UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO

UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

RODRIGO RONNER TERTULINO DA SILVA

MADS-WEB: METODOLOGIA APLICADA AO DESENVOLVIMENTO SEGURO DE APLICAÇÕES WEB

MOSSORÓ – RN

2014

RODRIGO RONNER TERTULINO DA SILVA

MADS-WEB: METODOLOGIA APLICADA AO DESENVOLVIMENTO SEGURO DE APLICAÇÕES WEB

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação – associação ampla entre a Universidade do Estado do Rio Grande do Norte e a Universidade Federal Rural do Semi-Árido, para a obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Rommel Wladimir de Lima – UERN.

Coorientadora: Prof. Dra. Cicilia Raquel Maia Leite – UERN.

MOSSORÓ – RN

2014

Catalogação da Publicação na Fonte.
Universidade do Estado do Rio Grande do Norte.

Silva, Rodrigo Ronner Tertulino da
Mads-web: metodologia aplicada ao desenvolvimento seguro de aplicações web. / Rodrigo Ronner Tertulino da Silva. – Mossoró, RN, 2014.

118 f.
Orientador(a): Prof. Dr. Rommel Wladimir de Lima; Cicilia Raquel Maia Leite

Dissertação (Mestre em Ciência da Computação). Universidade Federal Rural do Semi-Árido. Universidade do Estado do Rio Grande do Norte. Programa de Pós-Graduação em Ciência da Computação.

1. Software – Desenvolvimento – Segurança. 2. WEB – Vulnerabilidade – *Hackers. 3. Segurança da informação*. I. Lima, Rommel Wladimir de. II. Universidade do Estado do Rio Grande do Norte. III.Título.

UERN/BC CDD 005.1

Bibliotecária: Jocelania Marinho Maia de Oliveira CRB 15 / 319

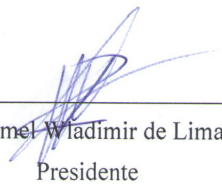
RODRIGO RONNER TERTULINO DA SILVA

MADS-WEB: METODOLOGIA APLICADA AO DESENVOLVIMENTO SEGURO DE APLICAÇÕES WEB

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação para a obtenção do título de Mestre em Ciência da Computação.

APROVADA EM: 28 / 11 / 2014.

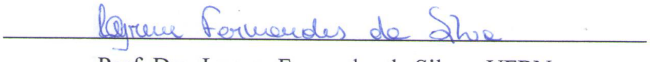
BANCA EXAMINADORA



Prof. Dr. Rommel Wladimir de Lima – UERN
Presidente

Prof. Dra. Cicília Raquel Maia Leite – UERN
Membro Interno

Prof. Dr. Francisco Milton Mendes Neto – UFERSA
Membro Interno



Prof. Dra. Lyrene Fernandes da Silva - UFRN
Membro Externo

Dedico este trabalho especialmente a meu pai, João Vicente Segundo, que veio a falecer quando estava na fase de conclusão deste trabalho. (in memoriam).

AGRADECIMENTOS

Gostaria de fazer um agradecimento especial à minha mãe e ao meu pai, pelos ensinamentos e valores passados a mim, pois com eles aprendi a nunca desistir, independentemente das dificuldades encontradas.

À Universidade Federal Rural do Semiárido (UFERSA) e Universidade do Estado do Rio Grande do Norte (UERN), em particular ao Programa de Pós-Graduação em Ciência da Computação (PPGCC), pela oportunidade de realização desse curso.

Aos professores do mestrado, por toda a dedicação e aprendizado.

Aos meus orientadores Rommel e Cicilia foram pessoas espetaculares, sempre prontos para atender. Além de terem sidos meus professores, ambos na graduação e no mestrado. Considero como amigos, já que nos conhecemos há tanto tempo.

Aos meus familiares, peço desculpas por tantas vezes que não tive como estar presente em ocasiões familiares, mas foi preciso para que esse sonho pudesse se tornar realidade.

Gostaria também de agradecer aos demais colaboradores do programa de pós-graduação em ciências da computação, em especial a Sra. Rosita, sempre pronta para ajudar e atender da melhor forma possível, também ao secretário por parte da UFERSA conhecido como Maninho.

Um agradecimento especial à diretoria do Tempero Regina, que na época autorizou que me ausentasse para assistir as aulas, foram pessoas altamente compreensivas, mesmo não mais fazendo parte da família me sinto na obrigação e dever de deixar isso registrado.

A minha esposa, Rafaela Duarte Borges Tertulino por sempre está presente e ao meu lado, nos momentos mais difíceis da minha vida.

A minha filha, Maria Luiza Borges Tertulino, por fazer parte da minha vida, tudo que faço e por ela.

“Se você acredita que a tecnologia pode resolver seus problemas de segurança, então você não conhece os problemas e nem a tecnologia.”
(Bruce Schneier).

RESUMO

A escalabilidade, portabilidade e fácil acesso providos pela plataforma Web têm popularizado seu uso no desenvolvimento de diversas aplicações. Porém, o crescente número de incidentes de segurança levanta preocupações quanto à sua seguridade. Uma parte destes incidentes decorre da falta de consideração de segurança durante as etapas de desenvolvimento, pois é comum que não sejam utilizadas técnicas para mitigação e prevenção de falhas de segurança no ciclo de vida de desenvolvimento de um *software*. A segurança da informação e o desenvolvimento de sistemas são áreas que neste trabalho estão integradas no ciclo de desenvolvimento de um *software*. Dessa forma, o presente trabalho tem como objetivo implementar uma Metodologia Aplicada ao Desenvolvimento Seguro de Aplicações Web (MADS-WEB), por meio da utilização de práticas de segurança de software ao longo do ciclo de vida do *software*. Neste sentido a metodologia tem como principal intuito orientar equipes de desenvolvimento que procuram fornecer segurança em *software* durante seu ciclo de desenvolvimento. Ainda com o resultado da metodologia MADS-WEB, foram desenvolvidas etapas e fases que proporcionem de forma sistemática uma visão sobre a forma como o *software* deve ser desenvolvido, levando em consideração os aspectos de seguranças durante seu ciclo de desenvolvimento. Como resultados desse processo foram realizados os testes de penetração com o propósito de validar a metodologia, a fim de comprovar e evidenciar falhas existentes que poderiam ter sido mitigadas com a inclusão de atividades de segurança no seu ciclo de desenvolvimento. Para validação da metodologia foi escolhido o Moodle para a realização dos testes pelo fato de ser um *software* Web bastante difundido e utilizado para a educação a distância.

Palavras-Chave: *Hackers*. Segurança. Moodle. Penetração. Vulnerabilidade.

ABSTRACT

Scalability, portability and easy access provided by the Web platform have popularized their use in developing diverse applications. However, the increasing number of security incidents raise concerns about their security. One of these incidents stem from the lack of security consideration during the development stages, it is not that common techniques for prevention and mitigation of security flaws in the software development life cycle of a software to be used. Information security and system development are areas that this work will be integrated in the development cycle of a software. Thus, this paper aims to implement a Methodology Applied to Secure Web Applications development (MADS-WEB), through the use of practical software security throughout the life cycle of software. In this sense the methodology is primarily intended to guide development teams seeking to provide security software during its development cycle. Yet with the methodology results MADS-WEB, steps and stages that provides a systematic way an insight into how software should be developed, taking into consideration the security aspects during the development cycle were developed. As a result of this process the penetration tests in order to validate the methodology were performed in order to confirm and highlight flaws that could have been mitigated with the inclusion of security activities in their development cycle. To validate the methodology was chosen Moodle for the tests because it is a Web software for widespread education and distance used.

Keywords: Hackers. Security. Moodle. Penetration. Vulnerability.

LISTA DE TABELAS

LISTA DE FIGURAS

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
API	*Application Programming Interface*
AVA	Ambiente Virtual de Aprendizagem
BSIMM	*Building Security In Maturity Model*
CASE	*Computer-Aided Software Engineering*
CSRF	*Cross Site Request Forgery*
CSS	*Cascading Style Sheets*
DFD	Diagrama de Fluxo de Dados
DHTML	*Dynamic HyperText Markup Language*
EAD	Educação a Distância
GPL	*General Public License*
HTML	*HyperText Markup Language*
IDE	*Integrated Development Environment*
ISO	*International Organization for Standardization*
ITFS	*Instructional Television Fixed Services*
LDAP	*Lightweight Directory Access Protocol*
MD5	*Message Digest Algorithm*
MSF	*Metasploit Framework*
OpenSAMM	*Open Software Assurance Maturity Model*
OpenSSL	*Open Source Implementation of The SSL and TLS*
OWASP	*Open Web Application Security Project*
PHP	*Personal Home Page*

POC	*Proof of Concept*
PT	*Penetration testing*
PTES	*Penetration Testing Execution Standard*
SDLC	*The Systems Development Life Cycle*
SLA	*Service Level Agreement*
SQL	*Structured Query Language*
SQUARE	*System Quality Requirements Engineering*
SSL	*Secure Socket Layer*
SSO	*Single Sign On*
TIC	Tecnologias de Informação e Comunicação
TLS	*Transport Layer Security*
URL	*Uniform Resource Locator*
UTF	*Unicode Transformation Formats*
W3af	*Web Application Attack and Audit Framework*
WWW	*World, Wide, Web*
XML	*Extensible Markup Language*
XSS	*Cross Site Scripting*

SUMÁRIO

1 INTRODUÇÃO

Com a crescente globalização e com a utilização em larga escala da internet, cada vez mais interativa, os usuários se veem cercados de facilidades no mundo digital. Hoje os sites estão cada vez mais dinâmicos e interativos, gerando, assim, uma troca de informações entre servidores e usuários. É nessa troca de informações que *hackers* podem se aproveitar e acessar informações em servidores Web. Uma aplicação desprotegida se torna alvo fácil para os *hackers* profissionais (JACOBS, 2011).

Os ataques estão cada vez mais utilizando métodos automatizados de exploração de vulnerabilidades. Segundo dados do Cert.br,[1] de janeiro a dezembro de 2013, 24% dos incidentes reportados no Brasil, tinham como foco tentativas de fraudes, 5% dessas tentativas estavam direcionadas a aplicações Web (CERT.BR, 2014).

Neste contexto, recentemente foi descoberta uma falha de segurança chamada *Heartbleed*, por conta de uma brecha no *OpenSSL*[2], diversos sites conhecidos podem ter fornecido informações pessoais para pessoas mal intencionadas (KARAPANOS *et al*. 2014).

Essa falha de segurança permitiu que *hackers* e outros tipos de pessoas mal intencionadas simplesmente observassem a troca de chaves que acontece entre usuário e um determinado site. Isso poderia ser feito diversas vezes, resultando em uma quantidade realmente considerável de dados roubados de todos os tipos (TSOUTSOS *et al.* 2014).

No entanto, verificou-se que esse recurso apresenta uma brecha na sua estrutura, ocasionada por um erro de programação, o pacote foi disponibilizado sem a realização de testes precisos relacionados à segurança. Essa falha durou mais de dois anos até ser descoberta (KARAPANOS *et al*. 2014).

Neste cenário, as vulnerabilidades acontecem por falha de projeto, implantação ou configuração de *software, e estas* quando exploradas por um atacante, ocasiona violação de segurança. Podem também resultar em roubo de dados confidenciais, quebra de integridade de

[1] Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil.
[2] É uma implantação de código aberto dos protocolos SSL e TLS. A biblioteca implementa as funções básicas de criptografia e disponibiliza várias funções utilitárias.

dados ou afetar a disponibilidade. Uma forma de mitigar as vulnerabilidades é fazer uso de uma metodologia para construção de *software* que contemple aspectos de segurança durante todo ciclo de desenvolvimento.

Também como forma de validar a proposta deste trabalho a metodologia foi utilizada em uma aplicação Web, a fim de comprovar e evidenciar falhas existentes que poderiam ter sido mitigadas com a inclusão de atividades de segurança no seu ciclo de desenvolvimento.

1.1 OBJETIVOS

O objetivo da pesquisa é implementar uma Metodologia Aplicada ao Desenvolvimento Seguro de Aplicações Web (MADS-WEB), com a utilização de práticas de segurança de *software* ao longo do seu ciclo de vida, fazendo com que as atividades de segurança sejam contempladas em todo ciclo de vida do *software* desde a sua concepção até a entrega ao usuário final.

A fim de alcançar os objetivos gerais mencionados anteriormente, são traçados os objetivos específicos:

- Identificar metodologias já existentes;
- Detalhar as etapas necessárias para o desenvolvimento de um software seguro;
- Propor uma metodologia de construção de *software*;
- Apresentar uma metodologia que seja útil para as equipes de desenvolvimento de *software*;
- Realizar testes de penetração para evidenciar a aplicabilidade da metodologia em uma aplicação Web.

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado como segue: o Capítulo 2 apresenta os trabalhos relacionados com o tema que fundamenta a dissertação. O Capítulo 3 mostra uma visão geral do referencial teórico, objetivando a compreensão dos conceitos e padrões utilizados pelas organizações para a construção de *software* seguro e os aspectos teóricos para a inclusão de boas práticas de segurança no ciclo de vida do *software*. O Capítulo 4 apresenta a metodologia utilizada no estudo e todas as suas etapas são detalhadas, incluindo a arquitetura proposta e os resultados obtidos através dos Testes de Vulnerabilidades realizados no Moodle. Por fim, o Capítulo 5 apresenta as conclusões e perspectivas futuras deste trabalho.

2 TRABALHOS RELACIONADOS

Viana *et al.* (2013) apresentam um *framework* de alto nível para incorporação das atividades de segurança a partir das primeiras fases do ciclo de vida de uma aplicação (*built-in security*).

A contribuição deste trabalho é um conjunto de recomendações de segurança e uma proposta de um *framework* para o desenvolvimento seguro de aplicações. Este *framework* está fundamentado na necessidade de intensificar as interações entre as equipes de desenvolvimento, arquitetura e administração de dados, reforçando a atenção necessária à segurança desde o início do ciclo de vida de uma aplicação. A figura 1 ilustra o *framework* de alto nível para incorporação das atividades de segurança a partir das primeiras fases do ciclo de vida de uma aplicação.

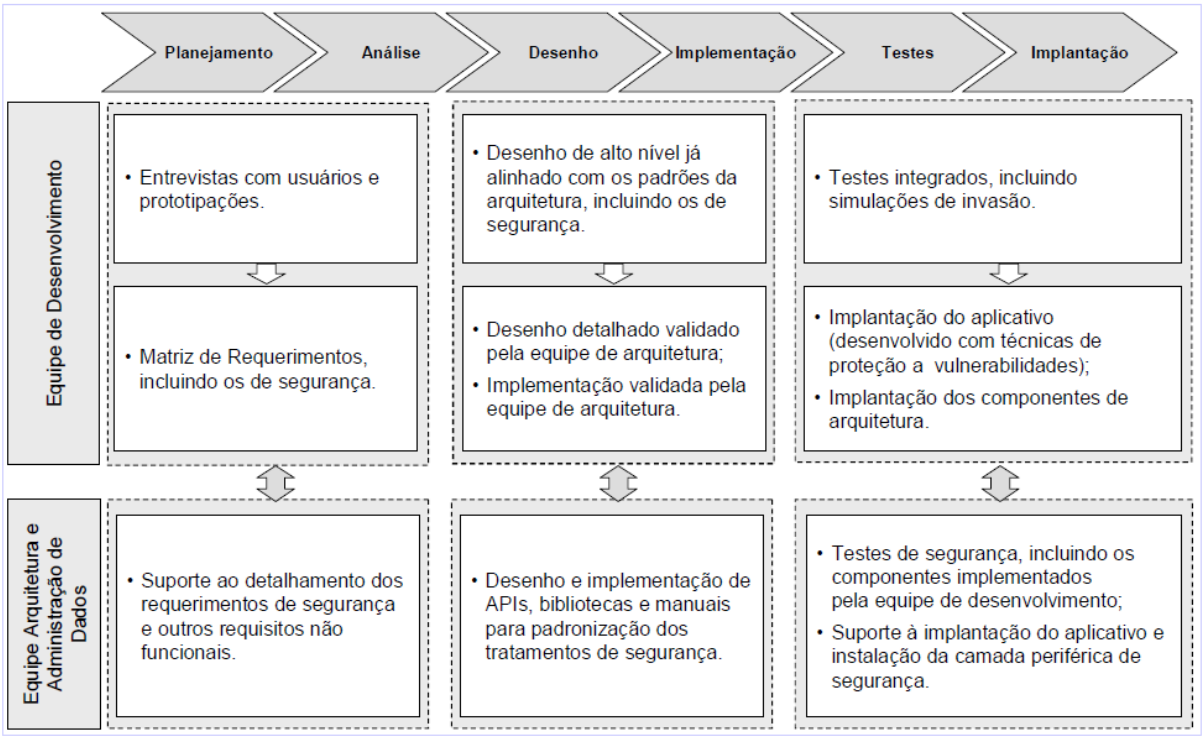


Figura 1 – Framework (*built-in security*).

Fonte: Viana *et al.* (2013).

Esta dissertação em relação ao trabalho apresentado em Viana *et al.* (2013), além de apresentar uma metodologia aplicada à construção de sistemas Web em que todas as fases e etapas são contempladas com aspectos de segurança, detalha e especifica todas as fases e etapas necessárias para a construção de um sistema seguro, ao contrario do citado artigo que

não específica e nem detalha quais os objetivos de cada etapa e não realiza nenhuma validação a fim de evidenciar sua aplicabilidade em sistemas Web existentes. Na dissertação é proposto um conjunto de ferramentas que podem ser utilizadas para a realização de Testes de Vulnerabilidades, o que não foi realizado no artigo em questão.

Aoki *et al.* (2011) propuseram práticas de segurança a serem aplicadas durante o processo de desenvolvimento de *software* Web, tendo como foco o processamento de formulários e sistemas de *login* que minimizam os riscos, aumentando, assim, a qualidade e confiabilidade do produto final. No artigo são apresentados: conceitos de segurança da informação, as vulnerabilidades mais comuns existentes em software Web e algumas práticas que devem ser aplicadas durante o desenvolvimento. O foco no primeiro momento é no processamento de formulários e sistemas de *login*, já que eles são os principais alvos de injeção de códigos e *Cross-site scripting (*XSS*).*

Neste artigo os autores não avaliaram todas as vulnerabilidades conhecidas em aplicações Web, conforme sugere OWASP TOP 10 (OWASP, 2010), eles apenas se detiveram a elencar as vulnerabilidades relacionadas ao *login* dos usuários. Na metodologia proposta nesta dissertação todas as 10 (dez) vulnerabilidades são avaliadas em um sistema Web existente a fim de evidenciar problemas relacionados a todas as vulnerabilidades.

Floyd *et al*. (2012) apresentam algumas vulnerabilidades encontradas no Moodle como: *Session Hijacking Found; XSS Injection; Session Management Flaw(s) e Quiz Engine Flaw(s).* Os testes foram aplicados na versão 2.1.

No capítulo quatro deste trabalho são demonstrados outros problemas descobertos na versão 2.7, além dos já constatados por Floyd *et al*. (2012). Isso faz com que se perceba o quanto é importante tratar de segurança desde a sua concepção, visto que solucionar problemas depois do software pronto pode ser mais dispendioso e complexo. Ao contrário de Floyd *et al*. (2012), que além de elencar tais vulnerabilidades propõem formas de solucionar esses problemas, enfatizam a adoção de uma metodologia baseada em MADS-WEB, proposta apresentada nesta dissertação.

Os trabalhos citados nessa seção abordaram algumas iniciativas dos autores em mitigar vulnerabilidades em aplicações Web, com o propósito de torná-las mais seguras. Tais trabalhos contribuíram para o reconhecimento dos desafios enfrentados com a necessidade de prover segurança em *software*. O foco dessa dissertação é apresentar metodologia MADS-

WEB que contemple aspectos de seguranças em todas as fases do ciclo de vida de um *software*, não contempladas nos trabalhos anteriores. Nesse sentido, o presente trabalho também procurou evidenciar outras falhas que não foram apresentadas nos trabalhos anteriores, além das já descobertas. Assim, são propostas formas de mitigação de vulnerabilidades em face às falhas que foram descobertas com a realização de técnicas de exploração de vulnerabilidades, haja vista não terem sido contempladas nos trabalhos anteriores.

3 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta o referencial teórico utilizado como base para estudo e conceituação dos aspectos que envolvem a construção da metodologia aplicada a esse trabalho. Sendo assim, ele encontra-se organizado da seguinte forma: na Seção 3.1, é apresentada uma visão geral sobre segurança da informação, segurança em *software*, requisitos de segurança relacionado à segurança e desenvolvimento de aplicações seguras, análise de vulnerabilidade e também algumas metodologias aplicadas a testes de penetração. Na seção 3.2 são apresentados aspectos que envolvem os fundamentos, além de modelos de processos comumente utilizados na engenharia de *software*, com ênfase nos conceitos para construção de *software* seguro, requisitos de segurança em software, qualidade de *software*, atividades no ciclo de vida de um *software*, vulnerabilidade em *software* e modelos existentes para maturidade para segurança de *software*. Por fim, a Seção 3.3 apresenta as considerações finais sobre este capítulo.

3.1 SEGURANÇA DA INFORMAÇÃO

A informação é um ativo que necessita ser adequadamente protegido, pois é essencial para os negócios de uma organização. Ela pode existir de diversas formas, especialmente no que se refere aos meios eletrônicos à informação, pois ela está exposta a um grande número de ameaças e vulnerabilidades, visto que cada vez mais os ambientes de negócio estão interconectados (ABNT/ISO/IEC 17799, 2005).

A segurança da informação pode ser definida como a preservação das propriedades de confidencialidade, integridade e disponibilidade da informação. Adicionalmente, outras propriedades como integridade, autenticidade, disponibilidade, não repúdio e confiabilidade podem estar envolvidas (ABNT/ISO/IEC 27001, 2013). A confidencialidade é a garantia de que a informação está protegida contra revelação não autorizada. A integridade é a propriedade pela qual a informação não será modificada indevidamente e a disponibilidade é a

percepção de que a informação estará disponível sempre que necessário. A relação de hierarquia entre a segurança da informação e os seus pilares pode ser observada na figura 2.

Figura 2 - Relação entre a Segurança da Informação e os seus Pilares.

Fonte: Adaptado de ABNT ISO IEC 27001 (2013).

As propriedades da segurança da informação são necessárias para assegurar e proteger a informação dos diversos tipos de ameaça e para auxiliar a continuidade das atividades relativas ao negócio. Para obtê-las, é necessário implementar um conjunto de controles adequados, incluindo politicas, processos, procedimentos e funções de *software*. Para garantir um nível aceitável de segurança da informação são exigidas contínua monitoração, análise e melhoria constante para garantir a segurança da organização e os seus objetivos de negócio (ABNT/ISO/IEC 27001, 2013).

