

Na Tabela 21 apresentam-se as páginas, em linguagem PHP, e os diretórios classificados pelas vulnerabilidades de médio e alto risco. Com base nos resultados apresentados possibilita-se analisar cada página e vulnerabilidade, buscando-se corrigi-las de forma a resolver ou mitigar o risco identificado.

Tabela 21 – Resultado das vulnerabilidades encontradas e confirmadas no sistema e-SIC Livre 1.4

Vulnerabilidade	Classificação OWASP	Quantidade
Blind SQL Injection (Confirmado)	A1 - Inserção de Código	1
Senha transmitida sobre protocolo HTTP (Confirmado)	A3 - Exposição de dados sensíveis	3
Auto Preenchimento Habilitado (Confirmado)	A6 - Configuração incorreta de segurança	2
Versões desatualizadas do PHP, jQuery e Apache (Confirmado)	A9 - Utilização de Componentes com Vulnerabilidades Conhecidas.	3

Fonte: NETSKAPER 4.9 - e-SIC Livre versão 1.4 (2018)

6.2.1 Vulnerabilidade: A1 - Inserção de Código.

Na Tabela 22 apresenta-se a página em PHP com a vulnerabilidade classificada como A1 - Inserção de Código.

Tabela 22 – A1 - Inserção de Código - e-SIC Livre 1.4

Vulnerabilidade	Página
Blind SQL Injection	/e-sic/restrito/index/index.php

Fonte - NETSKAPER 4.9 - e-SIC Livre versão 1.4 (2018)

A vulnerabilidade A1 - Inserção de Código é crítica e deve ser corrigida rapidamente, pois coloca em risco os dados e informações do banco de dados, através dessa vulnerabilidade um *hacker* pode ler, inserir, alterar e apagar dados de tabelas, criar, alterar e apagar tabelas, acessar os bancos de dados instalados e, até mesmo, acessar a alterar configurações do servidor MySql. A correção dessa vulnerabilidade é encontrada no Capítulo 3, item 3.3 deste trabalho.

6.2.2 Vulnerabilidade: A3 - Exposição de dados sensíveis.

Na Tabela 23 apresentam-se as páginas em PHP com as vulnerabilidades classificadas como A3 - Exposição de dados sensíveis.

Tabela 23 – A3 - Exposição de dados sensíveis - e-SIC Livre 1.4

Vulnerabilidade	Página
Senha transmitida sobre protocolo HTTP	/e-sic/restrito/index/index.php
	/e-sic/index.php
	/e-sic/alterasenha/index.php

Fonte - NETSKAPER 4.9 - e-SIC Livre versão 1.4 (2018)

Para corrigir esta vulnerabilidade recomenda-se configurar o uso pelo protocolo HTTPS com isto os dados sensíveis serão transmitidos criptografados sem o risco de serem interceptados e roubados por hackers.

6.2.3 Vulnerabilidade: A6 - Exposição de dados sensíveis.

Na Tabela 24 apresentam-se as páginas em PHP com as vulnerabilidades classificadas como A6 - Exposição de dados sensíveis.

Tabela 24 – A6 - Configuração incorreta de segurança - e-SIC Livre 1.4

Vulnerabilidade	Página
Auto Preenchimento Habilitado	/e-sic/index/index.php
	/e-sic/index/restrito/index.php

Fonte - NETSKAPER 4.9 - e-SIC Livre versão 1.4 (2018)

A vulnerabilidade A6 - Configuração incorreta de segurança tem um risco com o nível baixo e pode ser corrigida simplesmente desabilitando que o navegador do usuário guarde informações, tais como, o usuário e a senha.

6.2.4 Vulnerabilidade: A9 - Utilização de Componentes com Vulnerabilidades Conhecidas.

Na Tabela 25 apresentam-se as vulnerabilidades classificadas como A9 - Utilização de Componentes com Vulnerabilidades Conhecidas.

Tabela 25 – A9 - Utilização de Componentes com Vulnerabilidades Conhecidas - e-SIC Livre 1.4

Vulnerabilidade	Versão do software
PHP desatualizado	5.4.16
JQuery desatualizado	1.9.1
Apache desatualizado	2.4.6

Fonte - NETSKAPER 4.9 - e-SIC Livre versão 1.4 (2018)

A vulnerabilidade A9 - Utilização de Componentes com Vulnerabilidades Conhecidas tem um risco com o nível médio e precisam serem analisadas e corrigidas. A equipe de desenvolvimento precisa avaliar o impacto da atualização dos *softwares* anteriormente citados, realizando os devidos testes, para garantir que a aplicação não apresente problemas após as atualizações. As versões acima citadas como desatualizadas foram instaladas seguindo as recomendações e versões contidas no documento de instalação do sistema e-SIC Livre, porém evidenciamos que o PHP está na versão 7.1.11, o JQuery na versão 1.12.1 e o Apache na versão 2.4.29, na data da realização deste trabalho. Periodicamente as empresas de *softwares* atualizam seus produtos visando corrigir erros da aplicação e falhas de segurança, desta forma hackers podem usar de falhas de segurança conhecidas para explorar possíveis vulnerabilidades.

6.3 Sistema SIVAC 2.1

Na Tabela 26 apresentam-se as páginas, em linguagem PHP, e os diretórios classificados pelas vulnerabilidades de médio e alto risco. Com base nos resultados apresentados possibilita-se analisar cada página e vulnerabilidade, buscando-se corrigi-las de forma a resolver ou mitigar o risco identificado.

Tabela 26 – Resultado das vulnerabilidades encontradas e confirmadas no sistema SIVAC

Vulnerabilidade	Classificação OWASP	Quantidade
Cross-Site Scripting (Confirmado)	A7 - Cross-Site Scripting (XSS)	2
Senha transmitida sobre protocolo HTTP	A3 - Exposição de dados sensíveis	2

Fonte: NETSKAPER 4.9 - SIVAC 2.1 (2018)

6.3.1 Vulnerabilidade: A7 - Cross-Site Scripting (XSS).

Na tabela 27 apresentam-se as páginas PHP com vulnerabilidades confirmadas e classificadas como A7 - Cross-Site Scripting (XSS).

Tabela 27 – A7 - Cross-Site Scripting (XSS) - SIVAC

Vulnerabilidade	Página
Cross-Site Scripting	/ippes/Sistema/index.php
	/ippes/Sistema/Uf/index.php

Fonte: NETSKAPER 4.9 - SIVAC 2.1 (2018)

A correção das vulnerabilidades *Cross-site Scripting,* explicada no Capítulo 3, item 3.3 deste trabalho, é uma vulnerabilidade muito explorada por *hackers* , de alto grau de risco, exposição de dados sensíveis, *phishing* (forma utilizada por hackers para enganar os usuários e obter dados pessoais tais como: CPF, números de contas bancárias, cartões de crédito e senhas) e sequestro da sessão do usuário e desta forma tendo acesso à aplicação e os dados.

6.3.2 Vulnerabilidade: A3 - Exposição de dados sensíveis

Na Tabela 28 apresentam-se as páginas em PHP com as vulnerabilidades classificadas como A3 - Exposição de dados sensíveis.

Tabela 28 – A3 - Exposição de dados sensíveis - SIVAC

Vulnerabilidade	Página
Senha transmitida sobre protocolo HTTP	/ippes/Sistema/index.php
	/ippes/Sistema/Uf/index.php

Fonte: NETSKAPER 4.9 - SIVAC 2.1 (2018)

Para corrigir esta vulnerabilidade recomenda-se configurar o uso pelo protocolo HTTPS, com isto, os dados sensíveis serão transmitidos criptografados sem o risco de serem interceptados e roubados por hackers.

7 Análise do ambiente TIC e Estudos de Casos

Em complementação ao trabalho de laboratório realizado, buscou-se por casos reais que ratificam o trabalho aqui realizado, mostrando a importância de sua aplicação em prefeituras brasileiras.

Para identificar os casos, foi realizado uma pesquisa na *internet*, de notícias divulgadas no ano de 2018. No Anexo B são apresentadas as prefeituras e órgãos públicos que sofreram ataques e invasões e, estas, causando prejuízos financeiros e na operação de seus serviços à população. Nota-se que em alguns casos houve, até mesmo, a perda total do sistema e de seus dados, causando prejuízos incalculáveis às prefeituras, à população e partes interessadas.