3.1.1 Requisitos de Segurança

Os requisitos de segurança mitigam a capacidade de uma aplicação ficar mais suscetível a possíveis tentativas de ataques e invasões por terceiros. A Associação Brasileira de Normas Técnicas (ABNT) com a norma NBR ISO/IEC 27002:2013 define a análise, avaliação e tratamento de riscos, a legislação vigente e a política de segurança da informação como as três principais fontes de requisitos de segurança da informação de uma organização (ABNT/ISO/EIC 27002, 2013). Também existem alguns requisitos de segurança que estão ligados à qualidade de *software*, dentre eles (PAUL, 2011): confiança; resiliência; recuperabilidade.

Por meio deles são indicadas as principais dificuldades de segurança que uma aplicação deve lidar, por isso, os requisitos de segurança precisam ser considerados durante

todas as fases de construção e desenvolvimento de um *software*. A seguir, serão definidos alguns tipos de requisitos de segurança existentes.

3.1.2 Tipos de Requisitos de Segurança

Os requisitos de segurança devem ser definidos explicitamente e devem corresponder aos objetivos e metas de segurança da organização. Quando os requisitos são apropriadamente definidos e documentados é possível mensurar os objetivos e metas de segurança do projeto, facilitando a implantação e liberação do *software*.

Os requisitos de segurança de *software* podem também estar relacionados aos aspectos de regulação e atenção a normas nacionais e internacionais. Desse modo, a relação de alguns dos tipos de requisitos de segurança de *software*, conforme Paul (2011), pode ser:

- Confidencialidade;
- Integridade;
- Disponibilidade;
- Autenticação;
- Autorização;
- Auditoria.

Na fase de levantamento de requisitos do ciclo de desenvolvimento de *software*, a equipe de segurança de *software* é necessária apenas para identificar quais requisitos são aplicáveis no contexto da organização e para qual funcionalidade do *software* eles devem ser aplicados. Assim, os detalhes de como estes requisitos serão implementados devem ser decididos durante o projeto e desenvolvimento do *software* (PAUL, 2011).

3.1.3 Square - System Quality Requirements Engineering

O SQUARE (*System Quality Requirements Engineering*) é uma metodologia desenvolvida na Universidade de *Carnegie Mellon* [3] para apoiar as organizações a inserirem segurança, desde as primeiras fases da produção do ciclo de vida de *software*. É um processo de nove etapas que ajuda às organizações a construir a segurança, incluindo a privacidade nas fases iniciais do ciclo de vida de produção (GOERTZEL, 2007).

A metodologia é mais eficaz quando conduzida por uma equipe de analistas de requisitos em conjunto com os *stakeholders* do projeto. O SQUARE pode ser decomposto em nove etapas, conforme tabela 1, a seguir.

Tabela 1 - Nove passos para execução do SQUARE Fonte: Adaptado de Goertzel *et al.* (2007).

Etapa	**Entrada**	**Técnica**	**Participante**	**Saída**
Concordar com as definições.	Definição candidata do IEEE e outros padrões.	Entrevistas estruturadas, grupo foco.	*Stakeholders*, equipe de requisitos.	Definições acordadas.
Identificar ativos e objetivos de segurança.	Definições, objetivos candidatos, condutores de negócios, politicas e procedimentos, exemplos.	Sessões de trabalho facilitadas, pesquisas, entrevistas.	*Stakeholders*, analista de requisito.	Ativos e objetivos.
Desenvolver artefatos para suportar as definições de requisito de segurança.	Artefatos potenciais (cenários, modelos, formas, caso de abuso, entre outros).	Sessão de trabalho.	Analista de requisitos.	Artefatos necessários, caso de abuso, cenários, modelos, modelos e formas.
Realizar avaliação de risco.	Caso de abuso, cenários, objetivos de segurança.	Método de avaliação de risco, análise de risco antecipada contra tolerância de risco organizacional, incluindo análise de ameaças.	Analista de requisitos, *stakeholders*, *expert* em risco.	Avaliação de resultado de risco.
Selecionar elicitações técnicas.	Objetivos, definições, técnicas dos candidatos, experiência dos *stakeholders*, estilo.	Sessão de trabalho.	Engenharia de requisitos.	Técnicas de elicitações selecionadas.

[3] A universidade *Carnegie Mellon* é uma instituição privada de ensino e pesquisa, localizada na cidade de *Pittsburgh*, no estado da *Pensilvânia* nos Estados Unidos.

	organizacional, cultura, nível de segurança necessário, análise de custo-benefício, entre outros.			
Elicitar requisitos de segurança.	Artefato, risco, avaliação de resultados, técnicas selecionadas.	*Joint aplication development* (JAD), entrevistas, pesquisas, análise de modelo de base, *checklist*, lista de tipos de requisitos reutilizáveis, revisão de documentos.	*Stakeholders* facilitados por analistas de requisitos.	Corte inicial no requisito de segurança.
Categorizar requisitos pelo nível (sistema, *software*, outros) e quando for requisitos ou outro tipo de restrição.	Requisitos iniciais, arquitetura, avaliação dos resultados.	Sessão de trabalho utilizando categorias padronizadas.	Analista de requisitos, outros especialistas necessários.	Requisitos categorizados.
Priorizar requisitos.	Categorizar requisitos e riscos.	Métodos priorizados como triagem e ganha-ganha.	*Stakeholders* facilitados por analistas de requisitos.	Requisitos prioritários.
Inspecionar requisitos.	Requisitos priorizados, utilizar técnica de inspeção formal.	Método de inspeção, revisões por pares.	Equipe de inspeção.	Requisitos Selecionados, inicialmente, documentação do processo de tomadas de decisões e análise racional.

O SQUARE é organizado nas seguintes etapas:

- **Acordo de definições:** analistas de requisitos e *stakeholders* concordam com definições técnicas que servirão como referência para toda a futura comunicação;
- **Identificação de ativos e objetivos de segurança:** permite a avaliação de consistência entre as políticas organizacionais e o ambiente operacional de segurança;
- **Desenvolvimento de artefatos:** desenvolvimento de artefatos que nortearão as atividades subsequentes. A organização deve produzir objetivos sucintos, documentar o uso normal e os possíveis cenários de ameaças, casos de

utilização e casos de má-utilização, e demais documentos necessários para a definição de requisitos;

- **Realizar Análise de Riscos:** requer um especialista em análise de riscos, que deve recomendar um método específico de análise, baseado nas necessidades da organização. O resultado dessa fase pode auxiliar a identificar exposições de segurança de alta prioridade;
- **Selecionar Técnica de Elicitação:** um método formal de elicitação pode ser utilizado ou entrevistas estruturadas que ajudem a entender as necessidades de segurança dos *stakeholders*;
- **Elicitação de Requisitos de Segurança:** a técnica de elicitação utilizada será um fator decisivo nessa etapa. Os requisitos são construídos baseados na documentação gerada nas etapas anteriores;
- **Categorização de Requisitos:** permite a distinção entre requisitos essenciais, requisitos desejados e limitações de arquitetura. A Categorização também ajuda na priorização das atividades a seguir;
- **Priorização de Requisitos:** envolve a análise do custo benefício de requisitos e a determinação de quais deles são viáveis;
- **Inspeção de Requisitos:** o nível de formalidade pode variar entre inspeções altamente estruturadas ou revisão por pares.

O principal objetivo do SQUARE é propor um modelo para construir conceitos de segurança e qualidade em um estágio no início do ciclo de desenvolvimento. Ele pode ser utilizado para analisar os aspectos da documentação até a qualidade de um projeto. Sendo, assim, um modelo conceitual.

3.1.4 Metodologia para Testes de Penetração

Os Testes de Penetração podem ser definidos como uma tentativa formal (legal e autorizada) de localizar e explorar sistemas de computadores de forma bem-sucedida com o intuito de tornar esses sistemas mais seguros.

O processo inclui sondar as vulnerabilidades, bem como oferecer ataques que funcionem como prova de conceito para demonstrar que eles são reais. Os testes de invasão adequados sempre terminam com recomendações específicas para endereçar e corrigir os problemas descobertos durante os mesmos.

Esse processo como um todo é usado para ajudar a manter as redes e os computadores seguros contra ataques no futuro. A ideia geral consiste em identificar problemas de segurança usando as mesmas ferramentas e técnicas usadas por um invasor. É possível, então, atenuar os riscos identificados por essas descobertas antes que um *hacker* de verdade os explore (ENGEBRETSON, 2013).

Os testes de penetração também são conhecidos como:

- PT (*penetration testing*)
- *Hacking*
- *Hacking* ético
- *Hacking white hat*
- Segurança ofensiva
- *Red teaming* (equipe vermelha)

É importante diferenciar Testes de Penetração e avaliação de vulnerabilidades. Muitas pessoas (e fornecedores) na comunidade de segurança usam esses termos incorretamente de forma intercambiável.

Uma avaliação de vulnerabilidades corresponde ao processo de analisar serviços e sistemas em busca de problemas de segurança em potencial, enquanto um teste de penetração realmente executa explorações de falhas (*exploitation*) e ataques como Prova de Conceito (PoC, ou *Proof of Concept*) para provar a existência de um problema de segurança. Os testes de penetração vão um passo além das avaliações de vulnerabilidades, simulando a atividade de um hacker e enviando *payloads* ativos (BROAD, 2104).

Na próxima subseção será apresentada a área de engenharia de software que tem por objetivos a aplicação de teoria, modelos, formalismos e técnicas para auxiliar na construção de sistemas.

3.2 ENGENHARIA DE SOFTWARE

Nos dias atuais, as tecnologias e práticas abrangem linguagens de programação, bases de dados, ferramentas, plataformas, bibliotecas, padrões e processos. Fundamentos da engenharia de *software* englobam modelos abstratos e precisos para uma correta avaliação e garantia na qualidade do *software*, além de oferecer mecanismos de planejamento e gerenciamento durante todo o processo de análise e desenvolvimento.

Segundo Pressman (2006), a teoria de engenharia de *software* é composta de métodos que ajudam a construir um sistema, onde, estes métodos envolvem um conjunto abrangente de tarefas que incluem: o planejamento e a estimativa do custo, do prazo do desenvolvimento; da análise dos requisitos do sistema; do projeto da estrutura de dados; do projeto da arquitetura e do projeto de procedimentos, com codificação, testes e manutenção do sistema.

A complexidade do sistema de *software* se caracteriza por um conjunto de estruturas de dados e algoritmos, ambos encapsulados em procedimentos, funções, módulos, objetos e agentes interconectados, compondo, desta forma, uma arquitetura de *software*, que é executada pelos sistemas computacionais.

Segundo Paula Filho (2003), as tecnologias e práticas da ciência da computação são aplicadas no desenvolvimento e manutenção de sistemas, voltando-se, também, para o gerenciamento dos projetos na observância da qualidade, produtividade e organização e para as linguagens de programação, dos bancos de dados, das ferramentas, plataformas, bibliotecas, padrões e processos.

Conforme Sommerville (2011), na utilização de modelos abstratos e precisos, os engenheiros criam, analisam, desenvolvem, implementam e mantêm os sistemas de *software*, com a finalidade de avaliar e garantir a qualidade, a base estrutural da engenharia de *software*.

A engenharia de *software* é uma área ampla, interagindo desde o *hardware* e a engenharia de processos até o *software*, numa sequência lógica de práticas, designando o desenvolvimento do *software*, com atividades voltadas para a especificação, para o projeto, para a implantação e para a aprovação, caracterizando-se, assim, o intercâmbio de

mecanismos, ferramentas, métodos e pessoas. E nestas ferramentas pode-se averiguar informação de uma determinada fase do desenvolvimento, com verificação automática em relação a sua consistência com o método, para então disponibilizá-la para outras fases do desenvolvimento. Estas ferramentas além de reduzir o trabalho com criação de diagramas e documentação reduzem, também, o esforço de alterações necessárias no projeto. Isto está em conformidade com o pensamento de Pressman (2006), ao explicar que a engenharia de *software* une métodos e ferramentas, visando um desenvolvimento racional de *software*, gerenciável, com qualidade, numa sequência lógica, precisa e exata.

3.2.1 Os Fundamentos da Engenharia de Software

Paula Filho (2003) ressalta que os fundamentos da engenharia de *software* podem ser entendidos como disciplinas da prática humana, para a interação com máquinas, através de processos, métodos, ferramentas e ambientes, desempenhados por *softwares*, que satisfaçam ambos extremos, seja cliente/usuário, seja fornecedor/cliente, dentro de prazos e custos previstos, entendendo processos (os humanos atuando como máquinas), métodos (os planos de processos) ambientes (as máquinas apoiando os processos e métodos).

O McCall (2006, p. 158) designa as seguintes metas da engenharia de *software*:

a) Manutenção: habilidade para facilitar mudanças, otimização, aperfeiçoamento e progresso;

b) Segurança: habilidade para substituição de *software* quando necessário;

c) Eficiência: habilidade para utilizar os recursos computacionais de espaço e tempo;

d) Aproveitamento: habilidade para o usuário final aproveitar efetivamente e com facilidade o *software.*

Já Pressman (2006, p. 260-262) define os seguintes princípios da engenharia de *software*:

a) Modularidade: partilha e domínio;

b) Encapsulamento: oculta a implantação;

c) Localização: captura dados similares;

d) Abstração: é tudo que não é concreto;

e) Uniformidade: faz tudo parecer similar;

f) Completude: faz tudo que for requerido;

g) Corretude: que o *software* funciona corretamente.

O McCall (2006, p. 162-202) destaca ainda os seguintes fatores para uma efetiva qualidade de *software:*

a) Precisão: o sistema satisfaz dentro das especificações e atende à precisão do cliente;

b) Confiabilidade: o sistema executa todas as funções em conformidade com a precisão;

c) Eficiência: quantidade suficiente de recursos e códigos para o programa realizar sua função;

d) Integridade: controle do acesso restrito ao sistema e dados;

e) Usabilidade: o nível de operacionalidade do sistema;

f) Manutenibilidade: o grau de facilidade na correção de erros;

g) Flexibilidade: a alteração de um programa em fase de operação;

h) Testabilidade: o teste do sistema em relação ao seu desempenho;

i) Portabilidade: a transferência de um sistema para outro hardware ou *software* de sistema para outro;

j) Reusabilidade: o reuso de um programa ou de suas partes em outras aplicações;

k) Interoperabilidade: o acoplamento de um sistema a outro.

Segundo Pressman (2006, pag. 193) alguns fatores estão ligados à produtividade do desenvolvimento de sistemas:

a) Fatores Humanos: a experiência e o número de membros da equipe de desenvolvimento;

b) Fatores da Complexidade do Problema: a complexidade da resolução de um problema e a quantidade de alteração e limitações do projeto;

c) Fatores do Processo: a relação com o método de análise e projeto abordados, a linguagem utilizada e a adoção das ferramentas CASE[4];

d) Fatores dos Produtos: confiabilidade e excelência na performance do hardware e *software* de apoio ao desenvolvimento;

e) Fatores de Recursos: as ferramentas CASE e os recursos de *software* e *hardware*.

3.2.2 Processos de Software

Segundo Pressman (2006), processo de *software* é um "conjunto de atividades que objetivam o desenvolvimento e a evolução de *software*". Esse processo é composto de práticas objetivando o desenvolvimento de sistemas de *software*. Estas práticas se traduzem em atividades de especificação, projeto, implantação e testes, caracterizando-se interação de ferramentas, mecanismos, pessoas e métodos.

Também pode ser denominado de metodologia de desenvolvimento de *software*, devido ao conjunto destas atividades, a análise de requisitos e codificação, e resultados nos quais conjectura a forma como o processo será administrado.

Paula (2006) ainda define as atividades fundamentais, comuns a todos os processos, da seguinte forma:

Especificação de Software: aqui, definem-se as funcionalidades dos requisitos e das restrições do *software*. Nesta fase, o desenvolvedor e o cliente analisam e absorvem as reais

[4] Classificação que abrange todas as ferramentas baseadas em computadores que auxiliam atividades de engenharia de software, desde a análise de requisitos e modelagem até a programação e testes.

necessidades da empresa e definem todas as características do *software* que será desenvolvido;

Projeto e Implantação de Software: nesta fase, o *software* é produzido especificadamente de acordo com o que a empresa necessita, modelos são apresentados por diagramas e praticados na linguagem de programação. Ex: PHP, VisualBasic, DHTML etc;

Validação de *Software*: nesta fase todas as funcionalidades especificadas do *software* são praticadas e garantidas;

Evolução de *Software*: é essencial a evolução ou desenvolvimento para satisfação do cliente.

É interessante notar que é comum as empresas de médio e pequeno porte não adotarem tais processos, por elas não terem recursos disponíveis para isto e, como resultado, falta a sistematização da produção, interferindo na qualidade do produto final, além de atrapalhar a entrega do *software* dentro do que foi estipulado em termos de prazos e custos, inviabilizando, assim, sua evolução. Há, também, organizações que desenvolvem *softwares* sem a utilização de nenhum processo, por estes não se ajustarem as suas realidades.

Existem muitos processos de *softwares*, porém isto não implica em uma regra rígida, pois estes podem se adaptar ou até mesmo novos processos podem ser desenvolvidos, conforme a necessidade da empresa, visando, dessa forma, exterminar o excesso de documentação e burocracia.

3.2.3 Modelos de Processo de Software

Ciclo de vida de desenvolvimento de *software* ou *The Systems Development Life Cycle* (SDLC), ou processo de desenvolvimento de *software* em engenharia de *software*, sistemas de informação e engenharia de *software*, é o processo de criação e manutenção de sistemas de informação, e os modelos e metodologias que as pessoas utilizam para desenvolver esses sistemas. Em engenharia de *software*, o conceito de SDLC é fundamentado em torno das

várias metodologias de desenvolvimento de *software*. Essas metodologias formam a estrutura para planejamento e controle da criação do sistema de informação: o processo de desenvolvimento de *software*. Um desdobramento possível para SDLC é apresentado na figura 3 (GOERTZEL *et al*. 2007):

Figura 3 - Ciclo de Vida de Desenvolvimento de *Software*.

Fonte: Adaptado de Goertzel *et al.* (2007).

Cada etapa de desenvolvimento de *software* tem os seguintes objetivos:

Levantamento das necessidades: também chamado de análise de requisitos, identifica as necessidades de informações da organização;

Análise de alternativas: consiste na identificação e avaliação de sistemas alternativos;

Projeto: trata da construção das especificações detalhadas para o projeto selecionado. Essas especificações incluem o projeto das interfaces, banco de dados, características físicas do sistema, tais como, número, tipos e localizações das estações de trabalho, hardware de processamento, o cabeamento e os dispositivos de rede. Devem-se especificar os procedimentos, a seguir, para testar o sistema completo antes da instalação;

Desenvolvimento: inclui o desenvolvimento ou aquisição do *software*, a provável aquisição do *hardware* e o teste do novo sistema;

Implantação: ocorre após o sistema ter passado satisfatoriamente por testes de aceitação. O sistema é transferido do ambiente de desenvolvimento para o ambiente de produção. O sistema antigo (se existir) deve migrar para o novo;

Manutenção: refere-se a todas as atividades relacionadas a um sistema depois que ele for implementado. Devem-se incluir atividades, tais como, a correção de *software* que não funcione corretamente, a adição de novos recursos aos sistemas em resposta às novas demandas dos usuários.

Existem diversos modelos de SDLC, alguns modelos fazem a junção de desenvolvimento e implantação em uma única etapa. Outros fazem a junção de levantamento e a análise das necessidades também em uma única etapa. Alguns modelos dividem o projeto em projeto lógico e projeto físico.

Sendo assim, para reduzir problemas relacionados às vulnerabilidades dentro de aplicações Web, é essencial indicar atividades de segurança de *software* a serem aplicadas entre as fases do ciclo de vida das aplicações. Estas atividades estão relacionadas à identificação, construção e validação de técnicas que impossibilitem a exploração de vulnerabilidades na operação do *software*.

Com a utilização de uma metodologia especifica para a construção de *software* seguro partindo do seu ciclo de vida, e a partir de atividades que garantam identificar, avaliar, tratar, aplicar e validar controles de segurança da informação espera-se o aumento da qualidade e diminuição máxima das possibilidades de ataque dentro das aplicações. O levantamento e o desenvolvimento de *checklists* com os controles a serem aplicados podem auxiliar a incorporação de práticas de codificação defensiva ao longo da construção de aplicações Web.

O tratamento dos aspectos relacionados à segurança do *software* não implica em dizer que necessariamente representa aumento do custo no seu ciclo de vida de desenvolvimento, tendo em vista que corrigir problemas e falhas de segurança custam mais depois da aplicação pronta e em produção (MCGRAW, 2006). Outros modelos existentes são:

Sequencial ou Cascata: este modelo tem fases diferentes de especificação, de projeto e de desenvolvimento, é o modelo aplicado mais antigo e também o mais amplamente utilizado da engenharia de *software*. Sua abordagem é sistemática e sequencial, onde o resultado de uma fase é a constituição da entrada da fase seguinte, em ordem estrita, podendo haver sobreposição (Pressman, 2006).

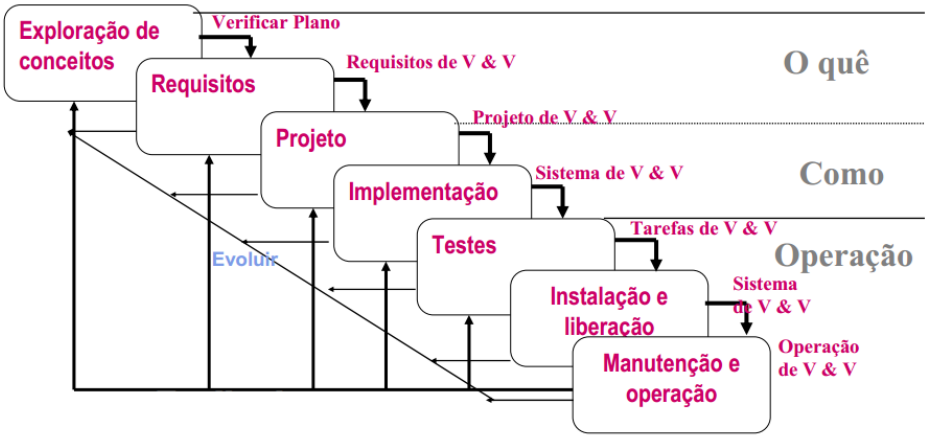


Figura 4 - Modelo Cascata.

Fonte: Adaptado de Pressman (2006)

Rapid Application Development **(RAD):** é um modelo sequencial linear que enfatiza um ciclo de desenvolvimento curto, com construção baseada em componentes. Este modelo é usado, principalmente, em aplicações de sistema de informação e cada função principal pode ser encaminhada para uma equipe RAD em separado, e, então, a partir daí, integrar formando o todo (ENGHOLM, 2010).