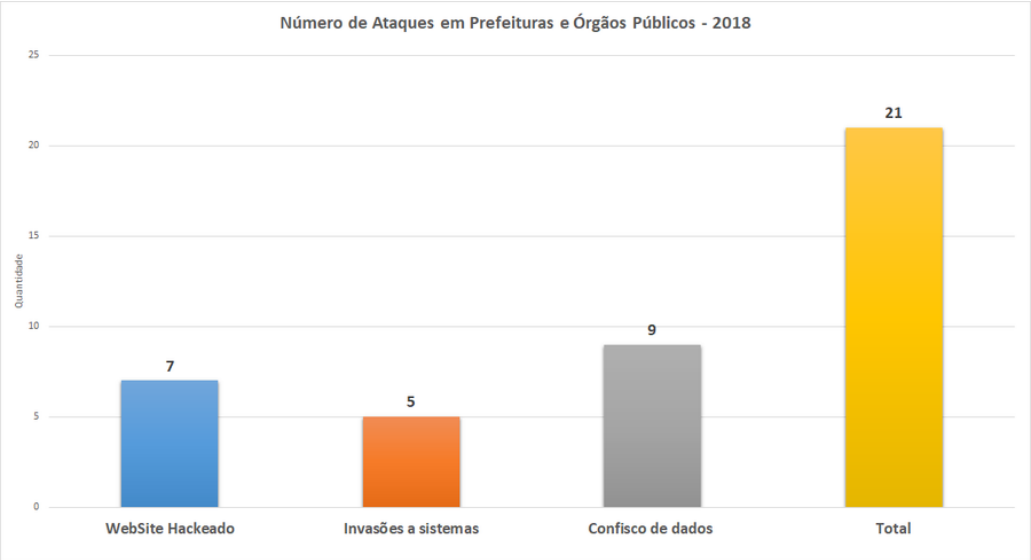
Destaca-se o aumento, nos últimos anos, de ataques a prefeituras e órgãos públicos, desde pequenas cidades até as grandes, todas são alvo e, dentre elas, algumas já foram vítimas de tentativas de ataques e invasões.

Algumas invasões foram motivadas devido à polarização política ocorrida no Brasil nesta última década, o que motiva grupos de invasores a lançarem ataques na forma de protesto ou revolta ao atual cenário político e social.

7.1 Incidentes de ataques a prefeituras e órgãos públicos divulgados em 2018

As informações na Figura 19 monstra a necessidade da implantação de metodologias e dispositivos de segurança que possam dar sustentação aos sistemas e seus ambientes TIC em prefeitura e órgãos públicos.

Figura 19 – Ataques em Prefeituras e Órgãos Públicos em 2018



Fonte: Pesquisa do autor em diversos meios de comunicação em 2018

7.2 Estudos de Caso e Discussões

São apresentados três estudos de casos, com base em incidentes ocorridos e descritos no Anexo B , onde o uso do método proposto de análise do ambiente TIC, propõe solucionar as brechas e riscos, desta forma garantindo a operação e funcionamento do ambiente e dos sistemas instalados.

7.2.1 Prefeitura de Serrana - Ataque: Ransomware

Abaixo segue a nota divulgada pela prefeitura em 08/03/2018:

Nota divulgada pela Prefeitura em 08/03/2018: “De acordo com uma nota divulgada pela administração municipal, o ataque foi descoberto quando funcionários chegaram para trabalhar e não conseguiram abrir os sistemas de cobranças de água e esgoto e, também, o de gerenciamento dos repasses para transporte escolar e de alunos que estudam fora do município. Uma mensagem exigia que os responsáveis pela prefeitura entrassem em contato com eles no prazo máximo de 72 horas, sob a ameaça de terem os arquivos completamente destruídos, caso não atendessem ao pedido. Os hackers pedem um resgate em moedas virtuais. Segundo a prefeitura, às informações armazenadas até o mês de janeiro foram recuperadas e as duas máquinas, que abrigavam os sistemas atacados, foram encaminhadas a um especialista na tentativa de recuperar os dados referentes aos meses de fevereiro e março. Por conta desse incidente, documentos como segunda via de boletos e contas de água não estão sendo emitidos. Há a possibilidade que estudantes que se beneficiam do repasse escolar façam um novo cadastro na secretaria da Educação. Os demais sistemas continuam funcionando normalmente.“

Fonte: Canal AcidadeOn. Disponível em: <https://www.acidadeon.com/ribeiraopreto/cotidiano/policia/NOT,0,0,1312267,
prefeitura+de+serrana+e+vitima+de+ataque+hacker.aspx>. Acesso em: 10/12/2018

Discussão com base na notícia duvilgada:

Propõe-se que a prefeitura de Serrana melhore seu gerenciamento de TIC, implantando processos e ferramentas que assegurem a proteção de seus dados, sistemas e continuidade dos serviços/negócios. A aplicação do método proposto, com base nos processos do COBIT 5, especificamente do processo **DSS05 - Garantir a Segurança do sistema**, norteia como deve-se realizar um gerenciamento dos serviços de segurança. No caso ocorrido a vulnerabilidade explorada foi *Ransomware* (Código malicioso que torna inacessível os dados armazenados em um equipamento, geralmente usando criptografia, e exige pagamento de um resgate em moedas virtuais - BitCoins) que é tratada pelo sub-processo **DSS5.01 - Proteger contra *malware*** onde, deve-se garantir que processos e medidas preventivas, detectivas e corretivas este-

jam aplicadas, tais como: frequente atualização de correções de segurança e anti-virus atualizados para não apenas os servidores mais para todos os ativos de TIC da prefeitura, de forma a proteger as informações e sistemas contra-ataques de *malwares* (vírus, *worms*, *spyware*, *spam*, desenvolvimento de *software* interno fraudulento, etc . . .)

A informação de que apenas *backup* de Janeiro/2018 estava disponível para recuperação, implica em perda de dados relevantes e importantes recentes, afetando diretamente os serviços, causando retrabalho e prejuízos financeiros. O processo **DSS04 - Gerenciar Continuidade**, define os processos e planos de ação necessários para continuidade dos negócios e serviços em casos de incidentes. Dentre os subprocessos destaca-se o **DSS04.07 - Gerenciar mecanismos de *backup***, onde devem ser criados processos e procedimentos para o armazenamento, retenção, descarte, backup e recuperação dos dados, isto aplicado para servidores e computadores críticos e para a organização. No caso, nota-se que os sistemas estavam instalados em computadores e não em servidores, o que dificulta a gestão eficiente e centralização dos *backups* por parte da equipe de TIC, portando, deve-se realizar a instalação dos sistemas em servidores, criar processos e instalar *softwares* apropriados para realização dos *backups* e , com periodicidade de *backups* diários, semanais e mensais, recomenda-se, também, o uso de conjuntos de mídias diferentes e apropriados para a retenção, conforme a Política de Backup e Retenção definida. Recomenda-se também, manter cópias de *backups* em locais externos controlados e monitorados, em caso de perca total do ambiente TIC, pode-se realizar a recuperação dos serviços e do ambiente TIC em outra localidade.

Destaca-se também, em uma primeira análise, que não há evidências de um plano ou processos que garantam a continuidade dos serviços, desta forma, recomenda-se a implantação do processo **DSS04 - Gerenciar Continuidade** , um plano de continuidade dos serviços deve ser desenvolvido, implementado, testado periodicamente, divulgado e treinamentos devem ser realizados.

Evidentemente um estudo e análise mais detalhada do ambiente e processos TIC deve ser realizada para ter um diagnóstico mais consistente e uma apresentação mais completa ser apresentada.

Também tomando como base as outras notícias vinculadas na tabela 38, as prefeituras de Jóia - RS, Rondonópolis - MT , São Carlos - SC, Mirandópolis - SP, Catuipe - RS, São Gabriel da Palha - ES, Ituiutaba - MG, Cambira - PR e Vera - MT sofreram o mesmo tipo de ataque por *Ransomware* e a solução aqui apresentada pela metodologia pode ser aplicada, em parte ou totalmente, e recomenda-se aplicar os processos da metodologia simplificada proposta neste trabalho, conforme apresentado na tabela 10.