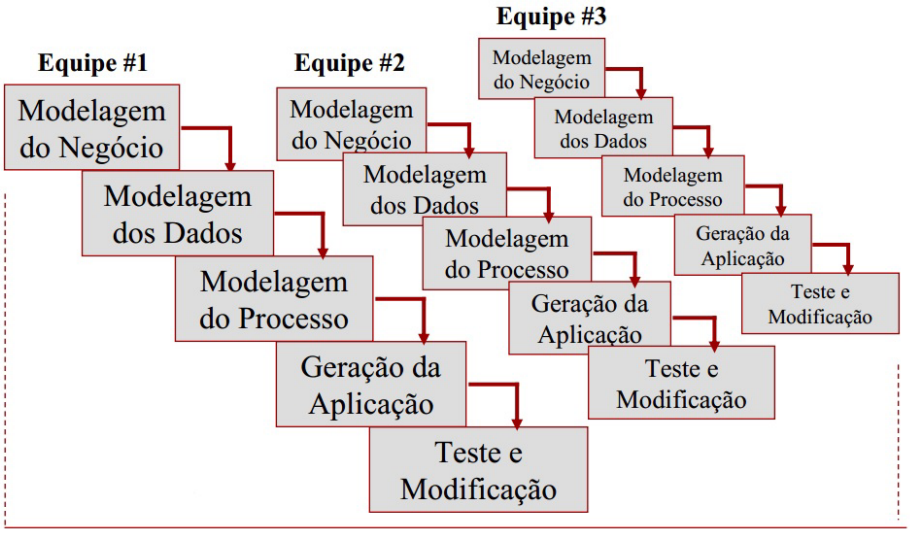


Figura 5 - Modelo RAD.

Fonte: Adaptado de Engholm (2010).

Prototipação: possibilita ao desenvolvedor criar um modelo, um protótipo do software que será construído, para rapidamente efetuar o teste do ambiente de desenvolvimento, com vistas à funcionalidade, performance, interface com banco de dados etc.

Pressman (2006, p. 196-540) explica que este modelo pode ser feito através das seguintes formas:

- Em um protótipo em papel ou mesmo baseado em PC que retratando a interação homem/máquina, permite que o usuário entenda a ocorrência da interação;
- Em um protótipo de trabalho de um programa existente que execute toda a função desejada, apenas para demonstração, uma vez que outras características lhe serão incorporadas para sua melhoria;
- Um protótipo do trabalho, implementando um subconjunto da função exigida do *software* que se deseja.

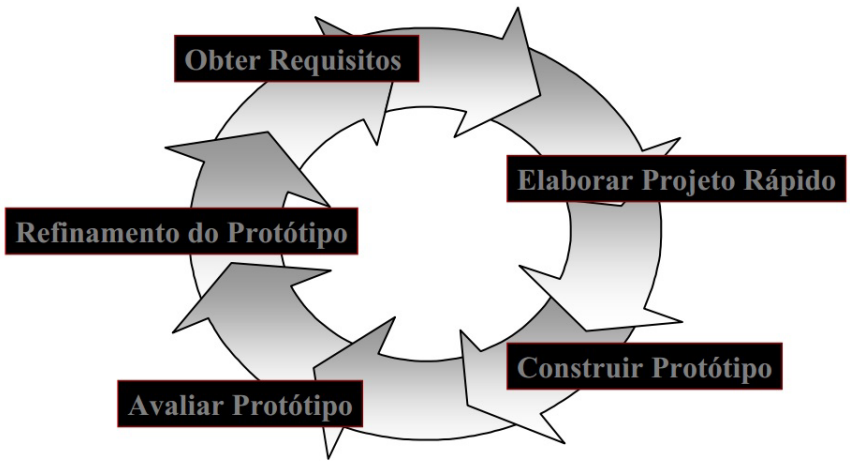


Figura 6 - Modelo Prototipação.

Fonte: Adaptado de Pressman (2006)

Incremental: Sommerville (2009) diz que este modelo incremental, combina elementos do modelo cascata, sequencial e repetidamente, sendo aplicado de maneira iterativa. Ele tem o propósito de trabalhar junto ao usuário para aperfeiçoamento dentro dos requisitos exigidos, até que o produto final seja obtido e aprovado. Inicia-se com o núcleo do produto e evolui com novas características que lhe serão adicionadas por sugestão do usuário.

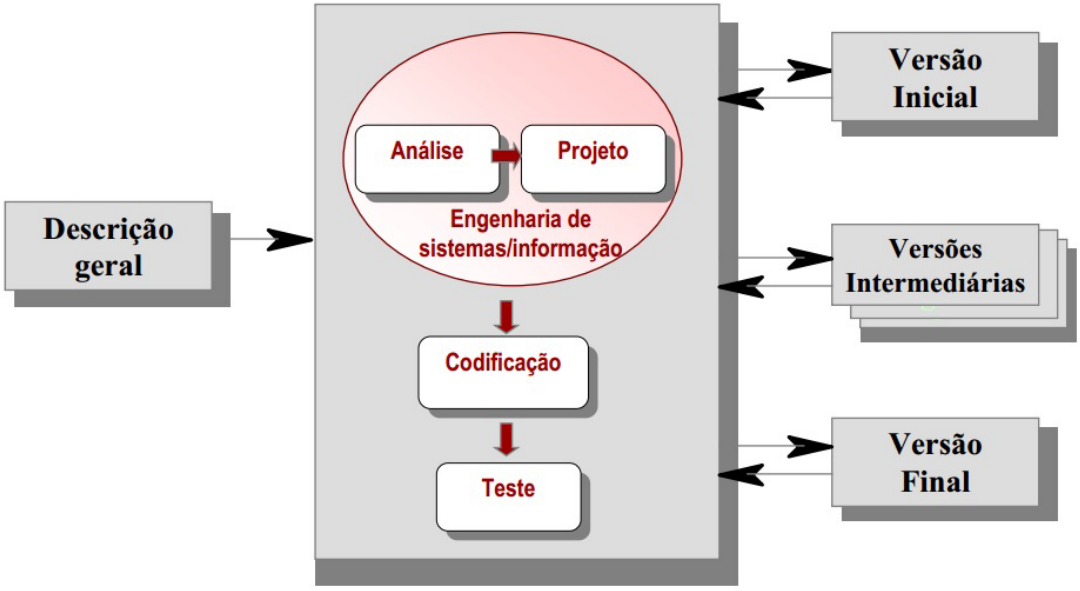


Figura 7- Modelo Incremental.

Fonte: Adaptado de Sommerville (2009).

Espiral: neste modelo ocorre uma evolução através de vários ciclos de especificação, do projeto e do desenvolvimento, sendo melhor utilizado para o desenvolvimento de sistema e de *software* de grande escala. Este modelo integra a iteração do modelo de prototipação com o modelo em cascata e engloba de 3 a 6 regiões de tarefa.

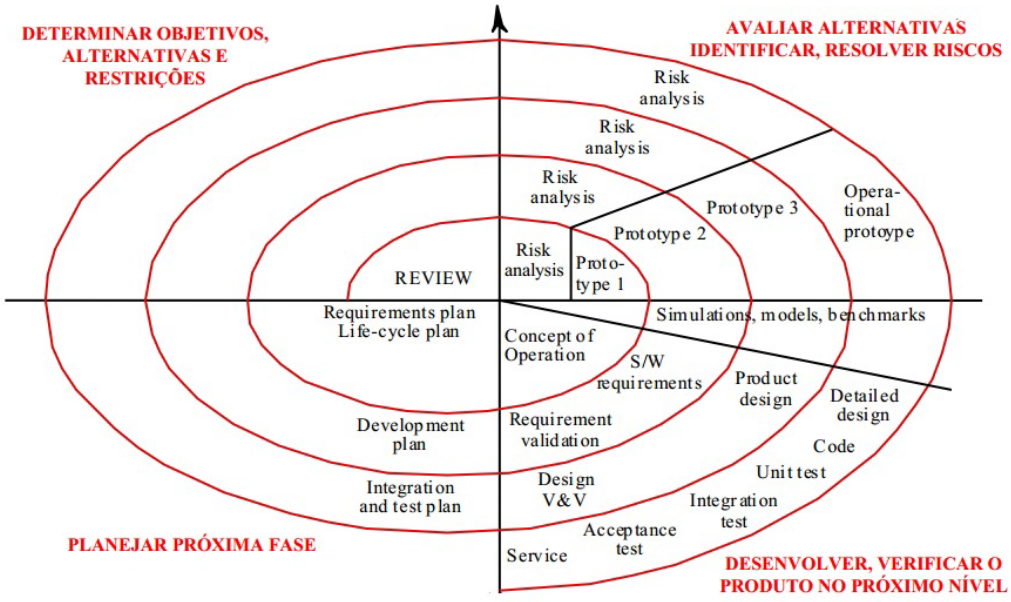


Figura 8 - Modelo Espiral.

Fonte: Adaptado de Pressman (2006).

Conforme Pressman (2006), neste modelo a forma de desenvolvimento do *software* não pode ser completamente prevista para a determinação de todas as suas fases.

Sucintamente, pode-se definir o processo de software como um conjunto de atividades uniformizadas a serem aplicadas sistematicamente e se encontram agrupadas em fases, cada uma delas com os seus intervenientes com responsabilidades, pois possui diversas entradas e produz diversas saídas. Isto é, define quem faz o quê, quando e como, para atingir certo objetivo.

Neste trabalhado, o modelo aplicado foi o SDLC por ser um modelo mais adaptável à metodologia que foi implementada na MADS-WEB, os demais modelos foram demonstrados como forma de apresentar os principais modelos de processos existentes na engenharia de software.

3.2.4 Requisitos de Segurança em Software

Cada aplicação tem os seus requisitos definidos de forma individualizada, pois é importante avaliar a relação custo-benefício para sua implantação. O levantamento dos requisitos de segurança deve acontecer na fase inicial do projeto, ou na construção de novas funcionalidades, para que possam ser avaliados e levados em consideração durante todas as fases seguintes do ciclo de desenvolvimento do *software*.

Na análise de requisito são as seguintes etapas que devem ser seguidas pelo desenvolvedor para concepção de um sistema Pressman (2006): concepção, levantamento, elaboração, negociação, especificação, validação e gestão de requisitos. Já Sommerville (2011) descreve apenas quatro passos para a engenharia de requisitos: elicitação de requisitos, análise de requisitos, validação de requisitos e gerenciamento de requisitos.

Para que a análise de requisito seja melhor desenvolvida, foram criados dois tipo de requisitos que devem ser levados em consideração para a absorção das necessidades do sistema frente aos s*takeholders*, um denominado de requisitos funcionais e outro denominado de requisitos não funcionais. Na análise de requisitos funcionais são levadas em consideração

necessidades que são em nível de usuário. Assim, pode-se explicitar o que o sistema não deve fazer ou ainda requisitos funcionais dos usuários. Estes podem ser: declarações de alto nível a respeito do que o sistema deve fazer.

Os requisitos não funcionais estão relacionados com as funções em nível de sistema, como desempenho e segurança. Restrições aos serviços ou funções oferecidas pelo sistema, tais como, restrições de tempo, restrições no processo de desenvolvimento, padrões, etc. Muitas vezes se aplica ao sistema como um todo ao invés de características individuais ou serviços.

Esses requisitos definem as propriedades e as restrições do sistema, por exemplo, confiabilidade, tempo de resposta e ocupação de área. As restrições são capacidades de dispositivos de E/S, as representações do sistema, dentre outros.

Os aspectos funcionais descrevem comportamentos que viabilizam a criação ou a manutenção da segurança e, geralmente, podem ser testados diretamente. Na maioria dos casos, remetem a mecanismos de segurança, por exemplo, controle de acesso baseado em papéis de usuários (administradores, usuários comuns, entre outros.), autenticação com o uso de credenciais (usuário e senha, certificados digitais, entre outros.), além de outros mecanismos.

Os aspectos não funcionais descrevem procedimentos necessários para que o *software* permaneça executando suas funções adequadamente, mesmo sob uso indevido. São exemplos de requisitos não funcionais: validação de dados de entrada e registro de eventos.

A elicitação de requisitos de segurança de *software* consiste na definição das necessidades de proteção exigidas pelo *software*. Tal atividade exige uma colaboração intensa entre os interessados no software, especialmente daqueles com visão negocial, que podem ter consciência das consequências no negócio decorrentes de incidentes de segurança, cujo vetor de ataque se localize no *software*.

3.2.5 *Software Assurance*

O termo *software assurance* até então estava mais relacionado a duas propriedades de *software*: qualidade e confiança. Nos últimos anos passaram a ser adotados para expressar a

ideia de garantia da segurança em um *software*. Quando comparado à garantia de segurança da informação, a qual é expressa pelo termo garantia da informação (GOERTZEL, 2007).

Todas as definições de *software assurance* transmitem a ideia de que este deve fornecer um nível aceitável de confiança de que o software funcionará corretamente, de forma previsível e de maneira consistente com seus requisitos documentados. Portanto, a função do *software* não deve ser comprometida, mesmo durante ataques diretos ou sabotagem.

Uma forma de definir melhor o termo *software assurance* seria que esta é a base para obter a confiança de que o *software* irá exibir, consistentemente, todas as propriedades necessárias para assegurar que o *software* em operação continuará a operar, apesar da presença de falhas intencionais (PRESSMAN, 2006). Em termos práticos, tal *software* deve ser capaz de resistir a danos, além de recuperar seu nível normal de operação o quanto antes e ser capaz de resistir e de tolerar qualquer ataque.

3.2.6 Atividades no Ciclo de Vida do Software Seguro

Uma maneira para melhorar a segurança e a qualidade de um *software* é realizar atividades de segurança através do ciclo de vida do *software*. Estas atividades que são, na verdade, formas de mitigar problemas quando se trata de segurança em *software*, são responsáveis por lidar com as preocupações de segurança e precisam ser aplicadas dentro das fases do ciclo de vida da aplicação em vez de fazê-lo apenas na fase de requisitos. Uma correlação entre as fases do ciclo de vida e as atividades pode ser visualizada na figura 9, elas foram descritas, de forma geral, para projetos de *software* por McGraw (2006).

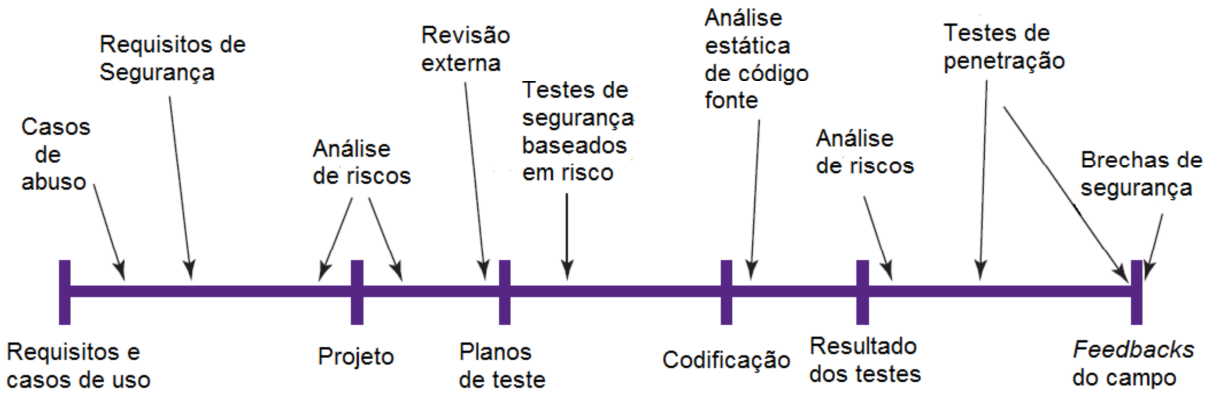


Figura 9 - Atividades de Segurança no Ciclo de Vida do *Software.*

Fonte: Adaptado de MCGRAW (2006).

A seguir, uma breve descrição das atividades a serem realizadas ao longo do ciclo de vida (MCGRAW, 2006):

Casos de abuso: construir casos de abuso é relevante para realizar uma relação entre os problemas e a análise de risco. É importante observar neste momento se algum padrão de ataque se encaixa no sistema ou nos requisitos do *software*. Este é um bom momento para modelar cenários de vulnerabilidades que podem ser explorados nas fases de revisão de código ou teste de penetração;

Requisitos de segurança: os requisitos de segurança precisam cobrir os requisitos funcionais de segurança. Nesta fase toda a necessidade de segurança do *software* precisa ser mapeada para garantir sua correta implantação. Um bom exemplo de requisito de segurança está relacionado com o uso correto de criptografia para proteger dados críticos;

Análise de risco: esta análise está relacionada com o levantamento dos riscos de segurança que podem estar presentes na arquitetura da aplicação que será construída e é uma pequena parte do processo de gerenciamento de risco que todo desenvolvedor precisa levar em consideração;

Revisão externa: avaliação e testes externos desempenham um papel fundamental. Análise externa (ou seja, a análise por alguém de fora da equipe de projeto) se faz necessária, quando se trata de segurança. A segurança do *software* é melhor analisada por pessoas que não estão envolvidas no projeto original nem na implantação do sistema;

Teste de segurança baseado em riscos: a estratégia de testes da aplicação precisa cobrir pelo menos dois tópicos principais, que são os testes dos requisitos de segurança com

técnicas de teste para requisitos funcionais e testes de segurança baseados em riscos, levantados pelos casos de abuso e pela avaliação dos padrões de ataque;

Análise estática de código fonte: após a fase de codificação e antes da fase de testes, a análise de código fonte é uma boa atividade para garantir que os requisitos de segurança foram bem implementados e que as vulnerabilidades listadas na análise de casos de abuso não estão presentes no *software*. A revisão de código pode ser automática e manual e cada estratégia tem prós e contras. Ferramentas automatizadas não cobrem todos os cenários, por este motivo a análise manual é sempre necessária (LONG *et al*, 2012);

Testes de penetração: este é um conjunto de técnicas e ferramentas utilizadas em conjunto para testar dinamicamente um *software* ou sistema contra falhas de projeto ou vulnerabilidades. Esta atividade é importante para garantir que a aplicação ou sua infraestrutura não possui nenhum problema potencial que pode ser explorado de uma forma particular para alterar o comportamento da aplicação em tempo de execução (MICROSOFT, 2008);

Brecha de segurança: é uma falha de segurança que pode ocasionar acesso não autorizado de dados, aplicativos, serviços, redes e/ou dispositivos, ignorando os seus mecanismos de segurança subjacentes.

Nesse sentido, ressalta-se que as atividades de segurança descritas podem ser aplicadas a qualquer tipo de ciclo de vida de *software*, bem como independem do modelo de desenvolvimento de *software* adotado pelos desenvolvedores. Estas atividades não têm nenhuma ligação direta com o modelo de desenvolvimento de *software*, apesar da sua eficácia na melhoria da qualidade das aplicações.

Para fomentar a implantação de boas práticas para construção de um *software*, pode-se estabelecer a criação de um grupo de segurança, formado por analistas desenvolvedores que possuem experiência real em desenvolvimento de aplicações com ênfase em segurança. Esta é uma maneira de garantir que essas atividades serão acompanhadas por pessoas com conhecimento técnico especializado no desenvolvimento de software seguro (BSIMM, 2012).

3.2.7 Vulnerabilidades em Software

Vulnerabilidades existentes no *software* são problemas inerentes ao seu processo de construção. Entre outras, essas vulnerabilidades podem estar relacionadas com o levantamento incorreto de requisitos de software funcionais e não funcionais em especial, aqueles relacionados às questões de segurança; lacunas no planejamento do projeto; falhas no processo de garantia de qualidade e teste, bem como, a falta de controles adequados dentro do ciclo de desenvolvimento de *software*.

O levantamento de riscos durante o *design* do aplicativo é fundamental para garantir que as principais vulnerabilidades serão mitigadas de forma adequada durante a fase de codificação. Mesmo se todos os controles, verificações e procedimentos relativos à segurança são alcançados e aplicados, ainda não existe garantia de que outras vulnerabilidades não possam acontecer.

Os problemas relacionados às vulnerabilidades em *software* também podem estar relacionados a uma má codificação por parte do desenvolvedor ou até mesmo por desconhecimento na linguagem que está sendo usada para codificação, sendo essencial a criação ou formulação para desenvolvimento de aplicações orientadas à segurança, visto que procurará mitigar esses problemas, ao passo que o próprio desenvolvedor terá que analisar meticulosamente seu código ou pelo menos preocupar-se em não cometer erros imerecidos.

Uma visão inocente pode diminuir a capacidade crítica de observar que as falhas são aspectos que vão além dos defeitos (*bugs*) e que as vulnerabilidades relacionadas com o risco de segurança não podem ser ignoradas.

3.2.8 Modelos de Maturidade para Segurança de Software

Um modelo de maturidade é uma representação simplificada do mundo e contém elementos essenciais para a construção de processos efetivos. Modelos de maturidade focam o

melhoramento dos processos em uma organização. Eles contêm os elementos essenciais para a construção de processos efetivos para uma ou mais disciplinas e descrevem um caminho de melhoria evolucionária que vai desde processos *ad-hoc*, seguindo por processos imaturos, disciplinados e finalizando nos processos maduros com qualidade e efetividade melhorada (SEI, 2010).

Os modelos de maturidade proveem orientação para ser utilizada no desenvolvimento de processos efetivos, porém, não são processos ou descrições de processos. O processo adotado em uma organização depende de inúmeros fatores, incluindo seus domínios de aplicações e a estrutura e o tamanho da organização. Em particular, as áreas de processo de um modelo de maturidade não mapeiam um a um os processos usados na organização como um todo (SEI, 2010). Existem dois modelos bastante difundidos que são: OpenSAMM e BSIMM, que serão demonstrados na próxima subseção.

3.2.8.1 OpenSAMM

O OpenSAMM *(Open Software Assurance Maturity Mode*l) ou (Modelo de Maturidade para qualidade de Software Aberto) é um *framework* aberto e foi projetado para ser flexível para auxiliar organizações a desenvolver e implementar suas estratégias de segurança de *software,* que são adaptadas aos riscos enfrentados pela organização. Os recursos oferecidos pelo OpenSAMM irão ajudar em (OPENSAMM, 2014):

- Avaliar as práticas de segurança de *software* existentes na organização;
- Construir e equilibrar o programa de garantia de segurança de *software* em interações bem definidas;
- Demonstrar melhoramentos concretos no programa de garantia de segurança;
- Definir e mensurar atividades relacionadas à segurança por toda organização.

OpenSAMM foi concebida com flexibilidade, de forma que pode ser utilizada por pequenas, médias ou grandes organizações e que façam uso de qualquer estilo de

desenvolvimento. Esse modelo pode ser aplicado por toda organização, para uma única linha de negócios ou para um projeto individual.