7.2.2 TSE investiga se hackers invadiram sistema da Justiça Eleitoral

Notícia divulgada: “TSE (Tribunal Superior Eleitoral) vai investigar se hackers tiveram acesso ao sistema interno da Justiça Eleitoral. Os invasores alegam ter o código-fonte do sistema Gedai-UE, que grava o sistema operacional e a lista de eleitores na urna eletrônica; no entanto, eles não teriam quebrado o sigilo do voto. O Gedai-UE (Gerenciador de Dados, Aplicativos e Interface com a Urna Eletrônica) é um programa para ’Windows’ que gera cartões de memória com três itens: o sistema operacional Uenux (Urna Eletrônica com Linux), a lista de candidatos e a lista de eleitores. Esses cartões são inseridos na urna eletrônica para atualizá-la antes de cada eleição. O código do Gedai-UE não é totalmente secreto. Alguns órgãos — como Ministério Público, OAB (Ordem dos Advogados do Brasil) e partidos políticos — podem obtê-lo caso assinem um termo de sigilo. *Hackers* alegam, no entanto, que tiveram acesso não-autorizado a esse sistema. O invasor diz que conseguiu usar a *intranet* do TSE por vários meses, e obteve milhares de códigos-fontes, documentos sigilosos e até mesmo credenciais. Isso incluiria as credenciais de Sérgio Banhos, ministro substituto do TSE e do secretário de tecnologia Giuseppe Janino, criador do coletor eletrônico de voto.“

Fonte: Tecnoblog. Disponível em: <https://tecnoblog.net/266785/tse-investiga-hackers-justica-eleitoral/>. Acesso em: 10/12/2018

Discussão com base na notícia divulgada:

Pela notícia, o grupo de *hackers* obteve acesso ou burlou a segurança de acesso da *intranet* do TSE (Tribunal Superior Eleitoral), desta forma, por vários meses tiveram acessos a milhares de códigos-fontes, documentos e até mesmo, as credenciais de usuários. Creio que ,neste ponto, a avaliação não deva dar foco a segurança do sistema Gedai-UE ou, até mesmo, se houve ou não qualquer manipulação durante o processo eleitoral ocorrido em 2018. Pois como mencionado, cópias do sistema com seu código-fonte são distribuídos para o Ministério Público, a OAB e a partidos políticos, isto mediante a assinatura de um termo de sigilo, portanto o vazamento, apenas do código-fonte do sistema Gedai-UE, ou possíveis fraudes poderiam ter ocorridos com origem em outra localidade ou ambiente.

Não se menciona na matéria, como o grupo de *hackers* conseguiu o acesso á *intranet* do TSE, certamente o grupo também não irá divulgar publicamente, porém pode-se explorar três hipoteses: 1ª - obteve-se acesso através de uma credencial válida e ativa de um funcionário ativo ou inativo, 2ª - obteve-se acesso através de uma conta administrativa, ou uma conta de serviço ou através da vulnerabilidade de um sistema ou aplicação, provavelmente utilizada em servidores, 3ª - obteve-se acesso físico a ativos de TIC internamente no TSE, desta forma obtendo acesso a *intranet*.

Na primeira hipótese, o processo **DSS05 - Gerenciar Serviços de Segurança** devem ser avaliados, recomenda-se a avaliação e implantação dos processos e sub-processos e suas atividades conforme Tabela 29.

Tabela 29 – TSE - Primeira hipótese - Acesso a uma credencial válida de funcionário ativo ou inativo.

Atividade	Descrição da Atividade
BAI09.02 - Gerenciar ativos críticos	
DSS05.02 - Gerenciar a segurança das conexões de rede e conectividade	
08	Assegurar que os serviços de acesso remoto e perfis de usuário (ou outros meios utilizados para a manutenção ou diagnóstico) estão ativos somente quando necessário.
DSS05.02 - Gerenciar a segurança da rede e conectividade.	
01	Baseado em avaliações de risco e requisitos de negócio, estabelecer e manter uma política para segurança de conectividade.
02	Permitir que somente dispositivos autorizados tenham acesso à informações corporativas e à rede. Configure estes dispositivos para forçar a entrada de senha.
03	Implementar mecanismos de filtragem, como *firewalls* e *software* de detecção de intrusão, com políticas apropriadas para controlar o trafego de entrada e saída.
04	Criptografar informações em trânsito de acordo com sua classificação.
05	Aplicar protocolos de segurança aprovados a conectividade de rede.
06	Configurar equipamentos de rede de maneira segura.
07	Estabelecer mecanismos confiáveis para suportar a transmissão e recepção segura de informações
08	Executar testes de invasão periodicamente para determinar a adequabilidade da proteção da rede

Atividade	Descrição da Atividade
09	Executar testes de segurança de Sistema periodicamente para determinar a adequabilidade da proteção dos sistemas.
DSS05.04 - Gerenciar identidade de usuários e acesso lógico	
01	Manter os direitos de acesso de usuários de acordo com os requisitos funcionais de negócio e de processo. Alinhar a gestão de identidades e direitos de acesso às funções e responsabilidades definidas, com base nos princípios de least-privilege, need-to-have e need-to-know.
02	Identificar unicamente toda atividade de processamento de informação por papéis funcionais, coordenando com as unidades de negócio para garantir que todos os papéis estejam consistentemente definidos, incluindo papéis pelas áreas de negócio, dentro de suas aplicações de negócio.
03	Autenticar todo acesso a ativos de informação baseado em sua classificação de segurança, coordenando com as unidades de negócio que gerenciam a autenticação em aplicações usadas em processos de negócio para garantir que controles de autenticação foram administrados corretamente.
04	Administrar todas as mudanças de direitos de acesso (criação, modificações e exclusões) para que tenham efeito no tempo apropriado baseado somente em transações documentadas e aprovadas por pessoal designado de gerenciamento.
05	Segregar e gerenciar contas de usuários privilegiados.
06	Realizar regularmente a revisão de todas as contas e privilégios relacionados.
07	Certificar-se que todos os usuários (internos, externos e temporários) e a sua atividade em sistemas de TIC (aplicações de negócios, infraestrutura de TIC, operações de sistema, desenvolvimento e manutenção) são exclusivamente identificáveis. Identificar exclusivamente todas as atividades de processamento de informações por usuário.
08	Manter uma trilha de auditoria de acesso à informação classificada como altamente sensível.

Fonte: ISACA (2012) - COBIT 5 - Habilitando Processos

Segunda hipótese, recomendam-se também os processos **DSS05 - Gerenciar Serviços de Segurança**, seus sub-processos e atividades conforme descritos na Tabela 30.

Tabela 30 – TSE - Segunda hipótese - Vulnerabilidade de segurança em contas de serviço

Atividade	Descrição da Atividade
DSS05.01 - Proteger contra *malware*	
01	Realizar conscientização sobre *software* malicioso e aplicar procedimentos para prevenção e responsabilidades.
02	Distribuir todos os *softwares* de proteção a partir de um ponto central (versão e nível de patch) usando gerenciamento de configuração e mudanças centralizados.
03	Filtrar táfego de entrada, como e-mail e *downloads*, para proteção contra informações não solicitadas (ex.: *spyware*, e-mails de phishing)
04	Conduzir periodicamente, treinamento sobre *malware* em e-mails e uso da *internet*. Treinar os usuários a não instalar software compartilhado ou não aprovado.
DSS05.03 - Gerenciar a segurança dos *endpoints*	
01	Configurar sistemas operacionais de maneira segura.
02	Implementar mecanismos de travamento.
03	Encriptar informações armazenadas de acordo com sua classificação.
04	Gerenciar acessos e controle remoto.
05	Gerenciar configurações de rede de maneira segura.
06	Implementar filtragem de tráfego de rede em dispositivos *endpoint*.
07	Proteger a integridade dos sistemas.

Fonte; ISACA (2012) - COBIT 5 - Habilitando Processos - adaptado pelo autor

Em geral, o acesso a *intranet,* por se tratar de uma rede interna, somente é realizado por meio do uso de *softwares* de VPN - *Virtual Private Network* (termo em inglês para Redes Privadas Virtuais), tecnologia que permite conectar-se a uma rede interna através de outras redes externas e públicas, e tendo acesso tanto a *Intranet*, aos sistemas, servidores de correio eletrônico internos e servidores de arquivos. Como pode ter ocorrido uma falha de segurança física, recomenda-se, também, dentro do processo **DSS05 - Gerenciar Serviços de Segurança**, os sub-processos e atividades descritas na Tabela 31

Tabela 31 – TSE - Terceira hipótese - Acesso físico a equipamentos físicos do TIC

Atividade	Descrição da Atividade
DSS05.05 - Gerenciar o acesso físico aos ativos de TIC	
01	Gerenciar as solicitações e concessões de acesso aos recursos de computação. Pedidos formais de acesso devem ser preenchidos, e autorizados pela gestão do *site* de TIC, e os registros mantidos. Os formulários devem identificar especificamente as áreas em que o indivíduo é concedido acesso.
02	Garantir que os perfis de acesso continuem atualizados. Associe o acesso às instalações de TI (salas de servidores, edifícios, áreas ou zonas) a funções de trabalho e responsabilidades.
03	Registrar e monitorar todos os pontos de entrada a instalações de TIC. Registre todos os visitantes e fornecedores às instalações.
04	Instruir todos os funcionários a apresentar uma identificação visível em todos os momentos. Impedir a emissão de identificações ou crachás sem a devida autorização.
05	Exigir que visitantes sejam acompanhados todo o tempo, enquanto no local. Se um indivíduo desacompanhado, desconhecido ou sem identificação pessoal é identificado, alertar o pessoal de segurança.
06	Restringir o acesso a *sites* de TI sensíveis, estabelecendo restrições de perímetro, tais como cercas, paredes e dispositivos de segurança nas portas interiores e exteriores. Certifique-se de que os dispositivos registrem a entrada e disparem um alarme em caso de acesso não autorizado. Exemplos de tais dispositivos incluem crachás ou cartões-chave, teclados para senha, circuito fechado de televisão e scanners biométricos.

Atividade	Descrição da Atividade
07	Conduza treinamentos de conscientização sobre segurança física regularmente.

Fonte; ISACA (2012) - COBIT 5 - Habilitando Processos - adaptado pelo autor

Recomenda-se a integração on-line dos sistemas, *intranet*, acessos a rede, acessos biométricos e catracas de acesso com o sistema de recursos humanos, de forma a, todos os acessos de um funcionário desligado, sejam imediatamente revogados e, também, que todos os acessos sejam suspensos em períodos de férias, afastamentos e licenças médicas.

Pela análise mais consistente, é fundamental investigar e descobrir como o grupo de *hackers* teve o acesso á *Intranet*, aos dados, códigos-fontes e credenciais. Visto que a matéria, apenas trata do acesso ás credenciais do ministro substituto Sérgio Banhos e do secretário de tecnologia Giuseppe Janino, desta forma, não é possível saber quais mais informações, códigos-fontes, documentos e credenciais os invasores tiveram acesso e, avaliar o grau do risco da exposição dos dados e informações. A divulgação dos outros documentos, outros códigos-fontes de sistemas roubados, outras credenciais e sua distribuição para outros grupos *hackers* ou grupos de interesse contrários ao atual processo eleitoral brasileiro pode comprometer sistemas, informações e a confiabilidade nos sistemas do TSE.

7.2.3 Invasão do web-site de Aracaju - SE

Notícia divulgada em 31/03/2018 pelo web-site F5 News: Web-site da Prefeitura de Aracaju é invadido por hackers. Um grupo de hackers invadiu o web-site da Prefeitura de Aracaju neste sábado (31). Na página principal, apareceu uma mensagem contra os políticos, de forma geral, e contra o presidente da República, Michel Temer. As mensagens, assinadas pelo *login* D4RKR0N, permaneceram na *internet* por algumas horas antes que a página fosse retirada do ar por algum tempo enquanto seus responsáveis corrigiam a publicação.

Fonte: F5News. Disponível em: <http://www.f5news.com.br/cotidiano/site-da-prefeitura-de-aracaju-e-invadido-por-hackers_45880> - Consulta em 10/12/2018.

Discussão com base na notícia divulgada:

Para invadir o *web-site* da Prefeitura de Aracaju o *hacker* pode ter utilizado de alguma vulnerabilidade conhecida no ambiente TIC onde o *web-site* está instalado, agora é possível consultar que o *site* está desenvolvido em PHP 5.4, HTML 5, Java Script e JQuery. Estas informações podem ser facilmente obtidas através dos *web-sites*, tais como: w3techs - https://w3techs.com ou BildWith - https://builtwith.com, onde são apresentados detalhes como a linguagem PHP, o sistema operacional Linux

Debian, serviço habilitado de OpenSSL e o uso do Apache 2. Estas informações são importantes para que o *hacker* possa explorar alguma vulnerabilidade conhecida e de versões desatualizadas.

Não é possível afirmar como o *hacker* conseguiu com sucesso realizar a invasão, porém, alguns processos do método proposto podem ser aplicados, de forma, a evitar uma nova exposição a riscos.

O processo **DSS05 - Garantir a Segurança do Sistema** , sub-processo **DSS05.01 - Prevenção, Detecção e Correção de *Softwares* Maliciosos** deve ser implementado, as atualizações de segurança dos *softwares* e do antivírus, tanto para o servidor de aplicação quanto para o servidor de banco de dados, devem ser realizados frequentemente.As versões dos *softwares*, sempre que possível e avaliado pela área de desenvolvimento de sistemas, devem estar com a última versão instalada, pois é uma prática comum em novas versões se tratarem problemas funcionais e problemas de segurança, desta forma, atualizando o ambiente a aplicação não esta exposta às vulnerabilidades conhecidas das versões anteriores. Na análise do *web-site* da Prefeitura de Aracaju, destaca-se que a versão do PHP esta na versão 5.4.9 e, na data de realização deste trabalho, a versão mais atualizada esta na versão 7.1.11, da mesma forma, evidencia-se que versão do Apache esta na versão 2.4.6 e a versão mais atualizada esta na versão 2.4.29.

O processo **DSS05.04 - Gerenciar identidade de usuários e acesso lógico** deve ser implementado para os acessos aos servidores, bloqueio de contas por tentativa, número de caracteres para as senhas, uso de senhas fortes, contas com super direitos devem ter seus acessos restritos a rede interna, não deve estar habitado com permissão de acesso remoto e ser frequentemente alterada (no mínimo a cada 90 dias). Também, quando o acesso é realizado através de outras contas, deve-se verificar a possibilidade de vincular o acesso ao cadastro de funcionários no departamento de recursos humanos, para que estes acessos sejam manualmente ou automaticamente revogados, em caso de saída do funcionário, terceiro ou prestador de serviço.

Também deve-se aplicar o processo **BAI06 - Gerenciar Mudanças** para as mudanças ocorridas no *web-site*, de maneira a garantir que novas vulnerabilidades não sejam inseridas na aplicação em novas versões do *web-site*.

Destaca-se a importância de uma análise mais profunda do ambiente TIC, para se identificar outros possíveis riscos, e como podem ser corrigidos ou mitigados, dessa forma recomendo a aplicação de todos os processos da proposta apresentada na Tabela 9.

A mesma solução apresentada também é indicada para o *web-site* da Prefeitura de Belo Horizonte que sofreu o mesmo tipo de ataque em 30/05/2018.

8 CONCLUSÃO

Este trabalho procurou responder a questão: Como aumentar a segurança dos web-sites desenvolvidos em linguagem PHP em Prefeituras?

Para isto, buscou-se contextualizar o uso e crescimento do *software* livre pelo setor público, o uso da linguagem PHP nos desenvolvimentos de *web-sites*, as iniciativas públicas privadas e, nesse sentido, foram apresentados três sistemas desenvolvidos para prestar serviços à população, dentro do modelo de Cidades Inteligentes. Também, se abordou os riscos existentes em operações de sistemas WEB e ambientes TIC.Três vulnerabilidades mais exploradas por hackers, conforme classificação da lista OWASP *Top Ten 2017*, que foram apresentados na Tabela 1, e como resolvê-las em linguagem PHP. Por fim se apresentou o *framework* COBIT 5 da ISACA, com os seus trinta e sete processos e como são classificados quando ao nível de maturidade.

Com base nos estudos, realizou-se uma pesquisa bibliográfica sobre ataque cibernético em ambientes WEB, metodologias de gerenciamento de TIC e ferramentas e métodos de teste de vulnerabilidades. Deste trabalho, propõe-se a adoção e aplicação de um metodologia simplificada para a análise de segurança em aplicações WEB e ambientes TIC, conforme ilustrada na Figura 14, onde, parte-se do entendimento do sistema objeto da avaliação, avalia-se o ambiente TIC, buscando aferir o nível de maturidade dos sete processos de segurança propostos, conforme apresentado na Tabela 9, aplica-se a avaliação das vulnerabilidades, que consiste em um processo utilizado para detectar, identificar e classificar vulnerabilidades e seu grau de risco, por meio do uso do *software* NETSPARKER 4.9 e apresentam-se os resultados do ambiente e as vulnerabilidades encontradas, classificadas pela lista OWASP *Top Ten* 2017. Com base nos resultados obtidos, buscou-se soluções ou mitigações para os riscos encontrados.

No Capítulo 3, Seção 3.3, foram apresentadas as três das principais vulnerabilidades exploradas por *hackers* em ambientes WEB e como resolvê-las, tomando como base a classificação do risco pela lista OWASP *Top Ten* 2017 e a importância da adoção e uso da metodologia de **Ciclo de Vida do Desenvolvimento Seguro** (SDLC - *Security Development Lifecycle*).

Para a validação proposta neste trabalho, pesquisou-se três sistemas WEB distribuídos gratuitamente para prefeituras, foi construído um laboratório de testes, conforme abordado no Capítulo 4, Seção 4.2.4, onde foram instalados os sistemas WEB apresentados no capítulo 4, seção 4.2.1, executando rigorosamente o procedimento de instalação para cada sistema, tais como: o sistema operacional recomendado, os *softwares*, serviços e nas versões conforme descrito por eles. Desta forma, buscou-se reproduzir a instalação em laboratório semelhantemente a uma instalação em

ambiente real. A análise de vulnerabilidade foi aplicado nos sistemas e-Cidade versão 2018-2, e-SIC Livre 1.4 e SIVAC 2.1, em todos os casos de estudo, foram detectadas vulnerabilidades, que foram classificadas, conforme seu grau de risco, e usando como referência a lista OWASP *Top Ten*, presentadas no Capítulo 4.