Além dessas características, OpenSAMM baseia-se nos princípios a seguir (CHANDRA, 2009):

- O comportamento da organização se modifica lentamente no decorrer do tempo. Um *software* de segurança bem sucedido deve ser especificado em pequenas interações que entreguem ganhos de garantias tangíveis, enquanto trabalha para objetivos de longo prazo;
- Não há uma única receita que funcione para todas as organizações. O *framework* de segurança de *software* deve ser flexível e permitir que as organizações formem suas escolhas baseadas em sua tolerância de riscos e na direção na qual se constrói e usam o *software*;
- Orientação relacionada a atividades seguras devem ser prescritivas. Todos esses passos na construção e avaliação em um programa de garantia de qualidade devem ser simples, bem definido e mesurável.

Esse modelo oferece modelos de roteiros para tipos comuns de organizações. A função do modelo é construída sobre o núcleo das funções do negócio de desenvolvimento de *software* com práticas seguras associadas a cada função. A construção do modelo é baseada em três níveis de maturidade definidos para cada uma das suas 12 (doze) práticas de segurança (CHANDRA, 2009).

3.2.8.2 BSIMM

O BSIMM *(Building Security In Maturity Model*) ou (Modelo de Maturidade para Construção Segura) é um estudo de iniciativas de segurança de *software*. O principal objetivo é ajudar ao *software* seguro a realizar e medir suas próprias iniciativas. Não se trata de um guia de como fazer, é um reflexo do estado da arte do *software* seguro aplicado às organizações (BSIMM, 2014).

O trabalho com o modelo de maturidade mostra que mensurar a iniciativa de segurança de uma organização é possível, e é extremamente útil. As mensurações podem ser utilizadas para planejar, estruturar e executar as evoluções de uma iniciativa de segurança de *software*.

O BSIMM pode ser usado por alguém responsável por criar e executar iniciativas de segurança de *software* e traz a confiança do conhecimento das melhores práticas sobre *software* seguro para estabelecer um *framework* de segurança de *software*.

Durante a criação do modelo foi conduzida uma série de entrevistas com executivos responsáveis por 09 (nove) iniciativas de *software* seguro. Nestas entrevistas, foram identificadas atividades em comum às 09 (nove) iniciativas avaliadas, atividades estas que compõem o *framework* de segurança de software do BSIMM (MCGRAW, 2012).

A fim de validar o trabalho, foi solicitado para cada participante revisar o *framework*, as práticas, e os cartões com as pontuações que foram criadas para suas iniciativas. As 51 (cinquenta e uma) organizações participantes foram retiradas de 12 (doze) diferentes setores: serviços financeiros, vendedores independentes de *software*, empresas de tecnologia, computação nas nuvens, mídia, segurança, telecomunicações, seguros, energia, varejo, saúde e provedores de internet.

Na média, os participantes tinham prática com segurança de *software* por aproximadamente 6 (seis) anos. Todas as 51 (cinquenta e uma) empresas concordaram que o sucesso de seus programas ocorreu devido ao seu grupo de segurança de *software* (MCGRAW, 2012).

O BSIMM foi criado como iniciativa para prover *software* seguro, neste sentido fornece recursos para organizações que procuram constituir ou melhorar suas próprias iniciativas no que diz respeito a este tipo de *software.*

No geral, qualquer iniciativa de *software* seguro é criada com alguns objetivos em mente. O BSIMM é apropriado se os objetivos do seu negócio para segurança de *software* incluem (MCGRAW, 2012):

- Decisões de gerenciamento de risco informadas;
- Clareza no que é a coisa certa a fazer para todos os envolvidos em segurança de *software*;
- Melhoria da qualidade de código.

3.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste capítulo, de forma geral, foram apresentados os principais conceitos utilizados como base para o desenvolvimento deste trabalho. Desde princípio sobre o que é segurança da informação, assim como aspectos relacionados à segurança em *software*, Foram abordados os requisitos de segurança, depois foi visto o funcionamento das atividades de análise de vulnerabilidade e explanado o funcionamento de como são realizados testes de vulnerabilidade em aplicações e seus aspectos mais comuns.

Ainda neste capítulo foram vistos aspectos de engenharia de *software*, desde os fundamentos da engenharia, dentre eles, modelos de desenvolvimento, assim como o SDLC que serviu como base na construção da metodologia nesse trabalho de dissertação. Também foi realizada uma análise sobre requisitos de *softwares* no contexto de engenharia de *software* que difere dos requisitos de segurança no contexto de segurança, também foram demonstrados dois modelos de maturidade, esses modelos focam no melhoramento dos processos em uma organização, são eles: OpenSAMM e BSIMM.

4 MADS-WEB: METODOLOGIA APLICADA AO DESENVOLVIMENTO SEGURO DE APLICAÇÕES WEB

Este capítulo apresenta a metodologia MADS-WEB desenvolvida neste trabalho. O capítulo foi dividido em 4 (quatro) seções: na seção 4.1 é feita uma contextualização da proposta deste trabalho, com ênfase na necessidade de se prover segurança em aplicações desde o início do seu desenvolvimento; na seção 4.2 é apresentada a especificação da metodologia proposta através da explicação das fases e a interação dessas fases entre si; na seção 4.3 é demonstrada a aplicabilidade de cada fase e o seu relacionamento com as demais fases; também são apresentados os resultados dos Testes de Vulnerabilidades, assim como a análise dos resultados; por fim, na seção 4.4 são abordadas as considerações finais sobre este capítulo.

4.1 CONTEXTUALIZAÇÃO

O objetivo deste trabalho é desenvolver um modelo *systems development life cycle* (SDLC) ou ciclo de vida do desenvolvimento de sistemas, denominado MADS-WEB, que se trata de uma metodologia para construção de *software* seguro voltada para aplicações Web, através da utilização de práticas de segurança de *software* ao longo do seu ciclo de vida.

Neste sentido, para a construção da metodologia implementada nesse trabalho, foi utilizada como forma de subsidiar a construção da MADS-WEB, o modelo de SDLC Clássico apresentado nas seções anteriores, o modelo apresentado por McGraw (2006) e a metodologia apresentada por Engebretson (2013).

Um modelo SDLC é de certa forma uma proposta de como e quais etapas devem ser envolvidas na construção de um sistema, existem vários tipos de SDLC, cada empresa pode adotar o seu. Neste trabalho foi construído um modelo SDLC que contempla segurança, desde as etapas inicias da construção de um *software* até a forma de mitigação de riscos.

A partir do modelo proposto por McGraw (2006), em que são sugeridas atividades e uma correlação entre as fases do ciclo de vida e as atividades de segurança, dentre estas, Casos de Abusos, Requisitos de Segurança e Análise de Risco. Estas etapas são o ponto de partida para a construção de um *software* seguro, pois nelas é essencial que os desenvolvedores possam ter uma ideia mais específica sobre as necessidades de segurança do sistema em comparação com as funcionalidades que o mesmo deverá prover aos usuários.

Neste sentido, na fase final do modelo proposto por McGraw (2006), é sugerido que a equipe de segurança realize Testes de Penetração com o objetivo de validar se o sistema é realmente seguro em face aos casos de abuso reportados e se os problemas de segurança foram mitigados.

Diante do exposto a metodologia não contempla por completo um modelo de desenvolvimento de *software*, apenas elenca etapas que deveriam ser incluídas em modelos já existentes para desenvolvimento de sistemas. Partindo dessa ideia a MADS-WEB inclui as atividades da fase inicial que são: Requisitos de Segurança, Casos de abuso e Análise de risco, assim como o Teste de Penetração na fase final do modelo proposto pelo McGraw (2006).

Também como forma de subsidiar a construção da metodologia usada neste trabalho, a metodologia para Teste de Penetração apresentada por Engebretson (2013), que tem o objetivo de incluir processos que possam ser utilizados para sondar vulnerabilidades em sistemas, por meio de técnicas de reconhecimento, *scanning*, exploração de falhas, mas que não fornecem um processo formal. Todos estes fatores foram importantes para definir a metodologia empregada na MADS-WEB, na terceira fase são utilizados aspectos como Testes de Penetração para análise de segurança, com o objetivo de explorar falhas em sistema em pleno funcionamento e também em sistemas que ainda estão em fase de Desenvolvimento.

A figura 10 ilustra as metodologias utilizadas como referência para construção do modelo SDLC MADS-WEB. Algumas etapas nas metodologias existentes foram suprimidas por serem contempladas em etapas utilizadas no modelo proposto. Com base, nessa ideia, pode-se afirmar que a MADS-WEB é uma evolução das metodologias existentes, contemplando a segurança dentro de todas as etapas do ciclo de vida do *software*.

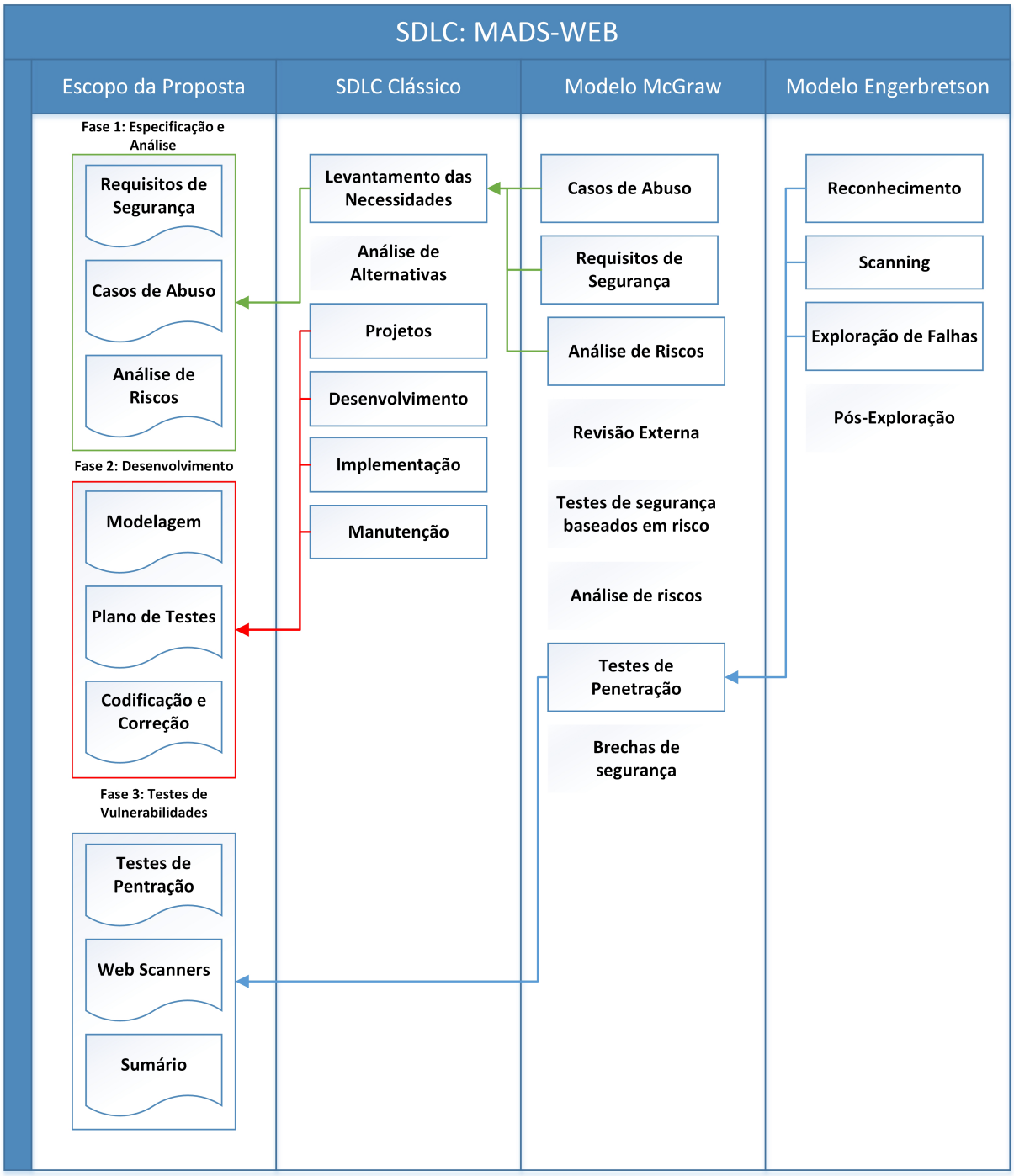


Figura 10 - Relacionamento Entre as Metodologias

4.2 ESPECIFICAÇÃO DA MADS-WEB

A MADS-WEB é dividida em três fases. 1) Especificação e Análise; 2) Desenvolvimento; 3) Testes de Vulnerabilidades. As próximas subseções apresentam as especificações para cada uma delas. A figura 11 demonstra a distribuição das fases na MADS-WEB.

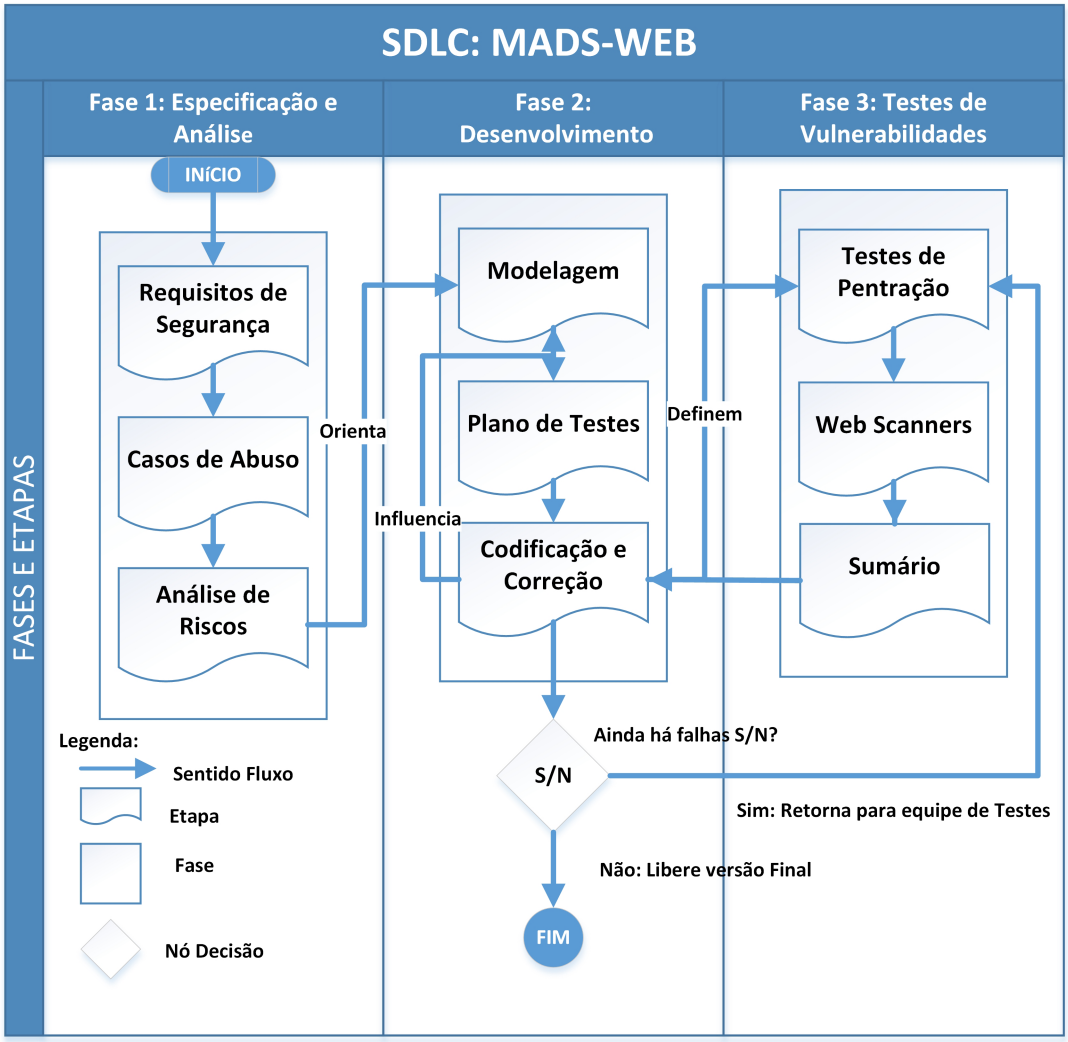


Figura 11 - Distribuição das Fases na MADS-WEB.

4.2.1 Especificação e Análise

Na primeira fase, Especificação e Análise, inicia-se o desenvolvimento do *software*. Essa fase engloba todas as tarefas que lidam com investigação, definição e escopo para construção de um sistema de atividades, como vulnerabilidades relacionadas à segurança e à confidencialidade que são pertinentes à aplicação. Elas devem ser identificadas para que sejam analisados as perdas e os impactos causados pelo comprometimento da confidencialidade das informações ou roubo das informações restritas a uma organização. Para isso, a primeira fase da MADS-WEB é composta por três etapas: Requisitos de Segurança, Casos de Abuso e Análise de Risco.

4.2.1.1 Requisitos de Segurança

A elicitação ou levantamento de requisitos é a fase inicial de qualquer projeto, sendo de extrema importância para seu sucesso, já que antes de iniciar qualquer ideia é preciso entender com clareza todos os objetivos e restrições envolvidos, não apenas para o correto planejamento, estudo de viabilidade ou estimativas de custos, mas para entregar ao usuário final um sistema que atenda as suas expectativas. Desta forma, torna-se fundamental para todos os projetos o entendimento e o controle correto desses requisitos para que os mesmos possam atender ao objetivo determinado.

Na MADS-WEB, o levantamento dos requisitos corresponde à definição dos Requisitos de Segurança que devem ser definidos explicitamente, e precisa corresponder aos objetivos e metas de segurança da organização. Quando os requisitos são apropriadamente definidos e documentados, é possível mensurar o escopo de segurança do projeto, facilitando a implantação e a liberação do *software*.

Como forma de levantamento dos Requisitos de Segurança para aplicações Web, nesse trabalho é proposto o uso do *Common Criteria for Information Technology Security Evaluation*, ou *Common Criteria* que foi desenvolvido por diversos órgãos de países distintos, com o objetivo de avaliar a segurança da tecnologia da informação (ISO/EIC 15408, 2009).

A metodologia MADS-WEB faz uso da segunda parte da norma em que são definidos os requisitos funcionais de segurança que devem ser levados em consideração na definição dos Requisitos de Segurança.

Dentro da norma, os requisitos funcionais de segurança são expressos em classes, famílias e componentes. As classes utilizadas na aplicação da metodologia são as seguintes: auditoria de segurança, comunicação, suporte à criptografia, proteção de dados do usuário, política de controle de acesso e identificação e autenticação.

A auditoria de segurança envolve o reconhecimento, gravação, armazenamento e análise de informações relacionadas com a segurança. Os registros de auditoria resultantes podem ser examinados para determinar quais atividades de segurança relevantes ocorreram e quem é o responsável por elas. As seguintes ações devem ser auditáveis e devem ser incluídas pelo menos uma, conforme prevê a norma:

a) Registro de ações que envolvam deleção, alteração e adição do grupo de usuários com permissão de acesso à leitura aos registros de auditoria;
b) Registros dos direitos para ver/modificar os eventos de auditoria;
c) Registros dos parâmetros que controlam o armazenamento de auditoria;
d) Registro de falhas devido a algum problema no armazenamento da auditoria;
e) Serviço de não repúdio.

Os requisitos de comunicação se preocupam essencialmente na troca de dados. Neste requisito o conceito de informação é usado devendo ser interpretado como objetos que estão sendo comunicados, assegurando a identidade do autor da informação transmitida (prova de origem) e a identidade do destinatário das informações transmitidas (comprovante de recebimento). Assim, o autor não pode negar ter enviado a mensagem, nem o destinatário negar o recebimento, ou seja, o não repúdio.

O suporte criptográfico deve fazer uso das funcionalidades de criptografia para satisfazer objetivos de segurança de alto nível. Estes incluem (mas não estão limitados a):

a) Identificação e autenticação;

b) Gerenciamento de segurança;

c) Privacidade;

d) Proteção das funcionalidades de segurança;

e) Utilização dos recursos de chaves públicas;

f) Uso de canais e caminhos confiáveis.

Os requisitos de proteção do usuário estão divididos em três partes:

a) Políticas de função do usuário de segurança de proteção de dados:

   1. Política de controle de acesso;
   2. Política de controle de fluxo de informação.

b) Formas de proteção de dados do usuário:

   1. Funções de controle de acesso;
   2. Funções de controle de fluxo de informação;
   3. Residual de proteção de informações;
   4. *Rollback*;
   5. Integridade dos dados armazenados.

c) Armazenamento *off-line*, importação e exportação:

   1. Autenticação de dados;
   2. Exportação dos dados.

A política de controle de acesso define o âmbito do controle das políticas que formam a parte de controle de acesso identificado, gerenciando os atributos usados para tornar o acesso explícito ou negação com base em politicas de segurança.

A identificação e autenticação definem requisitos para funções que estabelecem e verificam a identidade de um usuário em questão e devem ser capaz de:

1. Falhas de Autenticação: definir valores para tentativas sem sucesso de autenticação e ações;

2. Atributos de Usuário: definir requisitos para vinculação de atributos de segurança válidos;

3. Especificação de Senhas: definir requisitos para mecanismos que definem a quantidade da métrica de senhas e formas de satisfazer essa métrica;

4. Identificação do Usuário: define os tipos de autenticação suportados pelos usuários.

4.2.1.2 Casos de Abuso

Após a análise e definição dos Requisitos de Segurança, ou seja, aqueles que especificam os requisitos mínimos de segurança que uma aplicação Web deve prover, devem ser elaborados os Casos de Abuso. Os Casos de Abuso ligam vulnerabilidades conhecidas aos riscos que a aplicação possa ser submetida. Um Caso de Abuso é definido a partir de quais explorações uma aplicação pode sofrer, tendo em vista os riscos que ela possui.

Como forma de análise das vulnerabilidades existentes que envolvem aplicações Web, a metodologia MADS-WEB faz uso do OWASP *TOP* 10 para definir os riscos pertinentes que uma aplicação Web pode sofrer. Todas as 10 (dez) vulnerabilidades elencadas devem ser avaliadas na Terceira Fase, onde serão feitos os Testes de Vulnerabilidades, conforme define a metodologia em questão. Em seguida, a Análise de Risco determinará os riscos aos quais a aplicação é suscetível.

De acordo com estudos recentes da OWASP *(Open Web Application Security Project)*, é possível destacar um conjunto de 10 (dez) casos de abusos resumidos, conforme tabela 2, estes podem se tornar possíveis vulnerabilidades em aplicações Web (HAROLD, 2010).

Tabela 2 - Casos de Abuso Reconhecidos em Aplicações Web Fonte: Adaptado de Harold (2010).