Para cada sistema avaliado, suas vulnerabilidades foram individualizadas e tema de discussão no Capítulo 6, onde foram apresentadas propostas de solução e recomendação para mitigação, que servem como contribuição, auxiliando na tomada de decisão por parte das prefeituras e unidades de saúde, que pretendem adorar estas soluções, de forma a aplicar as correções dos riscos necessários e proposto neste trabalho para garantir um sistema e ambiente TIC seguros.

Por fim, foram apresentados no Capítulo 7, três estudos de caso ocorridos em prefeituras e no TSE no ano de 2018, onde hackers burlaram a segurança do ambiente TIC, tiveram acesso a informações confidenciais, código-fonte de sistemas, alteraram e derrubaram *web-sites* e sequestraram dados, causando prejuízos financeiros, dano a imagem e, até mesmo, perda de dados. Nos estudos de caso, focou-se apenas na notícia divulgada e não em uma análise mais profunda do ambiente TIC, dessa forma se relacionou os processos, apresentados na metodologia simplificada proposta na Tabela 9, bem como, nos processos a serem implantados ou melhorados, de forma a garantir um sistema e ambiente TIC seguro.

Este trabalho contribuiu com uma proposta de uma metodologia simplificada, onde são aplicados sete dos trinta e sete processos do COBIT 5, focando-se na segurança do ambiente TIC, a aplicação de um processo de análise de vulnerabilidades, desde o desenvolvimento do sistema e sua operação, conforme metodologia do Ciclo de Vida do Desenvolvimento Seguro (SDLC - *Security Development Lifecycle*) apresentada na Figura 10, nos testes e a identificação das vulnerabilidades e falhas de segurança existentes nos sistemas: e-CIDADE, e-SIC e SIVAC. Desta forma, os artefatos gerados pelos estudos de casos, possibilitam a tomada de decisão dos gestores responsáveis pela unidade governamental e, também, fornecem insumos para auxiliar a equipe de desenvolvimento, de cada sistema avaliado, nas correções e mitigações das vulnerabilidades aqui detectadas.

Conclui-se que foram atingidos os resultados esperados por meio da aplicação do método proposto e dos conhecimentos aqui apresentados, pois são úteis e relevantes para se garantir um sistema WEB e ambiente TIC seguro.

Rassalta-se a importância da adoção do método proposto neste trabalho, por prefeituras e órgãos públicos, garantindo a segurança de seus dados, sistemas e ambiente TIC, e disponibilidade e continuidade dos serviços prestados à população, em um mundo cada vez mais conectado e mais vulnerável a ataques cibernéticos.

Existem aspectos não abordados, por esse trabalho, que podem ser desenvolvidos em trabalhos futuros, tais como, a aplicação do método proposto em cenários reais, a verificação da eficiência de outros *softwares* de análise de vulnerabilidade, tanto as ferramentas gratuitas, quanto as de código-aberto , bem como, outras soluções comerciais. Também sua aplicação em outros sistemas WEB governamentais, sejam estes, de distribuição livre, de parceria pública privada ou desenvolvimento próprio.

Referências

ABNT, N. *ABNT ISO/IEC 27.002 Tecnologia da informação - Técnicas de segurança — Código de Prática para Controles de Segurança da Informação.* 1. ed. Rio de Janeiro: Associação Brasileira de Normas Técnicas, 2013.

ALTAF, I. et al. Vulnerability Assessment and Patching Management. In: ECE FET, M. Department of (Ed.). *2015 International Conference on Soft Computing Techniques and Implementations- (ICSCTI)*. Faridabad, India: [s.n.], 2015.

ASSUNÇÃO, M. F. A. *Honeypots e Honeynets*: Aprenda a Detectar e Enganar Invasores. Florianópolis: Visual Books, 2008.

ASSUNÇÃO, M. F. A. *Análise de Eficiência na Detecção de Vulnerabilidades em Ambientes WEB com o uso de Ferramentas de Código Aberto*. 2015. 85 p. Dissertação (Sistemas de Informação e Gestão da Informação) — UNIVERSIDADE FUMEC.

BALANCED SCORECARD INSTITUTE. *About Balanced Scorecard*. 2018. Disponível em: <https://www.balancedscorecard.org/BSC-Basics/About-the-Balanced-Scorecard>. Acesso em: 2018.

BELARMINO, V. F. *Análise de vulnerabilidades computacionais nos repositórios digitais das Universidades Federais*. 2014. 62 p. Dissertação (Graduação em Biblioteconomia) — Universidade Federal da Paraíba.

BINGCHANG, L. et al. Software Vulnerability Discovery Techniques: A Survey. *Fourth International Conference on Multimedia Information Networking and Security*, IEEE, Nanjing, China, p. 152 – 156, 2012. ISSN 2162-8998.

BRASIL - SENADO FEDERAL. Lei no 8.666/1993 - Licitações e Contratos. *Lei no 8.666/1993 - Licitações e Contratos*, 2017.

BRASIL, TRIBUNAL DE CONTAS DO ESTADO DO RIO DE JANEIRO. Governança de TI na Administração Municipal. *Levantamento em Tecnologia da Informação*, p. 627 – 651, 2013.

CAMPOS, A. *Sistema de Segurança da Informação - Controlando os Riscos*. 3. ed. Florianópolis: Visual Books, 2014.

CERT.BR - COMITÊ GESTOR DA INTERNET NO BRASIL. *Cartilha de Segurança para Internet versão 4.0*. São Paulo: [s.n.], 2012. Disponível em: <https://cartilha.cert.br/livro/cartilha-seguranca-internet.pdf>. Acesso em: 2019.

CHEN, S. *WAVSEP 2017/2018 - Evaluating DAST against PT/SDL Challenges*. 2018. Online. Disponível em: <http://sectooladdict.blogspot.com/2017/11/wavsep-2017-evaluating-dast-against.html>. Acesso em: 10 Janeiro 2019.

CISL - COMITÊ DE IMPLEMENTACÃO DE SOFTWARE LIVRE. Desenvolvimento de ferramentas e soluções em software livre. 2018. Disponível em: <http://www.softwarelivre.gov.br/levantamento/levantamento/levantamento>. Acesso em: 01/05/2018.

CRESPO, M. A. C.; CHÓEZ, R. E. R. *Estudio del impacto financiero de las vulnerabilidades de las páginas Web de los bancos en Ecuador*. 2012. 193 p. Dissertação (Graduação em Ingenieria de Sistemas) — Universidad Politécnica Salesiana, Guayaquil.

DOUPÉ, A. et al. Enemy of the State: A State-Aware Black-Box Web Vulnerability Scanner. In: *USENIX Security Symposium*. [s.n.], 2012. Disponível em: <https://www.usenix.org/system/files/conference/usenixsecurity12/sec12-final225.pdf>. Acesso em: 15 Dezembro 2018.

DOYLE, M.; WALDEN, J. An Empirical Study of the Evolution of PHP Web Application Security. *Third International Workshop on Security Measurements and Metrics*, p. 12 – 20, 2011.

ELMAGHRABY, A. S.; LOSAVIO, M. M. Cyber security challenges in Smart Cities: Safety, security and privacy. *Journal of Advanced Research*, p. 491 – 497, 2014.

FONSECA, J. J. S. *Metodologia da pesquisa científica*. Fortaleza: Universidade Estadual do Ceará, 2002.

GUIMARAES, B. D. A. S. M. *SQL Map*. 2018. Disponível em: <http://sqlmap.org/>. Acesso em: 02/06/2018.

HAQUE, M. F.; MIAH, M. B. A.; MASUD, F. A. Enhancement of Web Security Against External Attack. *European Scientific Journal*, v. 13, p. 228 – 239, Maio 2018.

HAUZAR, D.; KOFRON, J. On Security Analysis of PHP Web Applications. *36th International Conference on Computer Software and Applications Workshops*, p. 577 – 582, 2012.

HUANG, S.; LEE, C.; KAO, A. Balancing Performance Measures for Information Security Management: A Balanced Scorecard Framework. In: HUANG, S.; LEE, C.; KAO, A. (Ed.). *Industrial Management & Data Systems*. [S.l.: s.n.], 2006. p. 242 – 255.

IPPES - INSTITUTO DE PESQUISA P. E P DA EDUCACÃO E SAÚDE. *SIVAC - Sistema Online de Vacinação*. 2018. Disponível em: <https://softwarepublico.gov.br/social/sivac>. Acesso em: 2018.

ISACA. *COBIT 5*. 2012. Information Systems Audit and Control Association. Disponível em: <http://www.isaca.org/COBIT/Pages/default.aspx>. Acesso em: 2017.

JNENA, R. M. F. *Modern Approach for WEB Applications Vulnerability Analysis*. 2013. 86 p. Dissertação (Faculty of Engineering) — The Islamic University of Gaza.

KEIZUN, A. et al. *Ardilla*. 2018. Disponível em: <https://groups.csail.mit.edu/pag/ardilla/>. Acesso em: 02/06/2018.

KIEZUN, A. et al. Automatic Creation of SQL Injection and Cross-Site Scripting Attacks. *ICSE'09*, p. 199 – 209, Maio 2009.

KUMAR, D. A.; REDDY, K. Constructing Secure Web Applications With Proper Data Validations. In: *IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*. Jaipur, India: IEEE, 2014.

LLC, R. *Metasploit*. 2018. Disponível em: <https://www.metasploit.com/>. Acesso em: 02/06/2018.

MADAN, S.; MADAN, M. S. Security Standards Perspective to Fortify Web Database Applications From Code Injection Attacks. *International Conference on Intelligent Systems, Modelling and Simulation*, p. 226 – 230, 2010.

MARCONI, M. de A.; LAKATOS, E. M. *Metodologia do trabalho científico*: procedimentos básicos, pesquisa bibliográfica, projeto e relatório, publicações e trabalhos científicos. São Paulo: Atlas S.A., 2001.

MARCONI, M. de A.; LAKATOS, E. M. *Fundamento da Metodologia Científica*. 5. ed. São Paulo: Atlas S.A., 2003. ISBN 85-224-3397-6.

MEDEIROS, I.; NEVES, N.; CORREIA, M. Detecting and Removing Web Application Vulnerabilities with Static Analysis and Data Mining. *TRANSACTIONS ON RELIABILITY*, v. 65, n. 1, p. 54 – 69, Março 2016.

MERLO, E.; LATARTE, D.; ANTINIOL, G. SQL-Injection Security Evolution Analysis in PHP. *IEEE*, 2007.

MILENA, L. *Por que incentivar softwares livres no Brasil*. 2013. Disponível em: <https://jornalggn.com.br/noticia/por-que-incentivar-softwares-livres-no-brasil>. Acesso em: 01/05/2018.

MINISTÉRIO DE PLANEJAMENTO, DESENVOLVIMENTO E GESTÃO. *Estratégia da política de software livre no governo federal*. 2015. Disponível em: <http://www.planejamento.gov.br/assuntos/logistica-e-tecnologia-da-informacao/noticias/estrategia-da-politica-de-software-livre-no>. Acesso em: Outubro, 2018.

MINISTÉRIO DE PLANEJAMENTO, DESENVOLVIMENTO E GESTÃO. *e-Cidade - Sistema de Gestão de Cidades*. Brasília: [s.n.], 2018a. Disponível em: <https://softwarepublico.gov.br/social/e-cidade>. Acesso em: 2018.

MINISTÉRIO DE PLANEJAMENTO, DESENVOLVIMENTO E GESTÃO. *e-SIC - Sistema de Informação ao cidadão*. Brasília: [s.n.], 2018b. Disponível em: <https://softwarepublico.gov.br/social/e-sic-livre>. Acesso em: 2018.

MIRJALILI, M.; NOWROOZI, A.; ALIDOOSTI, M. A survey on web penetration test. *Advances in Computer Science: an International Journal*, v. 3, n. 12, Novembro 2014. ISSN 2322-5157. Disponível em: <www.ACSIJ.org>.

MONGA, M.; PALEARI, R.; PASSERINI, E. A hybrid analysis framework for detecting web application vulnerabilities. *SESS'09, Vancouver, Canada*, p. 25 – 32, 2009.

NEGRE, E.; ROSENTHAL-SABROUX, C.; GASCÓ, M. A KNOWLEDGE-BASED CONCEPTUAL VISION OF THE SMART CITY. *48th Hawaii International Conference on System Sciences*, p. 2317 – 2325, 2015.

NETSPARKER. *NETSPARKER DESKTOP 4.9*. 2018. Disponível em: <https://www.netsparker.com/>. Acesso em: 2018.

OFFENSIVE SECURITY. *Kali Linux*. 2018. Disponível em: <https://www.kali.org/>. Acesso em: 02/06/2018.

ORTNER, E. et al. Design of Interactional End-to-End Web Applications for Smart Cities. *International World Wide Web Conference Committee (IW3C2)*, p. 551 – 556, 2015.

OSÓRIO, T. L. G. et al. Utilização de Software Livre em Órgãos Públicos. *II Simpósio de Excelência em Gestão e Tecnologia – SEGeT*, p. 1039 – 1058, 2005.

OWASP. *OWASP Top Ten Project 2017*. 2018. Disponível em: <https://www.owasp.org/images/7/72/OWASP_Top_10-2017_(en).pdf.pdf>. Acesso em: 01/05/2018.

OWASP. *OWASP Zed Attack Proxy*. 2018. Disponível em: <https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project>. Acesso em: 02/06/2018.

PHPBB GROUP. *phpBB 3.2*. 2018. Disponível em: <http://www.phpbb.com>. Acesso em: 02/06/2018.

PINA, C. B. A. *O uso do software livre na Gestão Pública*. Brasília: [s.n.], 08/05/2014. Disponível em: <http://www.conteudojuridico.com.br/?artigos&ver=2.47910&seo=1>. Acesso em: 05/01/2019.

PLC, S. *SDL Certification*. 2018. Disponível em: <https://www.sdl.com/trainings/certification.html>. Acesso em: 18/05/2018.

PORTSWIGGER. *Burp Suite*. 2018. Disponível em: <https://portswigger.net/burp/>. Acesso em: 02/06/2018.

QASAIMEH, M.; SHAMLAWI, A.; KHAIRALLAH, T. Black Box Evaluation of WEB Application Scanners: Standards Mapping Approach. *Journal of Theoretical and Applied Information Technology*, v. 96, n. 14, Julho 2018. ISSN 1992-8645. Disponível em: <www.jatit.org>.

RICHARDSON, R. J. *Pesquisa Social*: Métodos e Técnicas. 3. ed. São Paulo: Atlas S.A., 1999.

SEN, M. et al. Issues of privacy and security in the role of software in smart cities. *International Conference on Communication Systems and Network Technologies*, p. 518 – 523, 2013.

SMART CITY EXPO. Smart City Expo Brazil. Curitiba - PR, 2018. Disponível em: <https://www.tecmundo.com.br/mobilidade-urbana-smart-cities/127482-smart-city-expo-maior-mundo-cidades-inteligentes-chega-brasil.htm>. Acesso em: 01/05/2018.

SOUZA, E. S. de. A gestão da TI dentro do Serviço Público. In: *Simpósio de Excelência em Gestão e Tecnologia*. [S.l.: s.n.], 2013.

TEIXEIRA, G. *Além das capitais: 4 cidades inteligentes pelo Brasil*. 2018. Disponível em: <https://descola.org/drops/alem-das-capitas-4-cidades-inteligentes-pelo-brasil/>. Acesso em: Outubro. 2018.

TENABLE. *Nessus Vulnerability Scanner*. 2018. Disponível em: <https://www.tenable.com/products/nessus/nessus-professional>. Acesso em: 02/06/2018.

TIOBE. 2019. Disponível em: <https://www.tiobe.com/tiobe-index/>. Acesso em: 19/01/2019.

VOITOVYCH, O.; YUVKOVETSKI, O.; KUPERSHTEIN, L. SQL Injection Prevention System. In: *2016 International Conference "Radio Electronics & InfoCommunications" (UkrMiCo)*. Kiev, Ukraine: IEEE, 2016.

YAQOOB, S. M. I. Penetration Testing and Vulnerability Assessment. *Journal of Network Communications and Emerging Technologies (JNCET)*, v. 7, Agosto 2017.

ZHAO, J.; GONG, R. A New Framework of Security Vulnerabilities Detection in PHP Web Application. *9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, p. 271 – 276, 2015.