Casos de Abuso	**Risco**
Injeção de Código	Ocorrem quando dados não confiáveis são enviados para um interpretador como parte de um comando ou consulta. Os dados manipulados pelo atacante podem iludir o interpretador para que este execute comandos indesejados ou permita o acesso a dados não autorizados.
Quebra de Autenticação e Gerenciamento de Sessão	Ocorre quando as funções da aplicação relacionadas à autenticação e gerenciamento de sessão são implementados de forma incorreta, permitindo que os atacantes comprometam senhas, chaves e *tokens* de sessão ou explorem outra falha da implantação para assumir a identidade de outros usuários.
Cross-Site Scripting (XSS)	Falhas XSS permitem aos atacantes executarem *scripts* no navegador da vítima.
Referência Insegura e Direta a Objetos	Ocorre quando um programador expõe uma referência à implantação interna de um objeto, como um arquivo, diretório, ou registro da base de dados.
Configuração Incorreta de Segurança	Uma boa segurança exige a definição de uma configuração segura em todos os níveis.
Exposição de Dados Sensíveis	Ocorre quando as aplicações não protegem devidamente os dados sensíveis, tais como cartões de crédito, IDs fiscais e credenciais de autenticação.
Falta de Função para Controle do Nível de Acesso	Ocorre quando as aplicações não verificam os direitos de acesso, em nível de função, antes de tornar essa funcionalidade visível na interface do usuário.

Cross-Site Request Forgery (CSRF)	Ocorre quando o navegador da vítima é forçado a executar uma ação maliciosa em favor do atacante.
Utilização de Componentes Vulneráveis Conhecidos	Componentes, tais como bibliotecas, *frameworks*, e outros módulos de *software* quase sempre são executados com privilégios elevados.
Redirecionamentos e Encaminhamentos Inválidos	Aplicações frequentemente redirecionam e encaminham usuários para outras páginas ou sites, e usam dados não confiáveis para determinar as páginas de destino.

Para verificar as vulnerabilidades, quando se trata de uma aplicação Web, todos os casos de abusos, relacionados na tabela 2, devem ser examinados.

Na subseção, a seguir, serão demonstradas as atividades pertinentes à etapa relacionada à Análise de Risco.

4.2.1.3 Análise de Risco

A terceira etapa da fase de Especificação e Análise, denominada Análise de Risco, envolve identificar perdas e impactos causados pelo comprometimento da confidencialidade das informações ou roubo das informações restritas a uma organização. Os riscos podem estar relacionados à distribuição de dados de importância crítica e à perda de integridade das informações. Essa etapa da metodologia estará diretamente ligada à modelagem do sistema, pois a partir desta etapa ele será definido ou modelado de acordo com as necessidades de segurança para cada organização.

Na MADS-WEB, a Análise de Risco é formada por quatro etapas, seguindo a proposta de Pressman (2006):

- **Etapa 1 - Identificação dos riscos:** a identificação dos riscos inclui determinar quais os riscos podem afetar o projeto e documentar suas características;
- **Etapa 2 - Projeção/Estimativas dos riscos:** a projeção ou estimativa dos riscos busca classificar cada ameaça, com a probabilidade dela acontecer, prevendo, assim, as consequências do seu acontecimento;
- **Etapa 3 - Administração dos riscos:** desenvolvimento de opções e ações para aumentar as oportunidades e reduzir as vulnerabilidades encontradas na aplicação;

- **Etapa 4 - Monitoramento dos Riscos:** acompanhamento dos riscos identificados, monitoramento dos riscos residuais, identificação dos novos riscos, execução de planos de respostas a riscos e avaliação da sua eficácia durante todo o ciclo de vida da aplicação.

Quando uma organização trabalha com uma infraestrutura na internet, se faz necessário tomar uma decisão de como os riscos devem ser tratados. É praticamente impossível eliminar todas as ameaças pelas quais uma organização pode passar. Neste sentido, faz-se necessário mitigar os riscos aos quais a aplicação está suscetível. Logo em seguida a Análise de Risco, inicia-se a fase de Desenvolvimento.

4.2.2 Desenvolvimento

A Segunda Fase, Desenvolvimento, é o momento onde parte das questões levantadas na fase de Especificação e Análise devem ser avaliadas e mitigadas. A Análise de Riscos, Requisitos de Segurança e os Casos de Abusos levantados na primeira fase, influenciam a etapa de Modelagem, assim como o Plano de Testes. Esta etapa, também chamada de implantação, é a mais central do processo de engenharia de software, pois é quando o *software* é efetivamente construído. Para isso, a Segunda Fase da MADS-WEB também é composta por três etapas: Modelagem, Plano de Testes e Codificação e Correção.

4.2.2.1 Modelagem

Um modelo é uma simplificação da realidade, ele é criado para facilitar o entendimento de sistemas complexos. Estes modelos podem abranger planos detalhados e também planos mais gerais com uma visão panorâmica do sistema (SOMMERVILLE, 2011).

Todos os sistemas podem ser descritos sob diferentes aspectos, com a utilização de modelos distintos, onde cada modelo será, portanto, uma abstração específica do sistema. Os modelos podem ser estruturais, dando ênfase à organização do sistema, ou podem ser comportamentais, dando ênfase à dinâmica do sistema. Há quatro objetivos principais para se criar modelos (SOMMERVILLE, 2011):

1. Ajudam a visualizar o sistema como ele é ou como deseja que ele seja;

2. Permitem especificar a estrutura ou o comportamento de um sistema;

3. Proporcionam um guia para a análise do sistema;

4. Documentam as decisões tomadas no projeto.

Através dos modelos, é possível obter múltiplas visões do sistema, particionando a complexidade do sistema para facilitar sua compreensão, e atuando como meio de comunicação entre os participantes do projeto. Portanto, uma linguagem de modelagem padronizada, tal como a UML[5], é fundamental para a construção e o entendimento de bons modelos.

No contexto da Modelagem MADS-WEB uma das múltiplas visões a serem oferecidas é a modelagem de ameaças. Para isso, o primeiro passo do processo de modelo de ameaças da metodologia é desenvolver uma representação visual das ameaças sob a forma de um diagrama de fluxo[6].

Para evitar erros é importante fornecer um modelo estruturado para modelagem, mas é importante compreender que esta etapa representa o fluxo de dados e não a codificação. Este é um erro muito comum por desenvolvedores, porque eles só pensam em escrever, sem antes modelar para primeiro entender a complexidade do *software* e as suas ameaças (RANSOME, 2013). Os principais passos envolvidos na modelagem de ameaças são:

1. Dividir a arquitetura de *software* utilizando diagramas de fluxo de dados;

2. Criar categorias com ameaças, identificando quais ameaças são aplicáveis para cada elemento em seu diagrama de fluxo de dados;

[5] Permite que desenvolvedores visualizem os produtos de seus trabalhos em diagramas padronizados.
[6] É uma representação gráfica do "fluxo" de dados através de um sistema de informação, modelando seus aspectos de processo.

3. Mapear todas as ameaças com vulnerabilidades pertinentes aplicáveis no contexto do cenário de uso;

4. Criar um *ranking* de ameaças atribuindo uma classificação de risco para cada ameaça e vulnerabilidade de forma a compreender o impacto e definir prioridades;

5. Definir um plano de mitigação com contramedidas para cada uma das vulnerabilidades identificadas;

6. Corrigir as vulnerabilidades que não são aceitáveis para o negócio.

O Diagrama de Fluxo de Dados (DFD) na figura 12 apresenta um modelo de ameaças ao qual uma aplicação Web está suscetível. O DFD proporciona uma visão mais ampla das ameaças. Neste sentido, não é possível fornecer um modelo padrão na MADS-WEB para todas as ameaças, visto que cada aplicação tem suas particularidades e as ameaças que estão suscetíveis dependem das funcionalidades que a mesma possui.

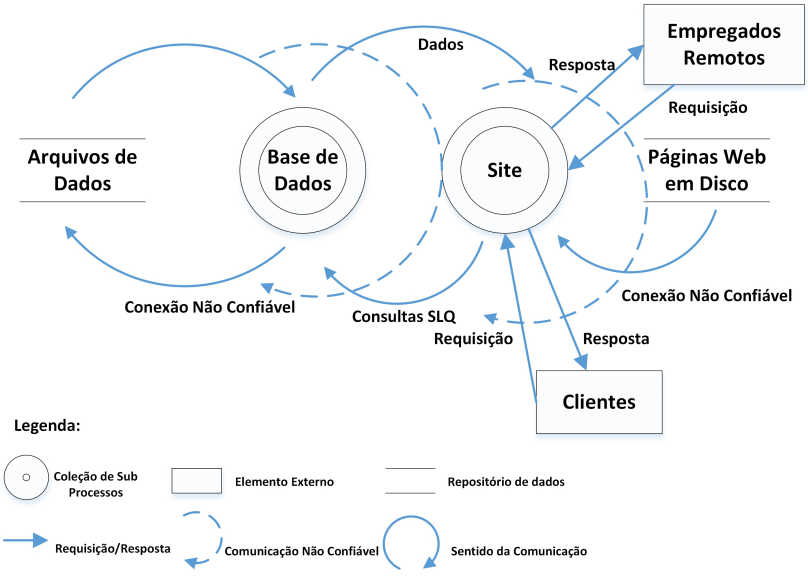


Figura 12 - Exemplo de Diagrama de Fluxo de Dados para a Modelagem de Ameaças Fonte: Adaptado de Ransome (2013).

Criar um DFD é a chave para obter um modelo de ameaça. É importante garantir que todas as peças do sistema estejam representadas. Cada um dos elementos (processos, arquivos de dados, fluxos de dados e usuários), conforme demonstrado na figura 12, ela modela as ameaças que são pertinentes ao *software* que se deseja desenvolver. Uma vez que o DFD seja concluído, será proporcionada uma visão geral de como os dados são processados pelo

software, incluindo a forma como ele se move. É o que acontece dentro do aplicativo e outros que podem estar associados ao *software*.

Na MADS-WEB a modelagem pode ser modificada, caso seja necessário realizar alguma correção no código, devido à descoberta de algum problema, por exemplo, através da realização dos Testes de Penetração poderão ser detectadas falhas em face aos resultados contidos nos sumários, tornando-se necessário que se realize a remodelação da aplicação para atender às correções.

4.2.2.2 Plano de Teste

A realização do Plano de Testes é uma das atividades do processo de desenvolvimento de sistema de *software* que visa executar um conjunto de ações de modo sistemático com o objetivo de encontrar falhas no processo de desenvolvimento do software.

O Plano de Teste é o conjunto de tarefas que são realizadas no processo de construção de uma aplicação. Ele proporciona a realização de testes que por sua vez validam a segurança de uma aplicação, ao mesmo tempo em que reduz a probabilidade de erros quanto à segurança sobre o produto que está sendo desenvolvido, antes que as falhas de segurança sejam descobertas por clientes ou usuários mal intencionados.

Neste sentido, se faz necessária a realização de testes em face aos Requisitos de Segurança pertinentes à aplicação a qual a mesma está suscetível, conforme levantamento realizado na etapa de Análise de Risco na primeira fase da metodologia. Exemplos de requisitos a serem testados são: desempenho; segurança; interface de usuário; controle de acesso e funcionalidades. Esses requisitos procuram garantir a competência da aplicação a partir de uma perspectiva de segurança, demonstrada através dos testes e seus artefatos, relatórios e ferramentas.

O objetivo da realização de testes não é testar a insegurança, mas sim, validar a robustez e a segurança da aplicação antes de tornar o produto disponível para os clientes.

O Plano de Teste pode ser elaborado pelo gerente de projeto ou gerente de testes. Esse plano visa planejar as atividades a ser realizadas, definir os métodos a ser empregados, planejar a capacidade necessária, estabelecer métricas e formas de acompanhamento do processo. Nesse sentido, ele deve conter:

1. Os itens a serem testados: o escopo e objetivos do plano devem ser estabelecidos no início do projeto;

2. Atividades e recursos a serem empregados: as estratégias de testes e recursos utilizados devem ser definidos, bem como toda e qualquer restrição imposta sobre as atividades e/ou recursos;

3. Os tipos de testes a serem realizados e ferramentas empregadas: os tipos de testes e a ordem cronológica de sua ocorrência são estabelecidos no plano;

4. Critérios para avaliar os resultados obtidos: métricas devem ser definidas para acompanhar os resultados alcançados.

O planejamento é necessário a fim de antecipar o que pode ocorrer e, portanto, provisionar os recursos necessários nos momentos adequados. Isto significa coordenar o processo de teste de modo a perseguir a meta de qualidade do produto (sistema de *software*).

A tabela 3 apresenta uma relação dos itens considerados imprescindíveis para um Plano de Teste.

Tabela 3 - Relação de Itens de um Plano de Teste.

Itens de um Plano de Teste	**Conteúdo**
1. Introdução	Contém uma identificação do projeto, descrição dos objetivos do documento, o público ao qual ele se destina e escopo do projeto a ser desenvolvido. Pode adicionalmente conter termos e abreviações usadas, além de informar como o plano deve evoluir.
2. Requisitos a serem testados	Descreve em linhas gerais o conjunto de requisitos a serem testados no projeto a ser desenvolvido, comunicando o que deve ser verificado.
3. Estratégias e ferramentas de teste	Apresenta um conjunto de tipos de testes a serem realizados, respectivas técnicas empregadas e critério de finalização de teste. Além disso, é listado o conjunto de ferramentas utilizadas.
4. Equipe e infraestrutura	Contém descrição da equipe e da infraestrutura utilizada para o desenvolvimento das atividades de testes, incluindo: pessoal, equipamentos, software de apoio, materiais, dentre outros. Visa garantir uma estrutura adequada para a execução das atividades de testes previstas no plano.
5. Cronograma de atividades	Contém uma descrição de marcos importantes das atividades (incluindo as datas de início e fim da atividade).
6. Documentação complementar	Apresenta-se uma relação dos documentos pertinentes ao projeto.

O Plano de Testes é imprescindível no processo de construção de um sistema, sendo que na metodologia MADS-WEB os testes têm como propósito a validação das funcionalidades pertinentes à aplicação. Não sendo dispensada a realização dos Testes de Penetração a serem realizados na Terceira Fase da metodologia, que tem como objetivo avaliar se todas as medidas de segurança realizadas foram cumpridas e se os riscos foram mitigados, evidenciando os problemas, caso existam.

4.2.2.3 Codificação

Na etapa da Codificação e Correção uma equipe de programadores é designada para trabalhar no *software*. Diagramas gerados na etapa de Modelagem são passados para equipe de desenvolvimento que irá transformá-los em código de uma linguagem de programação qualquer. Normalmente, as pessoas envolvidas nesse processo são divididas em dois grupos, um responsável pela implantação da interface gráfica e outro pela implantação da lógica da aplicação (SOMMERVILLE, 2011). Além disso, boa parte da estrutura básica da aplicação pode ser exportada diretamente dos diagramas em código, dependendo das ferramentas que estiverem sendo usadas pela equipe.

Na metodologia MADS-WEB não há como definir a especificação de como deverá ser realizada a codificação, visto que dependendo das funcionalidades pertinentes da aplicação a ser desenvolvida, poderá contar com uma ou mais linguagens de programação diferentes, de modo que cada aplicação possui particularidades que somente a equipe de programação definirá após a modelagem do sistema.

Portanto, a escolha fica a cargo da equipe que realizará a codificação do sistema. Ainda neste sentido, é importante ressaltar que o domínio da linguagem escolhida é um fator determinante para o sucesso da aplicação quanto à segurança do sistema, quanto mais domínio sobre uma determinada linguagem a equipe possuir, melhor será desenvolvido o *software*. Problemas como estouro de pilhas, inteiros e formato de *strings* são vulnerabilidades extremamente sérias para programas escritos em linguagem C ou C++ (RANSOME, 2013).

Na metodologia MADS-WEB como medida preventiva algumas recomendações são sugeridas como práticas gerais de codificação:

- Utilizar sempre código testado, gerenciado e aprovado em vez de criar código novo, não gerenciado, para tarefas comuns;
- Utilizar *Application Programming Interface* ou Interface de Programação de Aplicações (API) que insiram tarefas específicas para ser realizadas através do sistema operacional. Não permitir que a aplicação execute comandos diretamente no sistema operacional, especialmente através da utilização de *shells* de comando iniciados pela aplicação;
- Fazer uso de mecanismo de verificação de integridade por *checksum* ou *hash* para verificar a integridade do código interpretado, bibliotecas, arquivos executáveis e arquivos de configuração;
- Evitar requisições simultâneas para a aplicação ou utilizar um mecanismo de sincronização para evitar condições de disputa (*race conditions*), fazendo uso de mecanismos de *lock*;
- Proteger as variáveis compartilhadas e recursos contra acessos concorrentes inapropriados;
- Restringir os usuários de gerar um novo código ou alterar o código existente;
- Revisar todas as aplicações secundárias, códigos e bibliotecas de terceiros para determinar a necessidade do negócio e validar as funcionalidades de segurança, uma vez que estas podem introduzir novas vulnerabilidades;
- Identificar todas as fontes de dados e classificar as fontes como confiável/não confiável. Em seguida, validar os dados provenientes de fontes não confiáveis;
- As rotinas de validação de dados de entrada devem ser centralizadas na aplicação;
- Especificar um conjunto de caracteres apropriados, como UTF-8, para todas as fontes de entrada de dados;
- Codificar os dados para um conjunto de caracteres comuns antes da validação;
- Quando há falha de validação a aplicação deve rejeitar os dados fornecidos;

- Determinar se o sistema suporta conjuntos de caracteres estendidos UTF-8 e, em caso afirmativo, validar após efetuar a decodificação UTF-8;
- Validar todos os dados provenientes dos clientes antes do processamento, incluindo todos os parâmetros, campos de formulário, conteúdos das URLs e cabeçalhos HTTP, por exemplo: nomes e valores dos *cookies*. Certificar-se também de incluir automaticamente mecanismos de *postback*[7] nos trechos de código *JavaScript*, *Flash* ou qualquer outro código incorporado.

Após o *software* ser codificado deverá ser disponibilizada uma versão do sistema para que na Terceira Fase a equipe responsável pelos Testes de Penetração possa realizar análise de vulnerabilidade com base nos Casos de Abuso reportados na Primeira Fase.

Assim, a equipe de desenvolvimento de posse dos Sumários dos Testes de Penetração deverá realizar as correções, caso tenham sido sugeridas, retornando para que sejam realizados novos testes a fim de descobrir se a aplicação ainda continua com a vulnerabilidade, caso não haja mais problema, deve-se liberar a versão final para usuário.

A correção como atividade incluída nesta etapa se faz necessária à medida que a equipe de responsáveis pelos Testes de Penetração reporta os erros através dos sumários, logo em seguida, a equipe de desenvolvimento deve realizar as correções necessárias.

Neste sentido, como citado nas subseções anteriores, caso as correções impliquem em alterações criticas do ponto de vista da equipe, deverá ser realizada a remodelação da aplicação, necessitando que a equipe responsável realize as alterações no modelo para que logo em seguida possam ser realizadas as correções e os novos Testes de Penetração.

4.2.3 Testes de Vulnerabilidade

A Terceira Fase se torna essencial para avaliar se a aplicação contém vulnerabilidades, pois serão realizados Testes de Vulnerabilidades com o intuito de identificar quais falhas a aplicação possui. Essa fase é composta por três etapas: Teste de Penetração, Web *Scanners* e

[7] Medida tomada por uma página interativa, quando a página inteira e seus conteúdos são enviados para o servidor para processamento de algumas informações e, em seguida, o servidor mostra a mesma página de volta ao seu navegador.

Sumário. A partir dos Casos de Abusos, levantados na Primeira Fase, serão definidos os Testes de Penetração que serão utilizados para burlar o sistema. Após definição, os Testes de Penetração serão efetivamente aplicados através de Web *Scanners*. Com o resultado da análise dos Web *Scanners*, serão desenvolvidos Sumários com o resultado dos testes.

4.2.3.1 Teste de Penetração

Os Testes de Penetração são tentativas legais e autorizadas de localizar e explorar sistemas de computadores de forma bem-sucedida com o intuito de tornar esses sistemas mais seguros. Esse processo inclui analisar as vulnerabilidades, bem como oferecer ataques que funcionem como prova de conceito para demonstrar que eles são reais. Os Testes de Penetração adequados sempre terminam com recomendações específicas para endereçar e corrigir os problemas descobertos durante o teste.

A ideia geral consiste em identificar problemas de segurança usando as mesmas ferramentas e técnicas usadas por um invasor. É possível então atenuar os riscos identificados por essas descobertas antes que um *hacker* de verdade as explore (ENGEBRETSON, 2013).

A MADS-WEB faz uso da metodologia para técnicas de penetração que segundo Engebretson (2013), é dividida em quatro fases: Reconhecimento, *Scanning*, Exploração de falhas (*exploitation*) e Pós-Exploração (ou Preservação do Acesso). Compreender a sequência adequada em que esses passos são executados é fundamental para realizar um teste de penetração abrangente e realista. A figura 13 apresenta essas atividades.

Figura 13 - A metodologia Hacking para Testes de Penetração.

Fonte: Adaptado de Engebretson (2013).

O primeiro passo em qualquer Teste de Penetração é o reconhecimento considerando a atividade que cuida da coleta de informações sobre o alvo. Quanto mais informações coletar sobre o seu alvo, mais possibilidade haverá de ser bem-sucedido nos passos subsequentes.

O segundo passo nessa metodologia pode ser dividido em duas atividades distintas. A primeira atividade a ser realizada é o *scanning* de portas. Após concluir o *scanning* de portas, tem-se uma lista de portas abertas e de serviços em potencial sendo executados em cada um dos alvos. A segunda atividade da fase de *scanning* é o *scanning* de vulnerabilidades que corresponde ao processo de localizar e de identificar pontos fracos específicos nos softwares e nos serviços presentes no alvo.

Com os resultados obtidos no passo 2, o próximo passo será a fase de exploração de falhas (*exploitation*). Depois da descoberta exata de que as portas estão abertas, informar quais serviços estão executando nessas portas e quais vulnerabilidades estão associadas a esses serviços. Assim, a equipe responsável pelos Testes de Penetração pode começar a atacar o alvo. A exploração pode envolver diversas técnicas, Web *scanners* e códigos diferentes. O objetivo final da exploração consiste em identificar brechas de segurança.

A fase final a ser analisada é a de pós-exploração e preservação do acesso. Com frequência, os *payloads*[8] enviados na fase de exploração de falhas permitem apenas um acesso temporário ao sistema. Como a maior parte dos *payloads* não é persistente deve-se prosseguir

[8] É a parte dos dados transmitidos, que é o objetivo fundamental da transmissão, excluindo as informações enviadas com ela (como cabeçalhos ou *metadados*, também conhecido como dados complementares, que podem conter, dentre outras informações, a identificação da fonte e do destino dos dados) apenas para facilitar a entrega.

rapidamente para a fase de pós-exploração para criar uma porta dos fundos (*backdoor*[9]) mais permanente para o sistema. Esse processo permite que o acesso de administrador sobreviva quando os programas forem encerrados e até mesmo quando houver uma reinicialização do sistema.

Na MADS-WEB a etapa de pós-exploração se torna desnecessária devido não haver necessidade de manter acesso para futuras invasões, como sugere a própria metodologia de penetração, o propósito é apenas descobrir a quais falhas a aplicação está suscetível.