Anexos

ANEXO A – Entendimento do ambiente TIC e do sistema WEB

A.1 Perguntas propostas para avaliação e entendimento do do ambiente TIC

Tabela 32 – Geral

#	Descrição	Formato da Evidência	Sistema	Responsável	Data de Entrega
1.1	Metodologia e procedimentos para desenvolvimento/manutenção dos aplicativos	-			
1.2	Política de segurança de TI, incluindo normas e procedimentos	-			
1.3	Matriz de segregação de funções	-			
1.4	Planilha de informações Tecnológicas - Anexo Informações Técnológicas	Excel			

Elaborado pelo autor (2018)

Tabela 33 – Gerenciamento de Mudanças

#	Descrição	Formato da Evidência	Sistema	Responsável	Data de Entrega
2.1	Listagem com todos as atualizações instaladas (Apache/ISS/Banco de Dados) no ambiente de produção de dd/mm/aaaa até a data presente	Excel			

#	Descrição	Formato da Evidência	Sistema	Responsável	Data de Entrega
2.2	Impressão da tela evidenciando o método de atualização de correções e mudanças aplicados no ambiente de produção.	Impressão da Tela			
2.3	Listagem com todas as alterações e melhorias no sistema aplicados no ambiente de produção no período de dd/mm/aaaa até a data presente	Excel			
2.4	Impressão da tela evidenciando o método usado para obter a lista atualização do item 2.3	Impressão da tela			
2.5	Impressão da tela do diretório do ambiente de produção, aba “Security” da aplicação em escopo e contendo quais grupos/usuários acessam cada diretório e qual tipo de acessos cada um possui e as respectivas permissões	Impressão da tela			
2.6	Print da tela que evidencie a última alteração instalada no ambiente de produção do sistema	Impressão da tela			
2.7	Evidência (print-screen) que demonstre a efetiva segregação dos ambientes (desenvolvimento, homologação e produção)	Impressão da tela			
2.8	Qual o software de versionamento utilizado (Visual Source Safe, SVC, etc . . .) para controle de versionamento	-			

#	Descrição	Formato da Evidência	Sistema	Responsável	Data de Entrega

Elaborado pelo autor (2018)

Tabela 34 – Acesso Lógico

#	Descrição	Formato da Evidência	Sistema	Responsável	Data da Entrega
3.1	Listagem de funcionários/estagiários em TI admitidos e demitidos de dd/mm/aaaa até a data presente (Necessário conter os campos: “Nome; Matrícula/CPF; Cargo; Departamento; Data de Admissão;Data de Demissão”)	Excel			
3.2	Impressão da tela demonstrando o passo a passo de como a listagem de profissionais admitidos e demitidos foi extraída (de que sistema, onde foram inseridos os parâmetros de data, onde está o botão para exportar relatórios, etc).	DOC			
3.3	Listagem de terceiros contratados e demitidos em TI de dd/mm/aaaa até a data presente. (Necessário conter os campos: “Nome; Cargo; CPF;Departamento; Data de Início;Data de Termino”)	Excel			

#	Descrição	Formato da Evidência	Sistema	Responsável	Data da Entrega
3.4	Impressão da tela demonstrando passo a passo de como a listagem de terceiros admitidos foi extraída (de que sistema, onde foram inseridos os parâmetros de data, onde está o botão para exportar relatórios, etc).	DOC			
3.5	Impressão de tela da parametrização de senhas do sistema	DOC			
3.6	Listagem de funcionários com direito de acesso e/ou liberação de acesso ao Data Center	Excel			
3.7	Listagem dos usuários administradores do sistema com data de criação e status	Excel			

Elaborado pelo autor (2018)

Tabela 35 – Parâmetros de Segurança dos Sistemas Operacionais e Bancos de Dados

#	Descrição	Formato da Entrega	Sistema	Responsável	Data da Entrega
4.1	Lista dos usuários e grupos de usuários cadastrados no servidor de aplicação	Excel/CSV			
4.2	Lista dos usuários e grupo de usuários cadastrados no servidor de banco de dados	Excel/CSV			

#	Descrição	Formato da Entrega	Sistema	Responsável	Data da Entrega
4.3	Forma de autenticação utilizada para o banco de dados (por exemplo: Active Directory, contas locais e etc . . .)	-			

Elaborado pelo autor (2018)

Tabela 36 – Operações de TI

#	Descrição	Formato da Entrega	Sistema	Responsável	Data da Entrega
5.1	Impressão da tela da configuração dos jobs de backup para as aplicações escopo.	Impressão de Tela			
5.2	Topologia do ambiente de rede (apenas que suporte a aplicação)	-			
5.3	Evidência das Regras de Firewall habilitadas para a Aplicação (Produção)	Impressão de Tela			
5.4	Evidência da VLAN existente para o ambiente de Produção	Impressão de Tela			

Elaborado pelo autor (2018)

A.2 Informações do sistema alvo da avaliação

Tabela 37 – Dados da tecnilogia e do ambiente

Item	Descrição
Nome da Aplicação:	
Nome / Diretório Servidor de Desenvolvimento:	

Item	Descrição
Nome / Diretório Servidor de Testes:	
Nome / Diretório Servidor de Produção:	
Sistema Operacional e Versão:	
Banco de Dados e Versão:	
Aplicação Desenvolvida Externamente? (Sim/Não)/Nome da Empresa de Software:	
(Se a resposta anterior for Sim) Customizado?	
Data da implantação:	
Data da última alteração significativa	
Possui possibilidade de Acesso Remoto? (Sim/Não)	
Aplicação processada fora do ambiente da prefeitura? (Sim-Qual/Não)	

Elaborado pelo autor (2018)

ANEXO B – Incidentes de Ataques em Prefeituras divulgadas nos meios de comunicação em 2018

Tabela 38 – Incidentes de Ataques de hackers a Prefeituras em 2018

UF	Cidade	Data	Notícia
RS	Joia	12/01/2018	Hackers invadem site da prefeitura de Jóia e cobram pagamento de 4 mil dólares em bitcoin. https://g1.globo.com/rs/rio-grande-do-sul/noticia/hackers-invadem-site-da-prefeitura-de-joia-e-cobram-pagamento-de-4-mil-dolares-em-bitcoin.ghtml
PI	Teresina	03/02/2018	Prefeitura de Teresina retira sites oficiais do ar após invasão de grupo hacker. Um grupo hacker invadiu e retirou do ar alguns sites da prefeitura Municipal de Teresina. Os endereços tiveram sua página inicial encaminhado para uma página dos hackers que trazia propagandas do grupo. De acordo com Eduardo Aguiar, presidente da Prodater, a invasão foi detectada ainda na noite deste sábado (3) e os sites começaram a ser migrados para uma plataforma mais moderna. https://cidadeverde.com/noticias/265334/prefeitura-de-teresina-retira-sites-oficiais-do-ar-apos-invasao-de-grupo-hacker
MT	Rondonópolis	17/02/2018	Hackers tentam invadir sistema da Prefeitura de Rondonópolis. Ataque de hackers foi registrado na manhã de sábado (17). Desde então, os serviços de informática foram interrompidos e o atendimento ao público está comprometido. https://g1.globo.com/mt/mato-grosso/noticia/hackers-tentam-invadir-sistema-da-prefeitura-de-rondonopolis-mt.ghtml

UF	Cidade	Data	Notícia
SC	São Carlos	05/03/2018	No início deste ano, a prefeitura de São Carlos, no oeste de Santa Catarina, também foi alvo de um ataque cibernético que acabou inviabilizando o trabalho dos servidores municipais e suspendendo o atendimento ao público.Os invasores bloqueram todas as informações do cadastro de moradores, folha de pagamento, contabilidade, compras e tributação. http://www.diariodoiguacu.com.br/noticias/detalhes/prefeitura-de-sao-carlos-tem-servidor-hackeado-40267
SP	Serrana	08/03/2018	A Prefeitura de Serrana sofreu um ataque hacker na manhã desta quarta-feira (7). De acordo com uma nota divulgada pela administração municipal, o ataque foi descoberto quando funcionários chegaram para trabalhar e não conseguiram abrir os sistemas de cobranças de água e esgoto e, também, o de gerenciamento dos repasses para transporte escolar e de alunos que estudam fora do município. Uma mensagem exigia que os responsáveis pela prefeitura entrassem em contato com eles no prazo máximo de 72 horas, sob a ameaça de terem os arquivos completamente destruídos, caso não atendessem ao pedido. Os hackers pedem um resgate em moedas virtuais.Segundo a prefeitura, as informações armazenadas até o mês de janeiro foram recuperadas e as duas máquinas, que abrigavam os sistemas atacados, foram encaminhadas a um especialista na tentativa de recuperar os dados referentes aos meses de fevereiro e março. Por conta desse incidente, documentos como 2º via de boletos e contas de água não estão sendo emitidos. Há a possibilidade que estudantes que se beneficiam do repasse escolar façam um novo cadastro na secretaria da Educação. https://www.acidadeon.com/ribeiraopreto/cotidiano/policia/NOT,0,0,1312267, prefeitura+de+serrana+e+vitima+de+ataque+hacker.aspx

UF	Cidade	Data	Notícia
SC	Concórdia	15/03/2018	Empresa suspenderá prestação de serviço de internet em alguns setores da Prefeitura após ataque de hackers e prejuízo de R$ 2 milhões. O empresário Fiorelo Ruviaro, proprietário da empresa Turbo Net, registrou nas últimas horas um boletim de ocorrência na Delegacia de Polícia informando sobre um ataque hacker. Todo o sistema do provedor foi destruído e técnicos não estão conseguindo recuperar a transmissão nas redes da empresa. http://www.atualfm.com.br/site/empresa-suspendera-prestacao-de-servico-de-internet-em-alguns-setores-da-prefeitura-apos-ataque-de-hackers-e-prejuizo-de-r-2-milhoes/
MG	Ituiutaba	20/03/2018	A Prefeitura de Ituiutaba teve ser servidor centrar invadidos por hackers em 19/03 através de um Ransomware (Programa malicioso que criptografa as informações) e só podem ser recupados através da insersão de uma senha, desta forma os hackers sequestram as informoções e pedem dinheiro em BitCoins para informar a senha. http://pontalemfoco.com.br/tecnologia/veja-detalhes-da-invasao-hacker-aos-sistemas-das-prefeitura-de-ituiutaba/
SE	Aracaju	31/03/2018	Site da Prefeitura de Aracaju é invadido por hackers. Um grupo de hackers invadiu o site da Prefeitura de Aracaju neste sábado (31). Na página principal, apareceu uma mensagem contra os políticos, de forma geral, e contra o presidente da República, Michel Temer. As mensagens, assinadas pelo login D4RKR0N, permaneceram na internet por algumas horas antes que a página fosse retirada do ar por algum tempo enquanto seus responsáveis corrigiam a publicação. http://www.f5news.com.br/cotidiano/site-da-prefeitura-de-aracaju-e-invadido-por-hackers_45880/