Neste sentido, se faz necessário o uso de ferramentas específicas para realização dos Testes de Penetração que são denominadas Web *scanners*.

4.2.3.2 Web *Scanners*

São conjuntos de ferramentas capazes de detectar se uma aplicação contém vulnerabilidades. Estas ferramentas têm como objetivo facilitar e automatizar a busca por vulnerabilidades em uma aplicação, bem como mapear a estrutura do site, ou seja, possibilitam ter uma visão completa do estado de segurança da aplicação e ajudam a automatizar os testes, pois fazem uma varredura no site de destino que se pretende realizar análise quanto à segurança.

As ferramentas propostas na MADS-WEB para realização de Análise de Vulnerabilidade são todas de código fonte aberto e foram utilizadas como forma de realização de Testes de Penetração em uma aplicação Web, tais ferramentas fazem parte do *Kali Linux*[10]. A primeira coluna da tabela 4 apresenta o nome das ferramentas que foram utilizadas. A segunda coluna apresenta a sua respectiva descrição e o propósito de cada uma na realização de um Teste de Penetração.

[9] *Backdoor* é um recurso utilizado por diversos *malwares* para garantir acesso remoto ao sistema ou à rede infectada, explorando falhas críticas não documentadas existentes em programas instalados, *softwares* desatualizados e do *firewall* para abrir portas do roteador.
[10] *Kali Linux* é uma distribuição Linux baseada em Debian para realização de testes de penetração.

Tabela 4 - Apresenta as Ferramentas Utilizadas para Realização de Testes de Penetração.

Ferramenta	**Descrição**
Metasploit Framework (MSF)	Tem como objetivo realizar análise de vulnerabilidades de segurança e facilitar testes de penetração. Utilizada nos testes de Quebra de Autenticação.
Nikto	Trata-se de um scanner que realiza testes abrangentes contra servidores Web, incluindo mais de 6.700 arquivos potencialmente perigosos. Também verifica quais são os itens de configuração do servidor, tais como a presença de arquivos de índice, opções do servidor HTTP, tentando identificar os servidores Web instalados e quais *softwares* fazem parte. Itens de digitalização e *plugins* são atualizados com frequência e podem ser atualizados automaticamente. Utilizada nos testes de Referência Insegura e Direta a Objetos.
Web Application Attack and Audit Framework (W3af)	É um *scanner* de segurança de aplicações Web de código aberto. Fornece informações sobre vulnerabilidades de segurança e ajuda no esforço de teste de penetração. Utilizada nos testes de Injeção de Código *Sql.*
Spidering	São principalmente utilizados para criar uma cópia de todas as páginas visitadas para um pós-processamento por um motor de busca que irá indexar as páginas baixadas para prover buscas mais rápidas. *Crawlers* também podem ser usados para tarefas de manutenção automatizadas em um *website*, como checar os links ou validar o código HTML. Utilizada nos testes de Utilização de Componentes Vulneráveis Conhecidos.
WebScarab	Ferramenta de teste de aplicações de segurança Web. Ele serve como um *proxy* que intercepta e permite que as pessoas alterem pedidos feitos a navegadores Web (HTTP e HTTPS) e respostas do servidores Web. Utilizada nos testes de *Cross-Site Request forgery (CSRF).*
OWASP-ZAP	É uma ferramenta utilizada em testes de penetração desenvolvida para encontrar vulnerabilidades em aplicações Web. Fornece *scanners* automatizados, bem como um conjunto de ferramentas que permitem encontrar vulnerabilidades de segurança manualmente. Utilizada nos testes Redirecionamentos e Encaminhamentos Inválidos, Configurações Incorretas de Segurança, Exposição de Dados Sensíveis, Falta de Função para Controle do Nível de Acesso e *Cross-Site Scripting* (XSS).

4.2.3.3 Sumários

Refere-se ao documento formal onde devem constar todos os testes feitos e as variáveis que foram levadas em consideração, assim como os resultados e as soluções propostas para que a equipe responsável por fazer as correções do código possa utilizar como referência, como forma de agilizar as correções a serem feitas.

O propósito do sumário é oferecer uma visão geral simples, não técnica, de uma a duas páginas, relativas às descobertas. Esse relatório deve ressaltar e sintetizar os problemas mais graves descobertos pela equipe de teste. É de suma importância que ele possa ser lido e compreendido tanto pelo pessoal técnico quanto pelo não técnico. No relatório deve conter as medidas para mitigar os problemas encontrados e as soluções devem ser apresentadas para a equipe de desenvolvimento. Exemplos de sumários são demonstrados nos apêndices desse trabalho, onde constam resultados dos testes de penetração realizados em uma aplicação Web.

4.3 APLICANDO A MADS-WEB NO MOODLE

O Moodle é uma aplicação Web definida como uma plataforma de educação a distância usada em muitas universidades como forma de permitir a interação do aluno on-line. Instituições fazem uso do Moodle por causa da sua flexibilidade, adaptabilidade e facilidade de uso. Essa aplicação Web tem uma substancial base de usuários com 56.185 sites ativos e 46.343.749 usuários finais (MOODLE, 2014). Assim como qualquer outra aplicação Web disponibilizada, é suscetível a falhas de programação e de desenvolvimento. Por se tratar de uma aplicação de código fonte aberto possui características que permitem que seja modificado e melhorado por todos os usuários que possuam conhecimento técnico para desenvolver esta atividade. Isso faz com que a aplicação se torne vulnerável em face das suas características.

Dessa forma as vulnerabilidades podem ser encontradas no início, mas elas também podem ser exploradas antes que os *patches* estejam disponíveis. Novas vulnerabilidades como *backdoors*[11], podem ser adicionadas por contribuintes maliciosos ou por apresentar falhas de segurança ocasionadas por problemas de codificação, o que pode resultar em vulnerabilidade.

Outros riscos conhecidos, específicos no Moodle, podem permanecer inseridos em forma de autenticação, disponibilidade, confidencialidade, integridade. Destaca-se, também, que o gerenciamento de sessão no Moodle não é inerentemente seguro, pois as comunicações

[11] É um recurso utilizado para garantir acesso remoto ao sistema ou à rede infectada, explorando falhas críticas não documentadas existentes em programas instalados, softwares desatualizados e do firewall para abrir portas do roteador.

nem sempre são feitas através de SSL[12] ao longo de toda conexão, possibilitando, assim, um possível sequestro de sessão. O Moodle nem sempre exige que os usuários tenham uma nova autenticação, devido ao cache de sessão, e não restringe o acesso por meio de URLs. Ele possui um frágil armazenamento criptográfico (por não fazer uso de *hashes* MD5[13], por exemplo) (KUMAR, 2011).

Todos esses riscos de segurança elencados apresentam problemas não só para os usuários finais, mas também para a instituição, seja pública ou privada, visto que da mesma forma que um *software* proprietário, ela pode apresentar falhas semelhantes.

Neste sentido, se faz necessário que a equipe de desenvolvimento faça uso da metodologia para Testes de Penetração, validando se o sistema é seguro e se ele atende aos requisitos básicos de confiabilidade.

Diante do exposto, foi aplicada a Terceira Fase da metodologia MADS-WEB, Testes de Vulnerabilidades, direcionada ao Moodle, de forma a validar essa fase da metodologia onde foram realizadas as etapas de Testes de Penetração com a execução das ferramentas elencadas na etapa de Web *Scanners* em face aos Casos de Abuso reportados na Primeira Fase, conforme vulnerabilidades definidas no OWASP *TOP* 10.

A segunda fase da MADS-WEB não foi possível aplicar ao Moodle devido ao mesmo já se encontrar desenvolvido, não sendo possível no escopo da proposta desse trabalho realizar uma engenharia reversa no código do mesmo, o intuito é exatamente de evidenciar as falhas devido a não utilização de aspectos de segurança durante seu ciclo de desenvolvimento, como propõe a proposta na MADS-WEB. Neste trabalho serão apresentados os resultados dos testes na próxima subseção.

4.3.1 Nível das Criticidades e Vulnerabilidades

Para identificar o impacto das vulnerabilidades encontradas, as vulnerabilidades foram divididas em níveis de criticidade. Estes níveis de criticidade estão relacionados com o

[12] Protocolo de segurança que protege as telecomunicações via internet.
[13] Algoritmo de *hash* de 128 bits unidirecional, muito utilizado por softwares na verificação de integridade de arquivos e *logins*.

tamanho do impacto e a facilidade de exploração dos problemas presentes no *software* do estudo de caso.

Esta é uma medida subjetiva, tendo em vista que o impacto e a facilidade de exploração das vulnerabilidades dependem do tamanho da organização, seu nicho de negócio, da habilidade do atacante, entre outras características. A criticidade dos problemas encontrados pode ser dividida em quatro níveis, a seguir:

Informação: Não causa grande impacto ao negócio da organização, mas requer certa atenção, pois pode ocasionar problemas futuros a partir desses alertas encontrados;

Alta: causa grande impacto ao negócio da organização, como vazamento de informações, podendo ser explorado por *hackers* experientes;

Média: causa baixo impacto ao negócio da organização, podendo ocasionar vazamento relevante de informações, comprometimento mediano da imagem da organização;

Baixa: não causa impacto ao negócio da organização com baixo ou nenhum prejuízo, vazamento irrelevante de informação, baixo comprometimento da imagem da organização.

4.3.2 Configuração do Ambiente e Resultados dos Testes de Penetração

Foram feitos Testes de Penetração, utilizando ferramentas apropriadas citadas na MADS-WEB, especificamente na fase 3. A tabela 5 apresenta todos os *softwares* utilizados nos testes e os equipamentos que foram necessários.

Tabela 5 - Cenários dos Testes Realizados.

Máquina	Configuração	Finalidade
Servidor Físico DELL T410	• *Linux Centos* 7.0 • Moodle 2.7.1 • Apache/2.4.6 • PHP/5.4.16 • Maria DB 5.5.37	Esta máquina atua como Servidor do Moodle em ambiente *Linux*.
Máquina Virtual 01	• *Kali Linux*	*Kali Linux* é uma distribuição Linux baseada em Debian para realização de Testes de Penetração.
Máquina Virtual 02	• *Ubuntu* 14.04.1 LTS	Utilizado como máquina cliente para acesso ao Moodle através da rede.
Notebook Dell Vostro	• *Windows* 8.1 64 bits • *VirtualBox* 4.3.14	Utilizado como máquina onde foi instalado *Virtualbox*.

Uma série de testes foi realizada utilizando a configuração, conforme ilustrado na tabela 6. A primeira coluna refere-se ao nome das vulnerabilidades que foram analisadas, na segunda coluna são informados os *status* quanto à descoberta, ou não, de alguma vulnerabilidade. Na terceira coluna é possível visualizar, de forma quantitativa, as falhas que foram descobertas em face da aplicação das ferramentas elencadas na quarta coluna, como sugere a Terceira Fase da MADS-WEB.

Não foram encontradas vulnerabilidades quanto às falhas de segurança que incluem Injeção de Código Sql, Referência Insegura e Direta a Objetos, *Cross-Site Request forgery* (CSRF), Utilização de Componentes Vulneráveis Conhecidos, Redirecionamentos e Encaminhamentos Inválidos.

Tabela 6 - Resumo dos Resultados.

Vulnerabilidade	**Status**	**Qtd**	**Ferramenta**
Injeção de Código Sql	Não Encontrada	0	W3af
Referência Insegura e Direta a Objetos	Não Encontrada	0	Nikto
Cross-Site Request forgery (CSRF)	Não Encontrada	0	WebScarab
Utilização de Componentes Vulneráveis Conhecidos	Não Encontrada	0	Spidering
Redirecionamentos e Encaminhamentos Inválidos	Não Encontrada	0	OWASP Zap
Quebra de Autenticação	Encontrada	247	Metasploit
Configurações Incorretas de Segurança	Encontrada	262	OWASP Zap
Exposição de Dados Sensíveis	Encontrada	4	OWASP Zap
Falta de Função para Controle do Nível de Acesso	Encontrada	11	OWASP Zap
Cross-Site Scripting (XSS)	Encontrada	2	OWASP Zap

No que diz respeito à quebra de autenticação foi encontrado um total de 247 (duzentas e quarenta e sete) no Moodle. Esse resultado foi gerado devido a não utilização da *flag HttpOnly*, seu uso mitiga as ações de atacantes quando estes tentam realizar sequestro de sessão através de CSRF. *Cookies* com a *flag HttpOnly* não podem ser acessados diretamente por scripts no lado do cliente, por exemplo, *JavaScript*[14]. Isso significa que mesmo havendo uma vulnerabilidade permitindo que o CSRF e o usuário sejam incentivados a clicar em um *link* que explore essa falha, o navegador não enviará ou disponibilizará o *cookie* marcado com a *flag HttpOnly*. Porém, a *flag* é somente uma das técnicas para mitigar o risco de *CSRF*: se usada sozinha, não poderá eliminar a ameaça por completo. Por isso, é bastante aconselhável o uso dessa *flag* para mitigação de riscos.

Foram encontradas 262 (duzentas e sessenta e duas) configurações incorretas de segurança, isso é ocasionado pela falta de configuração no cabeçalho *X-Content-Type-Options*, onde a opção *Type-Options* não foi definida como '*nosniff*'. Isso permite que versões

[14] *JavaScript* é uma linguagem de programação interpretada.

mais antigas do Internet Explorer e Chrome possam executar *MIME-Sniffing* no corpo da resposta, podendo causar repostas no corpo a ser interpretadas e exibidas como um tipo de conteúdo que não seja o tipo de conteúdo declarado. Portanto, se faz necessário para mitigar problemas relacionados com configurações incorretas de segurança, setar a opção *Type-Options* como *nosniff*.

Quanto aos problemas com exposição de dados sensíveis foram encontrados um total de 4 (quatro) formulários de preenchimento dentro do Moodle. A função AUTOCOMPLETE não está desabilitada, sendo assim, senhas podem ser armazenadas e depois recuperadas. Desse modo, é altamente aconselhável que seja deixada a função AUTOCOMPLETE nos formulários, contendo senhas AUTOCOMPLETE*='OFF'*.

No que diz respeito à falta de função para controle do nível de acesso, foram encontradas um total de 11 (onde) vulnerabilidades, isso se deve ao fato de *X-Frame-Options header* não está incluído na resposta HTTP para proteger contra ataques *clickjacking* ("furto de click"). Ataques de *clickjacking* é uma técnica fraudulenta. O roubo de *click* é uma armadilha preparada para que o usuário pense que está fazendo uma ação num determinado site, mas na verdade os cliques executados nessa ação estão sendo usados pelo atacante para executar operações maliciosas. A ideia básica se concentra em sobrepor algum elemento clicável na tela com um *iframe*[15], ou quadro invisível contendo a página em que o atacante quer que o usuário execute uma atividade. No *clickjacking* típico esse *iframe* é 100% transparente, ou redimensionado e posicionado de forma que ocupe a área de um botão na página que a vítima acredita ser inofensiva. O usuário clica no botão da página inofensiva, mas na verdade seu clique é executado em outra página, naquela que foi carregada dentro do *iframe*.

Como forma de contornar esse problema, a solução mais prática seria incluir no cabeçalho HTTP a opção *X-Frame-Options,* marcado com *"SAMEORIGIN."*. Dessa forma, permitiria apenas que a página pudesse ser mostrada em um quadro que venha da mesma origem que a própria página.

[15] É uma 'página dentro de outra página'. Na verdade, é um "quadro", onde é inserida outra página. É útil para não precisar ficar repetindo um mesmo conteúdo várias vezes.

Foram detectadas 2 (duas) vulnerabilidades no que diz respeito a XSS. Através de um XSS, o *cracker*[16] injeta códigos *JavaScript* em um campo texto de uma página já existente e este *JavaScript* é apresentado para outros usuários, pois persiste na página. Exemplo de ataque: Imaginem que o *cracker* insira, em um fórum de um *website* alvo de ataque, um texto que contenha um trecho de *JavaScript*. Este *JavaScript* poderia, por exemplo, simular a página de *login* do site, capturar os valores digitados e enviá-los a um site que os armazenasse. Quando o texto do fórum for apresentado a outros usuários, um site atacado pelo XSS exibirá o trecho de *JavaScript* digitado anteriormente nos *browsers* de todos os outros usuários, provocando, assim, a brecha de ataque.

Como forma de evitar problemas com XSS, algumas medidas devem ser tomadas. Todos os dados de usuário a serem utilizados para construção do contexto HTML (corpo, atributo, *JavaScript*, CSS ou URL) devem ser verificados para assegurar que não contenham nenhum conteúdo ativo (*JavaScript, ActiveX, Flash, Silverlight*) e que sejam codificados de maneira apropriada, por exemplo, transformando metacaracteres[17] em códigos de escape HTML.

Uma das principais consequências da vulnerabilidade de XSS é a exposição negativa do sistema e a possibilidade de utilização da falha para a distribuição de *phishing*[18] e facilitação de fraudes.

4.4 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Este capítulo apresentou a metodologia definida nesta dissertação, foi enfatizada a importância de se prover segurança nas atividades que fazem parte do ciclo de

[16] *Cracker* é o termo usado para designar o indivíduo que pratica a quebra (ou *cracking*) de um sistema de segurança de forma ilegal ou sem ética.
[17] Um metacaractere é um caractere ou sequência de caracteres com significado especial em expressões regulares.
[18] Em computação, *phishing*, termo oriundo do inglês (*fishing*) que quer dizer pesca, é uma forma de fraude eletrônica, caracterizada por tentativas de adquirir dados pessoais de diversos tipos; senhas, dados financeiros como número de cartões de crédito e outros dados pessoais.

desenvolvimento de um *software*, isso se faz necessário devido à necessidade de prover segurança em sistemas desde sua concepção até a entrega ou liberação do produto ao usuário final, seja uma pessoa, empresa pública ou privada.

Em um passado recente, inúmeras organizações importantes foram comprometidas por meio de suas aplicações. Embora seus departamentos de relações públicas possam argumentar que foram vítimas de *hackers* altamente sofisticados, na realidade, a maioria desses ataques explorou vulnerabilidades simples, muito bem compreendidas há anos. Empresas menores que não se sentem no centro das atenções podem estar mais expostas ainda. E muitas daquelas que foram comprometidas desconhecem o fato.

Através de técnicas de exploração de falhas apresentadas neste trabalho, foi possível evidenciar vulnerabilidades, às quais o Moodle está suscetível. Foram realizados Testes de Penetração, com a utilização de ferramentas especificas elencadas na etapa Web *Scanners*, que proporcionaram a realização dos testes de forma dinâmica. Ainda neste sentido, também foram obtidos resultados e evidenciadas vulnerabilidades existentes no Moodle com aplicação da MADS-WEB. Desse modo, foram apresentadas formas de como contornar as vulnerabilidades descobertas.

Dando continuidade a esse estudo, o próximo capítulo traz uma discussão sobre o que foi abordado neste trabalho, as suas conclusões e as perspectivas para trabalhos futuros.

5. CONCLUSÕES E TRABALHOS FUTUROS

Nos últimos anos, houve mais ênfase no desenvolvimento de *softwares* seguros e, como consequência, as aplicações Web atuais são muito mais seguras do que as versões anteriores. Houve uma forte pressão no sentido de incluir a segurança nos estágios inicias do ciclo de vida do desenvolvimento de *software* e de formalizar a especificação dos Requisitos de Segurança de forma padronizada. Também houve um aumento enorme na organização de diversas comunidades dedicadas à segurança de aplicações Web. Empresas de grande porte como a Microsoft, IBM *e* Google, são extremamente preocupadas com segurança, visto que suas aplicações são utilizadas por pessoas em todo mundo, podendo colocar em risco sua reputação, caso falhas sejam descobertas e exploradas.

Entre as inúmeras vulnerabilidades, estudos recentes realizados pela OWASP (2014) indicam as dez mais recorrentes e mais exploradas pelos invasores. Muitas destas vulnerabilidades identificadas pela simples realização de atividades de testes de penetração, como é sugerido na fase três da metodologia apresentada neste artigo MADS-WEB.

A implantação de um software seguro exige a observância das características de segurança já nas fases iniciais, em conjunto com o entendimento do processo de negócio, ao invés de adicionar segurança apenas no final do ciclo de vida.

Neste sentido, a metodologia apresentada neste trabalho favorece uma integração forte entre a equipe que faz levantamento dos Requisitos de Seguranças, Casos de Abuso e Análise de Risco, na fase um, como a equipe de Modelagem, Plano de Testes e Codificação, na fase dois e a equipe de Testes de Penetração que engloba Web *scanners* e os Sumários, na fase três, visto que toda equipe está envolvida com aspectos quanto à segurança do sistema em todo seu ciclo de vida. De modo que, beneficia as equipes de desenvolvimentos, inserindo elementos de segurança no ciclo de vida do *software*.

A partir dos testes aplicados, é perceptível que a maioria das vulnerabilidades está relacionada às tentativas e técnicas que procuram enganar os usuários, como é o caso da

vulnerabilidade por XSS, mas elas que devem ser mitigadas pela equipe de desenvolvimento, visto que o usuário é a parte mais fraca do sistema.

Neste sentido, é importante ressaltar que a segurança do Moodle deve ser estudada e levada a sério, pois problemas como os reportados neste estudo podem, não só comprometer as instituições que implementaram o Moodle, mas também ao público que utiliza esta plataforma de educação a distância.

De forma geral, este trabalho contribui para que equipes de desenvolvimento possam fazer uso da metodologia apresentada neste trabalho para a construção de aplicações mais seguras em função da utilização de aspectos de segurança no ciclo de vida de desenvolvimento de um *software*. O que torna fundamental o abandono da cultura de que segurança é algo a ser acoplado ao final do projeto de desenvolvimento de software. No que diz respeito aos trabalhos futuros, sugere-se:

1. A utilização da metodologia em outros projetos de *software* para desenvolvimento de sistemas e estabelecimento de métricas de tempo, custo e retorno do investimento;

2. O estabelecimento de um modelo de maturidade de segurança de *software* voltado à construção do *software*;

3. O estabelecimento de um modelo de levantamento de requisitos de segurança de software;

4. Desenvolvimento de um *framework* especializado que realize os testes de vulnerabilidade de forma dinâmica em uma única aplicação.

5.1 PUBLICAÇÕES

Esta dissertação proporcionou algumas publicações, tais como:

Em eventos (Aprovados):

1. SILVA, R. R. T., LIMA, R. W., LEITE, C. R. M.

RSAET - Uma Metodologia para Desenvolvimento de Software Seguro In: STIN - Seminário de Tecnologia da Informação do Noroeste do Estado do RS, 2014, Santo Ângelo.

2. SILVA, R. R. T., LIMA, R. W., LEITE, C. R. M.

Vulnerabilidades de Segurança no Moodle In: Escola Potiguar de Computação e suas Aplicações (EPOCA), 2014, Santa Cruz-RN.

Em periódicos (Aprovados):

1. SILVA, R. R. T., LIMA, R. W., LEITE, C. R. M.

Investigação de Segurança no Moodle.

RETEC - Revista de Exatas e Tecnológicas. ISSN: 2236-739X

Em periódicos (Submetidos):

1. SILVA, R. R. T., LIMA, R. W., LEITE, C. R. M.

MADS-WEB: Metodologia Aplicada ao Desenvolvimento Seguro de Aplicações Web.

iSys - Revista Brasileira de Sistemas de Informação, Rio de Janeiro. ISSN: 1984-2902

2. SILVA, R. R. T., LIMA, R. W., LEITE, C. R. M.

Análise de Vulnerabilidades em Ambientes Virtuais de Aprendizagem: Um Estudo de Caso no Moodle.

Revista de Sistemas de Informação da FSMA, Rio de Janeiro. ISSN: 1983-5604

REFERÊNCIAS

ABNT ISO IEC 17799. **Tecnologia da informação – Técnicas de segurança – Código de prática para a gestão da segurança da informação**, 2ª ed, Rio de Janeiro, 2005.

ABNT NBR ISO/IEC 27001:2013, **Tecnologia da informação — Técnicas de segurança — Sistemas de gestão da segurança da informação — Requisitos**. Novembro 2013.

ABNT NBR ISO/IEC 27002:2013, **Tecnologia da informação — Técnicas de segurança — Código de prática para controles de segurança da informação**. Novembro 2013.

ACUNETIX - **Web Vulnerability**, 2014. Disponível em http://www.acunetix.com. Acessado em 24/08/2014.

AL-AJLAN, A. S., **A comparative study between e-learning features, methodologies, tools and new developments for e-learning**, dr. elvis pontes (ed.). In: Methodologies, Tools and New Developments for E-Learning. INTECH, 2012. p. 25. Qassim University Kingdom of Saudi Arabia. Disponível em: <http://www.intechopen.com/books/methodologies-tools-and-new-developments-for-e-learning/acomparative- study-between-e-learning-features>.

ALAVA, S. **Ciberespaço e formações abertas: rumo a novas práticas educacionais?** Porto Alegre: Artes Médicas, 2002.

ALJBORI, MOHANNED ABDULLAH; GUIRGUIS, SHAWKAT K; MADBOULY, MAGDA M. **Adaptable mobile user interface for securing e-learning environment. Transactions on Networks and Communications**, [S.l.], v. 2, n. 4, p. 64-83, sep. 2014. ISSN 2169-6129. Available at: <http://scholarpublishing.org/index.php/TNC/article/view/360>. Date accessed: 13 Sep. 2014. doi:http://dx.doi.org/10.14738/tnc.24.360.

ANTOANELA NAAJI AND COSMIN HERMAN. 2011. **Implementation of an e-learning system. optimization and security-related aspects.** In Proceedings of the 15th WSEAS international conference on Computers, Nikos Mastorakis, Valeri Mladenov, Zoran Bojkovic, Fragkiskos Topalis, and Kleanthis Psarris (Eds.). World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA, 412-417.

ALENCAR , GLINER DIAS. QUEIROZ , ANDERSON APOLONIO LIRA.; QUEIROZ, RUY JOSÉ GUERRA BARRETTO. **Insiders: análise e possibilidades de mitigação de ameaças internas.** Revista Eletrônica de Sistemas de Informação, v. 12, n. 3, set-dez 2013, artigo doi:10.5329/RESI.2013.1203006

AOKI , ERIC KOMIYAMA. CARVALHO , ALAN HENRIQUE PARDO. **Práticas de segurança para o desenvolvimento de sistemas Web**.; Fasci-Tech – Periódico Eletrônico da FATEC-São Caetano do Sul, São Caetano do Sul, v. 1, n. 5, Out/Dez 2011, p.56 a 66.

ATAÍDE, M. A. **Novos tempos: educação a distância e a revolução da tecnologia na sociedade em rede.** Revista Vozes dos Vales da UFVJM, Minas Gerais, ano 2, n. 3, 05/2013.

BACUDIO, A. G. et. al. An Overview of Penetration Testing, International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.6, Novembro 2011.

BARBERO, J. M. **Dos meios às mediações: comunicação, cultura e hegemonia**. Rio de Janeiro: UFRJ, 2003.

BARBOSA, R. M. **Ambientes Virtuais de Aprendizagem**. Porto Alegre: Editora Artmed, ISBM:85-363-0515-0, 2005. 47-48 p.

BARRINGTON, REBECCA. **Moodle Gradebook - Set up and customize the gradebook to track student progress through Moodle**. Packt Publishing, BIRMINGHAM – MUMBAI, 2012.

BOOCH, G. AND RUMBAUGH, J. AND JACOBSON, I., **UML: Guia do usuário**. Campus - RJ, Elsevier Brasil, 2006.

BROAD, JAMES. BINDNER, ANDREW. **Hacking com Kali Linux: Técnicas práticas para testes de invasão**., São Paulo: Novatec , 2014.

BSIMM - **The Building Security In Maturity Model**, 2014. Disponível em http://www.bsimm.com/online/intelligence/am/. Acessado em 05/07/2014.

CARBONELL, J. A **Aventura de Inovar**. Porto Alegre: Artmed, 2002.

CARVALHO, A. A. A. **Multimédia: um conceito em evolução**. revista portuguesa de educação, n.15, a. 1, p. 245-268. In: . [S.l.: s.n.], 2002.

CERT.BR - **Centro de Estudos, Respostas e Tratamentos de Incidentes de Segurança no Brasil**, 2014. Disponível em: <http://www.cert.br/stats/incidentes/2013-jan-dec/tipos-ataque.html>. Acessado em 15/09/2014.

CHANDRA, P. et al. **OpenSAMM – Open Software Assurance Maturity Model**, OWASP, Cigital, Creative Commons, 2012.

COSTA, J. W. da; OLIVEIRA, M. A. M. **Novas linguagens e novas tecnologias: educação e sociabilidade.** Petrópolis: Vozes, 2004.

CREATIVE COMMONS, Março, 2009. **Electronics Computer Technology (ICECT),** 2011 3rd International Conference on (Volume:5), v. 5, p. 5, 2011. Disponível em: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=arnumber=5942032url=http

DEFTA (CIOBANU) COSTINELA – LUMINITA, "**Information security in E-learning Platforms**",3rd World Conference on Educational Sciences, Volume 15, pp. 2689–2693, Istanbul, Turkey 2011.

ENGEBRETSON, PATRICK. **Basics of hacking and penetration testing, the ethical hacking and penetration testing made easy**., SYNGRESS (ELSEVIER), 2013.

ENGHOLM JÚNIOR, HÉLIO,. **Engenharia de Software na Prática**. São Paulo: Novatec Editora, 2010.

ENRICONE, D. **Ser Professor**. Porto Alegre: Edipucrs, 2002.

FADI HAJ SAID. 2011. **Security-Based Risk Assessment for Software Architecture**. Ph.D. Dissertation. West Virginia Univ., Morgantown, WV, USA. Advisor(s) Hany H. Ammar. AAI3530557

FLOYD, COLTON. SCHULTZ, TYLER AND FULTON, STEVEN. **Security Vulnerabilities in the open source Moodle eLearning System**. Proceedings of the 16th Colloquium for Information Systems Security Education. Lake Buena Vista, Florida June 11 - 13, 2012.

GOERTZEL, K. M et al. **Software Security Assurance: A State of the Art Report (SOAR), Information Assurance Technology Analysis Center (IATAC),** Julho 2007.

HAROLD F. TIPTON. 2010. **Official (Isc)2 Guide to the SSCP Cbk**, Second Edition (2nd ed.). Auerbach Publications, Boston, MA, USA.

HOGLUND, GREG. **Como quebrar códigos: a arte de explorar (e proteger) software** / Greg Hoglund, Gary Macgraw; tradução Docware Traduções Técnicas; revisão técnica Luiz Gustavo C. Barbato. São Paulo: Pearson Makron Books, 2006.

HOHLFELDT, A.; MARTINO, L. C.; FRANçA, V. V. **Teoria da Comunicação: conceitos, escolas e tendências**. Petrópolis, Rio de Janeiro: Vozes, 2002.

ISO/IEC 12207, **Systems and software engineering - Software life cycle processes**, Geneva, 2008.

ISO/IEC 15408-1:2009 **Information technology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction and general model**. Geneva 2009.

ISO/IEC 15408-1:2009 **Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Introduction and general model**. Geneva 2009.

ISO/IEC 15408-1:**2009 Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Introduction and general model**. Geneva 2009.

ISO/IEC 21827, **Systems Security Engineering - Capability Maturity Model, 2ª ed**, Geneva, 2008.

JACOBS, STUART. **Engineering information security: The application of systems engineering concepts to achieve information assurance**. p. cm. ISBN 978-0-470-56512-4 (hardback). July 2011, Wiley-IEEE Press.

RANSOME, JAMES AND MISRA, ANMOL. 2013. **Core Software Security: Security at the Source. Auerbach Publications**, Boston, MA, USA.

KUMAR, S.; GANKOTIYA, AK.; DUTTA, K., "**A comparative study of moodle with other e-learning systems," Electronics Computer Technology (ICECT)**, 2011 3rd International Conference on , vol.5, no., pp.414,418, 8-10 April 2011 doi: 10.1109/ICECTECH.2011.5942032 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5942032&isnumber=5941942

KARAPANOS, NIKOLAOS. CAPKUN, SRDJAN.; **On the effective prevention of TLS man-in-the-middle attacks in web applications**. In Proceedings of the 23rd USENIX conference on Security Symposium (SEC'14). USENIX Association, Berkeley, CA, USA, 671-686.

KUMAR, S. KAMLESH, D.; **Investigation on Security in LMS Moodle,** Proceedinggs of International Journal of Information Technology and Knowledge Management, Kurukshetra University, Kurukshetra, India, 2011, 233-238.

LAKHAN, SHAHEEN E., AND KAVITA JHUNJHUNWALA., **Open Source Software in Education.** EDUCAUSE QUARTERLY 2 Nov. 2008: 33-40. Web.

LITTO, F. M. **Perspectivas da educação à distância no brasil: três cenários a ponderar** [1997-2002]. revista brasileira de aprendizagem aberta e à distância, vol. 2, n. 3, p. 89-97. In: . [S.l.: s.n.], 2003.

LONG, F. et al. **The CERT Oracle Secure Coding Standard for Java**, 1ª ed, Michigan: Pearson Education Inc, 2012.

LYRA, M. R. **Segurança e Auditoria em sistemas de Informação**. São Paulo: Editora Ciência Moderna, 2008.

PAUL, MANO., 2011. **Official (Isc)2 Guide to the CSSLP** (1st ed.). CRC Press, Inc., Boca Raton, FL, USA.

MANOTTI, A. **Curso Prático Auditoria de Sistemas**. São Paulo: Ciência Moderna, 2010.

MATOS, RODOLFO; CARVALHO, FILIPE. **Moodlewatcher: One year experience of detecting and preventing fraud when using Moodle quizzes**. 2011 Artigo em Conferência Internacional, EDULEARN12 Proceedings CD. http://hdl.handle.net/10216/66546

McCALL, J., RICHARDS, P., WALTERS, G.. **Factors in Software Quality.,3** Volumes, NTIS AD-A049-014, McGrawHill, 2006.

MCGRAW, G. Software Security, **IEEE Security and Privacy**, pp 80-83, Março/Abril 2004.

MCGRAW, G. Software security: **building security in, Boston: Addison Wesley Professional**, 2006.

MCGRAW, G., CHESS, B., MIGUES, S. **BSIMM - Buiding Security In Maturity Model**, Cigital, Creative Commons, 2012.

MCGRAW, G., CHESS, B., MIGUES, **S. BSIMM - Buiding Security In Maturity Model**. Disponível em http://www.bsimm2.com/. Acesso em: 21 Abril 2014.

MEAD, N. R., Chen, Peter, Dean, Marjon, Adams, Don. Osman, Hassan. Lopez, Lilian. Xie, Nick. **Security Quality Requirements Engineering (SQUARE) Methodology**, Carnegie Mellon University, Pittsburgh, 2004.

MICROSOFT, **Testes de Penetração**, 2008. Disponível em: http://msdn.microsoft.com/pt-br/magazine/cc507646.aspx. Acesso em: 21 Abril 2014.

MOODLE. **Modular Object-Oriented Dynamic Learning Environment**. 2014. [Acessado Agosto-2014]. Disponível em: <http://docs.moodle.org/dev/Moodlearchit ect ure>.

MSDN Magazine. **Encontre e corrija vulnerabilidades antes de lançar seu aplicativo** Disponível em http://msdn.microsoft.com/pt-br/magazine/cc163312.aspx Acessado em 12/07/2014.

NAKAMURA, E. T. **Segurança de Redes em Ambientes Cooperativos**. São Paulo: Novatec Editora, 2007.

NASCIMENTO, T. P. C. **Educação à distância em organizações públicas Mesa-redonda de pesquisa-ação**. [S.l.]: ENAP - Escola Nacional de Administração Pública, 2006.

OPENSAMM - **Software Assurance Maturity Model**, 2014. Disponível em http://www.opensamm.org/. Acessado em 05/07/2014.

OWASP, **OWASP Top Ten – 2010 The Ten Most Critical Web Application Security Risks**. Disponível em https://www.owasp.org/index.php/Top_10_2010-Main. Acessado em 05/07/2014.

PAULA FILHO, Wilson de Pádua, **Engenharia de Software**, RJ: LTC 2003.

PAULI, JOSH.; **Introdução ao Web Hacking - Ferramentas e Técnicas para Invasão de Aplicações Web**. São Paulo: Novatec Editora Ltda, 2014.

PAUL, MANO. 2011. **Official (Isc)2 Guide to the CSSLP (1st ed.)**. CRC Press, Inc., Boca Raton, FL, USA.

PELLANDA, N. M.; PELLANDA, E. C. **Ciberespaço: um hipertexto com Pierre Lévy**. Porto Alegre: Artes e Ofícios, 2000.

PONTES, ELVIS., SILVA, ANDERSON., GUELFI, ADILSON., KOFUGI, SÉRGIO. **Methodologies, Tools and New Developments for E-Learning**. ISBN 978-953-51-0029-4, 332 pages, Publisher: InTech, Chapters published February 03, 2012 under CC BY 3.0 license DOI: 10.5772/1115.

PRATYUSA K. MANADHATA AND JEANNETTE M. WING. 2011. **An Attack Surface Metric**. IEEE Trans. Softw. Eng. 37, 3 (May 2011), 371-386. DOI=10.1109/TSE.2010.60 http://dx.doi.org/10.1109/TSE.2010.60.

PRESSMAN, ROGER S., **Engenharia de Software** - (6ª edição), São Paulo, Ed. McGrawHill, 2006.

PTES - **Penetration Testing Execution Standard**. Disponível em http://www.pentest-standard.org. Acessado em 12/07/2014.

RIDGEWELL, W.W.; KUMAR, V.; KINSHUK, "**Immersive and Authentic Learning Environments to Mitigate Security Vulnerabilities in Networked Game Devices**," Signal-Image Technology & Internet-Based Systems (SITIS), 2013 International Conference on , vol., no., pp.1042,1048, 2-5 Dec. 2013 doi: 10.1109/SITIS.2013.168

SARAIVA, T. **Educação a Distância no Brasil: lições históricas**. Revista em Aberto, Brasília, ano 16, n.70, abr/jun 1996. Disponível em: <http://www.emaberto.inep.gov.br/index.php/emaberto/article/view/1048/950>. Acesso em: 04 de out. 2013.

SEARCH SECURITY - **TechTarget**, 2013. Disponível em http://searchsecurity.techtarget.com/feature/Software-Insecurity-BSIMM-V-does-a-number-on-secure-software-dev. Acessado em 05/07/2014.

SEI., **CMMI for Development Version 1.3**, Technical Report, Carnegie Mellon University, Pittsburgh, 2010.

SILVA, PEDRO; CARVALHO, HUGO; TORRES, CATARINA. **Segurança dos Sistemas de Informação - Gestão Estratégica da Segurança Empresarial**. Coleção: Sociedade da Informação. Lisboa, Portugal. Editora Centro Atlântico, 2003.

SOMMERVILLE, LAN. **Software engineering 9th ed**. Pearson, 2011

TSOUTSOS, NEKTARIOS GEORGIOS. MANIATAKOS, MICHAIL. **Trust No One: Thwarting "heartbleed" Attacks Using Privacy-Preserving Computation**. In Proceedings of the 2014 IEEE Computer Society Annual Symposium on VLSI (ISVLSI '14). IEEE Computer Society, Washington, DC, USA, 59-64. DOI=10.1109/ISVLSI.2014.86 http://dx.doi.org/10.1109/ISVLSI.2014.86

TRAVASSOS, G. H.; GUROV, D.; AMARAL, E. A. G. **Introdução a Engenharia de Software Experimental**, RT-ES-590/02, COPPE/UFRJ, Rio de Janeiro, 2002.

VIANA, SIDNEY. SILVA, RICHARD F. CENTRO, JUDITH PAVÓN. LAINE, JEAN M. **Segurança no Desenvolvimento de Aplicações Web com a Qualidade dos Dados**. Revista de Sistemas e Computação, Salvador, v. 3, n. 2, p. 93-104, jul./dez. 2013. http://www.revistas.unifacs.br/index.php/rsc

VALENTINI, C. B.; SOARES, E. M. S. **Aprendizagem em Ambientes Virtuais: compartilhando ideias e construindo cenários**. Caxias do Sul: EDUCS, 2005.

VIRGINIA N. L. FRANQUEIRA, THEIN THAN TUN, YIJUN YU, R. WIERINGA, AND B. NUSEIBEH. 2011. **Risk and argument: A risk-based argumentation method for practical security**. In Proceedings of the 2011 IEEE 19th International Requirements Engineering Conference (RE '11). IEEE Computer Society, Washington, DC, USA, 239-248. DOI=10.1109/RE.2011.6051659 http://dx.doi.org/10.1109/RE.2011.6051659

WOHLIN, C. et al. **Experimentation in Software Engineering: an introduction.** Kluwer
YALLI, J. S. Educação à distância. tecnologia educacional, v.22, p. 123-124. In:. [S.l.: s.n.], 2000.

WYSOPAL, CHRIS. NELSON, LUCAS NELSON. ZOVI, DINO DAI. DUSTIN, ELFRIEDE. **The Art of Software Security Testing: Identifying Software Security Flaws**. Published November 1st 2006 by Addison-Wesley Professional

APÊNDICES

APÊNDICE 1: Casos de Abuso

Aplicação	Moodle						Versão:		2.7
Nome	Injeção de código Sql								
Autor	Rodrigo Ronner						Data:		02/08/2014
Ferramenta	W3af								
Nível Alerta	Informação	X	Baixa		Média		Alta		Total = 0

Vulnerabilidade Encontrada, breve descrição:

Não foram encontradas falhas ou informações relevantes que possibilitassem um *hacker* realizar ataque por falhas de injeção de código Sql.

APÊNDICE 2: Casos de Abuso

Aplicação	Moodle					Versão:		2.7
Nome	Quebra de autenticação e Gerenciamento de Sessão							
Autor	Rodrigo Ronner					Data:		02/08/2014
Descrição	*Cookie set without HttpOnly flag*							
Ferramenta	*Metasploit*							
Nível Alerta	Informação	X	Baixa		Média		Alta	Total = 247

Vulnerabilidade Encontrada, breve descrição:

HttpOnly é uma *flag* opcional incluído em um cabeçalho de resposta HTTP *Set-Cookie*. Usando a *flag HttpOnly* ao gerar um cookie ajuda a atenuar o risco de script do lado do cliente à acessa o *cookie* protegido.

Requisição:

GET http://192.168.100.12/moodle HTTP/1.1 User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0;) Cache-control: no-cache Host: 192.168.100.12

Resposta:

HTTP/1.1 200 OK

Date: Sat, 02 Aug 2014 22:13:16 GMT

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_auth_kerb/5.4 mod_fcgid/2.3.9 mod_nss/2.4.6 NSS/3.15.4 Basic ECC PHP/5.4.16 mod_wsgi/3.4 Python/2.7.5

X-Powered-By: PHP/5.4.16

Set-Cookie: MoodleSession=sokib725pfgvfs45ka53fbc0t2; path=/moodle/

Expires: Mon, 20 Aug 1969 09:23:00 GMT

Cache-Control: no-store, no-cache, must-revalidate

Pragma: no-cache

Content-Language: pt

Content-Script-Type: text/javascript

Content-Style-Type: text/css

X-UA-Compatible: IE=edge

Cache-Control: post-check=0, pre-check=0, no-transform

Last-Modified: Sat, 02 Aug 2014 22:13:17 GMT

Accept-Ranges: none

X-Frame-Options: sameorigin

Content-Type: text/html; charset=utf-8

Solução indicada:

Habilitar *flag HttpOnly* para todos os cookies.