UF	Cidade	Data	Notícia
ES	São Gabriel da Palha	09/04/2018	O servidor de informática da Prefeitura de São Gabriel da Palha, região Noroeste do Estado, foi invadido e bloqueado. O invasor exigiu um pagamento na moeda virtual bitcoin para que seja liberado.Os funcionários perceberam que na manhã de segunda-feira (09), os sistemas utilizados na prefeitura não estavam funcionando. Desta forma, descobriram que o servidor havia sido invadido.A prefeita do município, Ceia Ferreira (SD), disse que alguns serviços administrativos estão parados, mas a prefeitura funcionou normalmente. “Alguns serviços como a contabilidade estão parados, mas a prefeitura está funcionando. Será instalado um outro servidor para normalizar a situação”, disse. https://www.gazetaonline.com.br/noticias/norte/2018/04/hacker-invade-servidor-de-prefeitura-no-es-e-pede-resgate-em-bitcoin-1014126577.html
PE	Recife	23/04/2018	Site da Prefeitura do Recife é invadido por hacker. O hacker colocou uma mensagem página da Prefeitura. https://jconline.ne10.uol.com.br/canal/cidades/geral/noticia/2018/04/24/site-da-prefeitura-do-recife-e-invadido-por-hacker-336644.php
BA	Guanambi	23/04/2018	Sites de prefeituras de Guanambi e região são atacados por hackers. Na tarde desta segunda-feira (23), os sites de várias prefeituras da região foram invadidos por hackers. O conteúdo das páginas foi substituído por uma imagem de pessoas com bandeiras do brasil e uma máscara de um personagem do filme V de Vingança, e por uma mensagem pedindo a saída do presidente Michel temer e de sua bancada no congresso.A invasão afetou vários sites com o endereço .gov.ba.br, além da prefeitura de Guanambi, os sites das prefeituras de Candiba, Pindaí, Urandi, Matina, Igaporã, Riacho de Santana, Palmas de Monte Alto, Iuiu, Malhada, entre outros, foram afetados pela invasão. Com uma invasão, os responsáveis pelo sistema de informação retiraram as páginas do ar e o acesso começou a ser restabelecido no meio da noite. O grupo denominado D4RKR0N assumiu a autoria das invasões. http://agenciasertao.com/2018/04/23/sites-de-prefeituras-de-guanambi-e-regiao-sao-atacados-por-hackers/

UF	Cidade	Data	Notícia
SC	Blumenau	02/05/2018	Prefeitura de Blumenau diz que invasão de site não atingiu banco de dados. Hacker não teve acesso a dados importantes do sistema, como cadastros dos contribuintes do IPTU. Essa não é a primeira vez que o site é invadido. Há cerca de um ano o portal foi hackeado. De acordo com Costa, as tentativas são frequentes e algumas vezes os dispositivos de segurança não conseguem inibir a ação. https://omunicipioblumenau.com.br/prefeitura-de-blumenau-diz-que-invasao-de-site-nao-atingiu-banco-de-dados/
MT	Cuiabá	06/05/2018	Um dia depois do ataque cibernético ao site oficial da Universidade Federal de Mato Grosso (UFMT), o alvo dos hackers neste domingo (6) foi o site oficial da Prefeitura de Cuiabá. A invasão ao sistema foi confirmada pela Secretaria de Comunicação, que informou ainda não ter detalhes sobre a ação criminosa. http://www.gazetadigital.com.br/editorias/cidades/site-da-prefeitura-de-cuiaba-e-hackeado-e-sai-fora-do-ar/539090
MG	Belo Horizonte	30/05/2018	O site da Prefeitura de Belo Horizonte foi hackeado na manhã desta quarta-feira (30) e ficou por cerca de cinco horas sem funcionar. Os invasores publicaram na página manifesto contra o governo e em apoio à greve dos caminhoneiros. https://www.hojeemdia.com.br/site-da-prefeitura-de-belo-horizonte-%C3%A9-invadido-por-hackers-em-apoio-a-caminhoneiros-1.625893
RS	Porto Alegre	26/06/2018	O site da prefeitura de Porto Alegre foi hackeado na tarde desta terça-feira. Uma página escura, com dizeres contrários a corrupção no país ficou no ar até por volta das 15h. De acordo com a assessoria de imprensa, o Executivo tomou conhecimento da invasão às 13h e, imediatamente, a equipe da Procempa iniciou os trabalhos para regularizar a página. Assinando a publicação, o perfil identificado como Bruno Santos, disse que o principal objetivo de toda a invasão é mostrar a insegurança em sistemas. https://noticias.r7.com/cidades/correio-do-povo/site-da-prefeitura-de-porto-alegre-e-hackeado-26062018

UF	Cidade	Data	Notícia
PR	Cambira	08/07/2018	O município de Cambira, no Paraná, foi hackeado e teve todas as informações contidas no sistema sequestradas, ocasionando o bloqueio de acesso ao servidor.Com o ataque, serviços como o Portal da Transparência ficaram indisponíveis para a população de cerca de 7 mil habitantes. A invasão ocorreu no dia 8 de julho e até o dia 27 de julho a prefeitura tentava restabelecer os serviços para a população. Até o momento não há informações sobre a autoria do ataque ou se houve pedido de resgate. http://www.cambira.pr.gov.br/comunicado-da-prefeitura-de-cambira/
MT	Vera	20/07/2018	Prefeitura de Vera suspende atendimento nos dias 20 a 23 de julho por invasão de hackers.A Prefeitura de Vera por meio da Secretaria Municipal de Administração comunica que o atendimento ao público estará suspenso nos dias 20 a 23 de julho por conta de uma invasão de hackers no servidor de dados que armazena o sistema da Prefeitura.Por esse motivo, o mesmo estará em manutenção razão pela qual não haverá atendimento ao público na sede da Prefeitura Municipal. As atividades no Executivo Municipal retornam no dia 24 de julho das 07h às 13h. https://www.vera.mt.gov.br/Noticias/Prefeitura-de-vera-suspende-atendimento-nos-dias-20-a-23-de-julho-por-invasao-de-hackers-63/
AM	Manaus	17/08/2018	Sistema do Amazonas ATUAL sofre 8 mil tentativas de intrusão em dois dias. As tentativas de invadir o sistema já haviam aumentado nos últimos dias, mas foram intensificados no fim de semana. https://amazonasatual.com.br/sistema-do-amazonas-atual-sofre-8-mil-tentativas-de-intrusao-em-dois-dias/
SP	Mirandopólis	01/10/2018	O sistema virtual da Prefeitura de Mirandópolis foi invadido na última segunda-feira (01/10) por um ataque denominado “Ransomware da família Combo” e criptografou informações de arquivos existentes, o que irá prejudicar serviços à população e adiar o pagamento dos funcionários que vinha sendo feito sempre no 5º dia útil de cada mês. https://www.mirandopolis.sp.gov.br/noticias/sistema-da-prefeitura-sofre-ataque-virtual-e-pagamento-sera-adiado/

UF	Cidade	Data	Notícia
RS	Catuípe	09/10/2018	O sistema da Prefeitura de Catuípe sofre ataque de hackers, todas as movimentações financeiras e administrativas do município foram comprometidas. https://www.radioprogresso.com.br/sistema-da-prefeitura-de-catuipe-sofre-ataque-de-hackers/
DF	Brasília	08/11/2018	Hackers alegam ter login de Sérgio Banhos, ministro substituto do TSE, e código do sistema Gedai-UE para a urna eletrônica. O TSE (Tribunal Superior Eleitoral) vai investigar se hackers tiveram acesso ao sistema interno da Justiça Eleitoral. Os invasores alegam ter o código-fonte do sistema Gedai-UE, que grava o sistema operacional e a lista de eleitores na urna eletrônica; no entanto, eles não teriam quebrado o sigilo do voto. https://tecnoblog.net/266785/tse-investiga-hackers-justica-eleitoral/

Fonte: Diversos sites de notícias (2018)