URLs Testadas:

http://192.168.100.12/moodle
http://192.168.100.12/moodle/
http://192.168.100.12/moodle/?lang=en
http://192.168.100.12/moodle/?lang=pt
http://192.168.100.12/moodle/?time=1401580800
http://192.168.100.12/moodle/?time=1404172800
http://192.168.100.12/moodle/?time=1406851200
http://192.168.100.12/moodle/?time=1409529600
http://192.168.100.12/moodle/?time=1412121600
http://192.168.100.12/moodle/?time=1414800000
http://192.168.100.12/moodle/?time=1417392000
http://192.168.100.12/moodle/?time=1420070400
http://192.168.100.12/moodle/?time=1422748800
http://192.168.100.12/moodle/calendar/export.php
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689479

http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689484
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689488
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689503
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689505
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689507
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689511
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689512
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689479
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689484
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689488
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689503
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689505
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689507
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689511
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689512
http://192.168.100.12/moodle/calendar/export_execute.php?authtoken=c75ba0e368def19ab4220fd65c2adae80bc92148&cal_d&cal_m&cal_y&generateurl=Get+calendar+URL&preset_time=weeknow&preset_what=all&userid=0
http://192.168.100.12/moodle/calendar/export_execute.php?authtoken=c75ba0e368def19ab4220fd65c2adae80bc92148&cal_d&cal_m&cal_y&generateurl=Obter+URL+do+calend%C3%A1rio&preset_time=weeknow&preset_what=all&userid=0
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3RpbWU9MTQxNDY4OTQ3OSZjb3Vyc2U9MQ%3D%3D&sesskey=Me4bKln9YX&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3RpbWU9MTQxNDY4OTQ3OSZjb3Vyc2U9MQ%3D%3D&sesskey=Me4bKln9YX&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA2ODUxMjAwJmNvdXJzZT0x&sesskey=71jGMGtegh&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA2ODUxMjAwJmNvdXJzZT0x&sesskey=71jGMGtegh&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=HBxWcxukzD&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=HBxWcxukzD&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=vonb7fSSmc&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=vonb7fSSmc&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=zWeKF3VCtH&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3

ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=zWeKF3VCtH&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=cudCvJMP8T&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=cudCvJMP8T&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=hkGQTmsvmN&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=hkGQTmsvmN&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=zPVMiamWO4&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=zPVMiamWO4&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=DQT3Sni0xz&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=DQT3Sni0xz&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=xUxSwXx9w9&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=xUxSwXx9w9&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=ySwf17uhk5&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=ySwf17uhk5&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=XFgUI37sbh&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=XFgUI37sbh&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=dg8bLbOlGr&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=dg8bLbOlGr&va

r=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=rfiLuoStl2&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=rfiLuoStl2&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDc5JmNvdXJzZT0x&sesskey=wQ3EAqTS8Y&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDc5JmNvdXJzZT0x&sesskey=wQ3EAqTS8Y&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=JkHRjwCjNt&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=JkHRjwCjNt&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=P7fzKsGDkt&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=P7fzKsGDkt&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=uBlLC1OkZL&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=uBlLC1OkZL&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE3MzkyMDAwJmNvdXJzZT0x&sesskey=NdI0p2exGj&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE3MzkyMDAwJmNvdXJzZT0x&sesskey=NdI0p2exGj&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDEyMTIxNjAwJmNvdXJzZT0x&sesskey=LCadRrFo8s&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDEyMTIxNjAwJmNvdXJzZT0x&sesskey=LCadRrFo8s&var=showglobal
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1404172800&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1406851200&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1409529600&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1412121600&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689459&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689460&view=month

http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689461&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689479&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689479&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689484&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689484&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689488&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689488&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689503&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689503&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689505&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689505&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689507&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689507&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689511&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689511&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689512&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689512&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414800000&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1417392000&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1420070400&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1406851200&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1409529600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1412121600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689459&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689460&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689461&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689479
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689479&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414800000&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1417392000&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1406851200&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1409529600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1412121600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689459&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689460&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689461&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689479
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689479&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414800000&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1417392000&view=month
http://192.168.100.12/moodle/course/index.php

http://192.168.100.12/moodle/course/index.php?lang=en
http://192.168.100.12/moodle/course/index.php?lang=pt
http://192.168.100.12/moodle/course/search.php
http://192.168.100.12/moodle/course/search.php?lang=en
http://192.168.100.12/moodle/course/search.php?lang=en&search=ZAP
http://192.168.100.12/moodle/course/search.php?lang=pt
http://192.168.100.12/moodle/course/search.php?lang=pt&search=ZAP
http://192.168.100.12/moodle/course/search.php?search=ZAP
http://192.168.100.12/moodle/course/view.php?id=1
http://192.168.100.12/moodle/login/forgot_password.php
http://192.168.100.12/moodle/login/forgot_password.php?lang=en
http://192.168.100.12/moodle/login/forgot_password.php?lang=pt
http://192.168.100.12/moodle/login/index.php
http://192.168.100.12/moodle/login/index.php?lang=en
http://192.168.100.12/moodle/login/index.php?lang=pt
http://192.168.100.12/moodle/login/index.php?testsession=1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=0esiyDseG7&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=0hhN9xE2Fc&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php%3Fsearch%3DZAP
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=10OKZGbd36&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689488%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=13Fs6vrPRk&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=1IkaTBj2hK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=2P3now6K09&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=2TKqBIUd3u&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=32JKjGeNCq&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689511%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3C92K5tvZ5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3D8MCl2X4O&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3VkfyKmkso&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3ptCPFEhH4&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3qONrpoAP7&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=5Z9QoIKeSe&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=5gGwzAqVNk&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime

%3D1414689503%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=5pu59cuJq9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689479%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=6MDUAaqM4D&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=6nkpKuTndm&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=71jGMGtegh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1406851200
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=7WD29W3FcA&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=7anbgqLjHg&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=8oCPDFa1E0&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=9QtpqdqCUF&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Ac4llmxqPk&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Bd2CaxQD48&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=BudYqsI9l0&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=C4syehGN2i&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Cgo1fQLYec&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=DQT3Sni0xz&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689460
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=DrK3ReMVAY&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=DsGhyuSrYP&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=EntFIop9z5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fhelp.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=GHVlZtkz9o&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=HBxWcxukzD&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1409529600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=HuSIfGR0hI&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=J8vApIaZ8e&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=JIfiINUWHw&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%

3D1414689507%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=JXMFTWwioK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=JkHRjwCjNt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414800000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=LCadRrFo8s&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1412121600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Me4bKln9YX&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Ftime%3D1414689479
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=N2Xe1JvlaN&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=NdI0p2exGj&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1417392000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OBjuWf1IKK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OntY58a1g8&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OovSDHaU9g&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OrA0WnWYgC&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OzV1uTRryj&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=P7fzKsGDkt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414800000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Q7balo6p3F&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Fforgot_password.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=SUclY8eH33&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=SrGQQXPMDt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=U4IR5jQzr1&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689512%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=UV50k2FvUN&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=UwyszpZg6Q&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=VIQ0c9Mjvg&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=WhEI5eq960&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=XFgUI37sbh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3

Dmonth%26amp%3Btime%3D1414689461
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=YTKp0gcoyt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Ynvmd2facf&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Zfs2HfFjuh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=aJoLithWWK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=aqVdmXiJ4q&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=bG9QOWAbzc&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689505%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=c6mKfGRNPJ&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689484%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=cDm9t3f0Se&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689479%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=cOxd8R92z9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=cudCvJMP8T&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689459
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=dXEn6sAZDz&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fhelp.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=dg8bLbOlGr&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689461
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=dx4iXYhaii&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=eAmH95KoTG&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=ekwxl5MeTh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hY8HgMuAxZ&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hkGQTmsvmN&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689459
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hpTB5qzzxL&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php%3Fsearch%3DZAP
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hyx8uXEWFt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=j3gq0Uhgta&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=kVLAm2V

YfC&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=kjid8hxhRd&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=lCZJxaC2iX&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=mp3zxgfni6&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689479%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=n2Wd8W46k4&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=n3H4FklMwv&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=oGtkLPnx80&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=p9ejeS4W4r&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Fforgot_password.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=qJYclqfF3F&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=qJukNVmSbz&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=r6M9BRRri2&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=rKjHl9yNod&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php%3Fsearch%3DZAP
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=rQ1PJQAwv2&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=rfiLuoStl2&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689461
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=s8hIQYVMCx&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=sRCbQ4s5r9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=strkMFe5pb&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=uBlLC1OkZL&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414800000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=uYJicmPJw5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Fforgot_password.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=utdOlcRmJQ&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=vonb7fSSmc&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1409529600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=w6fJiBdrca&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=wQ3EAqTS

8Y&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689479
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=wpCHvrGdVM&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=xUxSwXx9w9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689460
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=xXlyVCiheG&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=yQosNMJRkE&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=ySwf17uhk5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689460
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=zPVMiamWO4&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689459
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=zWeKF3VCtH&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1409529600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=zlShTIMiBM&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/calendar/export.php
http://192.168.100.12/moodle/login/forgot_password.php
http://192.168.100.12/moodle/login/index.php

APÊNDICE 3: Casos de Abuso

Aplicação	Moodle					Versão:		2.7
Nome	Referência Insegura e Direta a Objetos							
Autor	Rodrigo Ronner					Data:		02/08/2014
Ferramenta	*Nikto*							
Nível Alerta	Informação	X	Baixa		Média		Alta		Total = 0

Vulnerabilidade Encontrada, breve descrição:

Não foram encontradas referência insegura direta a objetos.

APÊNDICE 4: Casos de Abuso

Aplicação	Moodle	Versão:	2.7

Nome	Configuração Incorreta de Segurança								
Autor	Rodrigo Ronner						Data:		02/08/2014
Descrição	*X-Content-Type-Options header missing*								
Ferramenta	OWASP-ZAP								
Nível Alerta	Informação	X	Baixa		Média		Alta		Total = 262

Vulnerabilidade Encontrada, breve descrição:

O cabeçalho *Anti-MIME-Sniffing header X-Content-Type-Options* não foi definido como *'nosniff'*. Isso permite que versões mais antigas do *Internet Explorer* e *Chrome* possam executar *MIME-Sniffing* no corpo da resposta, podendo causar repostas no corpo a ser interpretado e exibido como um tipo de conteúdo que não seja o tipo de conteúdo declarado.

Requisição:

GET http://192.168.100.12/moodle HTTP/1.1

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0;)Pragma: no-cache

Cache-control: no-cache

Content-Length: 0

Host: 192.168.100.12

Resposta:

HTTP/1.1 200 OK

Date: Sat, 02 Aug 2014 22:13:16 GMT

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_auth_kerb/5.4 mod_fcgid/2.3.9mod_nss/2.4.6 NSS/3.15.4 Basic ECC PHP/5.4.16 mod_wsgi/3.4 Python/2.7.5

X-Powered-By: PHP/5.4.16

Set-Cookie: MoodleSession=sokib725pfgvfs45ka53fbc0t2; path=/moodle/

Expires: Mon, 20 Aug 1969 09:23:00 GMT

Cache-Control: no-store, no-cache, must-revalidate

Pragma: no-cache

Content-Language: pt

Content-Script-Type: text/javascript

Content-Style-Type: text/css

X-UA-Compatible: IE=edge

Cache-Control: post-check=0, pre-check=0, no-transform

Last-Modified: Sat, 02 Aug 2014 22:13:17 GMT

Accept-Ranges: none

X-Frame-Options: sameorigin

Content-Type: text/html; charset=utf-8

Solução indicada:

Setar os cabeçalho *X-Content-Type-Options* para '*nosniff*' para todas as páginas da web. Se possível, garantir que o usuário final usa um navegador compatível com os padrões web modernos, que não executem *MIME-sniffing* em tudo, ou que possa ser configurado no servidor de aplicações web para não executarem *MIME-sniffing*.

URLs Testadas:

http://192.168.100.12/moodle
http://192.168.100.12/moodle/
http://192.168.100.12/moodle/?lang=en
http://192.168.100.12/moodle/?lang=pt
http://192.168.100.12/moodle/?time=1401580800
http://192.168.100.12/moodle/?time=1404172800
http://192.168.100.12/moodle/?time=1406851200
http://192.168.100.12/moodle/?time=1409529600
http://192.168.100.12/moodle/?time=1412121600
http://192.168.100.12/moodle/?time=1414800000
http://192.168.100.12/moodle/?time=1417392000
http://192.168.100.12/moodle/?time=1420070400
http://192.168.100.12/moodle/?time=1422748800
http://192.168.100.12/moodle/calendar/export.php
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689479
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689484
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689488
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689503
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689505
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689507
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689511
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=en&time=1414689512
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689479
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689484
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689488
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689503
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689505
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689507
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689511
http://192.168.100.12/moodle/calendar/export.php?course=1&lang=pt&time=1414689512
http://192.168.100.12/moodle/calendar/export_execute.php?authtoken=c75ba0e368def19ab

4220fd65c2adae80bc92148&cal_d&cal_m&cal_y&generateurl=Get+calendar+URL&preset_time=weeknow&preset_what=all&userid=0
http://192.168.100.12/moodle/calendar/export_execute.php?authtoken=c75ba0e368def19ab4220fd65c2adae80bc92148&cal_d&cal_m&cal_y&generateurl=Obter+URL+do+calend%C3%A1rio&preset_time=weeknow&preset_what=all&userid=0
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3RpbWU9MTQxNDY4OTQ3OSZjb3Vyc2U9MQ%3D%3D&sesskey=Me4bKln9YX&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3RpbWU9MTQxNDY4OTQ3OSZjb3Vyc2U9MQ%3D%3D&sesskey=Me4bKln9YX&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA2ODUxMjAwJmNvdXJzZT0x&sesskey=71jGMGtegh&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA2ODUxMjAwJmNvdXJzZT0x&sesskey=71jGMGtegh&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=HBxWcxukzD&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=HBxWcxukzD&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=vonb7fSSmc&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=vonb7fSSmc&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=zWeKF3VCtH&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDA5NTI5NjAwJmNvdXJzZT0x&sesskey=zWeKF3VCtH&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=cudCvJMP8T&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=cudCvJMP8T&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=hkGQTmsvmN&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=hkGQTmsvmN&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=zPVMiamWO

4&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDU5JmNvdXJzZT0x&sesskey=zPVMiamWO4&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=DQT3Sni0xz&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=DQT3Sni0xz&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=xUxSwXx9w9&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=xUxSwXx9w9&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=ySwf17uhk5&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYwJmNvdXJzZT0x&sesskey=ySwf17uhk5&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=XFgUI37sbh&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=XFgUI37sbh&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=dg8bLbOlGr&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=dg8bLbOlGr&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=rfiLuoStl2&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDYxJmNvdXJzZT0x&sesskey=rfiLuoStl2&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDc5JmNvdXJzZT0x&sesskey=wQ3EAqTS8Y&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0Njg5NDc5JmNvdXJzZT0x&sesskey=wQ3EAqTS8Y&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=JkHRjwCjNt&var=showcourses

http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=JkHRjwCjNt&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=P7fzKsGDkt&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=P7fzKsGDkt&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=uBlLC1OkZL&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE0ODAwMDAwJmNvdXJzZT0x&sesskey=uBlLC1OkZL&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE3MzkyMDAwJmNvdXJzZT0x&sesskey=NdI0p2exGj&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDE3MzkyMDAwJmNvdXJzZT0x&sesskey=NdI0p2exGj&var=showglobal
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDEyMTIxNjAwJmNvdXJzZT0x&sesskey=LCadRrFo8s&var=showcourses
http://192.168.100.12/moodle/calendar/set.php?return=L2NhbGVuZGFyL3ZpZXcucGhwP3ZpZXc9bW9udGgmdGltZT0xNDEyMTIxNjAwJmNvdXJzZT0x&sesskey=LCadRrFo8s&var=showglobal
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1404172800&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1406851200&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1409529600&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1412121600&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689459&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689460&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689461&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689479&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689479&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689484&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689484&view=upcoming

http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689488&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689488&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689503&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689503&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689505&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689505&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689507&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689507&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689511&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689511&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689512&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414689512&view=upcoming
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1414800000&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1417392000&view=month
http://192.168.100.12/moodle/calendar/view.php?course=1&time=1420070400&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1406851200&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1409529600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1412121600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689459&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689460&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689461&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689479
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414689479&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1414800000&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=en&time=1417392000&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1406851200&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1409529600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1412121600&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689459&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689460&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689461&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689479
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414689479&view=month
http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1414800000&view=month

http://192.168.100.12/moodle/calendar/view.php?lang=pt&time=1417392000&view=month
http://192.168.100.12/moodle/course/index.php
http://192.168.100.12/moodle/course/index.php?lang=en
http://192.168.100.12/moodle/course/index.php?lang=pt
http://192.168.100.12/moodle/course/search.php
http://192.168.100.12/moodle/course/search.php?lang=en
http://192.168.100.12/moodle/course/search.php?lang=en&search=ZAP
http://192.168.100.12/moodle/course/search.php?lang=pt
http://192.168.100.12/moodle/course/search.php?lang=pt&search=ZAP
http://192.168.100.12/moodle/course/search.php?search=ZAP
http://192.168.100.12/moodle/course/view.php?id=1
http://192.168.100.12/moodle/help.php?component=moodle&identifier=cookiesenabled&lang=en
http://192.168.100.12/moodle/help.php?component=moodle&identifier=cookiesenabled&lang=pt
http://192.168.100.12/moodle/help.php?lang=en
http://192.168.100.12/moodle/help.php?lang=pt
http://192.168.100.12/moodle/lib/javascript.php/1406061308/lib/javascript-static.js
http://192.168.100.12/moodle/login/forgot_password.php
http://192.168.100.12/moodle/login/forgot_password.php?lang=en
http://192.168.100.12/moodle/login/forgot_password.php?lang=pt
http://192.168.100.12/moodle/login/index.php
http://192.168.100.12/moodle/login/index.php?lang=en
http://192.168.100.12/moodle/login/index.php?lang=pt
http://192.168.100.12/moodle/login/index.php?testsession=1
http://192.168.100.12/moodle/theme/image.php/_s/clean/core/1406061309/help
http://192.168.100.12/moodle/theme/image.php/_s/clean/core/1406061309/i/warning
http://192.168.100.12/moodle/theme/image.php/_s/clean/core/1406061309/moodlelogo
http://192.168.100.12/moodle/theme/image.php/_s/clean/core/1406061309/t/hide
http://192.168.100.12/moodle/theme/image.php/_s/clean/theme/1406061309/favicon
http://192.168.100.12/moodle/theme/javascript.php/clean/1406061309/footer
http://192.168.100.12/moodle/theme/javascript.php/clean/1406061309/head
http://192.168.100.12/moodle/theme/styles.php/_s/clean/1406061309/all/chunk0
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=0esiyDseG7&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=0hhN9xE2Fc&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php%3Fsearch%3DZAP
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=10OKZGbd36&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689488%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=13Fs6vrPRk&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=1IkaTBj2hK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=2P3now6K09&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=2TKqBIUd3u&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F

http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=32JKjGeNCq&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689511%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3C92K5tvZ5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3D8MCl2X4O&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3VkfyKmkso&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3ptCPFEhH4&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=3qONrpoAP7&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=5Z9QoIKeSe&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=5gGwzAqVNk&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689503%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=5pu59cuJq9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689479%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=6MDUAaqM4D&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=6nkpKuTndm&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=71jGMGtegh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1406851200
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=7WD29W3FcA&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=7anbgqLjHg&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=8oCPDFa1E0&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=9QtpqdqCUF&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Ac4llmxqPk&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Bd2CaxQD48&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=BudYqsI9l0&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=C4syehGN2i&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Cgo1fQLYec&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=DQT3Sni0xz&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689460

http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=DrK3ReMVAY&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=DsGhyuSrYP&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=EntFIop9z5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fhelp.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=GHVlZtkz9o&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=HBxWcxukzD&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1409529600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=HuSIfGR0hI&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=J8vApIaZ8e&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=JIfiINUWHw&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689507%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=JXMFTWwioK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=JkHRjwCjNt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414800000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=LCadRrFo8s&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1412121600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Me4bKln9YX&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Ftime%3D1414689479
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=N2Xe1JvlaN&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=NdI0p2exGj&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1417392000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OBjuWf1IKK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OntY58a1g8&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OovSDHaU9g&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OrA0WnWYgC&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=OzV1uTRryj&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=P7fzKsGDkt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414800000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Q7balo6p3F&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Fforgot_password.php

http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=SUclY8eH33&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=SrGQQXPMDt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=U4IR5jQzr1&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689512%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=UV50k2FvUN&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=UwyszpZg6Q&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=VIQ0c9Mjvg&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=WhEI5eq960&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=XFgUI37sbh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689461
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=YTKp0gcoyt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Ynvmd2facf&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=Zfs2HfFjuh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=aJoLithWWK&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=aqVdmXiJ4q&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=bG9QOWAbzc&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689505%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=c6mKfGRNPJ&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689484%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=cDm9t3f0Se&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689479%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=cOxd8R92z9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=cudCvJMP8T&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689459
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=dXEn6sAZDz&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fhelp.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=dg8bLbOlGr&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689461
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=dx4iXYhaii&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F

http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=eAmH95KoTG&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=ekwxl5MeTh&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hY8HgMuAxZ&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hkGQTmsvmN&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689459
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hpTB5qzzxL&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php%3Fsearch%3DZAP
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=hyx8uXEWFt&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=j3gq0Uhgta&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=kVLAm2VYfC&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=kjid8hxhRd&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=lCZJxaC2iX&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=mp3zxgfni6&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fexport.php%3Ftime%3D1414689479%26amp%3Bcourse%3D1
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=n2Wd8W46k4&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=n3H4FklMwv&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=oGtkLPnx80&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=p9ejeS4W4r&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Fforgot_password.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=qJYclqfF3F&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=qJukNVmSbz&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=r6M9BRRri2&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=rKjHl9yNod&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcourse%2Fsearch.php%3Fsearch%3DZAP
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=rQ1PJQAwv2&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=rfiLuoStl2&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689461
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=s8hIQYVMCx&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F

http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=sRCbQ4s5r9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=strkMFe5pb&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=uBlLC1OkZL&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414800000
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=uYJicmPJw5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Fforgot_password.php
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=utdOlcRmJQ&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=vonb7fSSmc&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1409529600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=w6fJiBdrca&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=wQ3EAqTS8Y&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689479
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=wpCHvrGdVM&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=xUxSwXx9w9&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689460
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=xXlyVCiheG&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=yQosNMJRkE&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2F
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=ySwf17uhk5&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689460
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=zPVMiamWO4&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1414689459
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=zWeKF3VCtH&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Fcalendar%2Fview.php%3Fview%3Dmonth%26amp%3Btime%3D1409529600
http://192.168.100.12/moodle/theme/switchdevice.php?device=default&sesskey=zlShTIMiBM&url=http%3A%2F%2F192.168.100.12%2Fmoodle%2Flogin%2Findex.php
http://192.168.100.12/moodle/theme/yui_combo.php?rollup/1406061308/mcore-min.js&rollup/3.15.0_1/yui-moodlesimple-min.js
http://192.168.100.12/moodle/theme/yui_combo.php?rollup/3.15.0/yui-moodlesimple-min.css
http://192.168.100.12/moodle/calendar/export.php
http://192.168.100.12/moodle/login/forgot_password.php
http://192.168.100.12/moodle/login/index.php
http://192.168.100.12/moodle

APÊNDICE 5: Casos de Abuso

Aplicação	Moodle						Versão:		2.7
Nome	Exposição de Dados Sensíveis								
Autor	Rodrigo Ronner						Data:		02/08/2014
Descrição	*Password Autocomplete in browser*								
Ferramenta	*OWASP-ZAP*								
Nível Alerta	Informação		Baixa	X	Média		Alta		Total = 4

Vulnerabilidade Encontrada, breve descrição:

AUTOCOMPLETE não está desabilitado HTML *FORM/INPUT* elementos contendo *password* de acesso. Senhas podem ser armazenadas e depois recuperadas.

Requisição:

GET http://192.168.100.12/moodle/login/index.php HTTP/1.1

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0;)

Pragma: no-cache

Cache-control: no-cache

Host: 192.168.100.12

Resposta:

HTTP/1.1 200 OK

Date: Sat, 02 Aug 2014 22:13:23 GMT

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_auth_kerb/5.4 mod_fcgid/2.3.9 mod_nss/2.4.6 NSS/3.15.4 Basic ECC PHP/5.4.16 mod_wsgi/3.4 Python/2.7.5

X-Powered-By: PHP/5.4.16

Set-Cookie: MoodleSession=qobhrmka7p8815uujbve5tl8t0; path=/moodle/

Expires:

Cache-Control: private, pre-check=0, post-check=0, max-age=0, no-transform

Pragma: no-cache

Content-Language: pt

Content-Script-Type: text/javascript

Content-Style-Type: text/css

X-UA-Compatible: IE=edge

Accept-Ranges: none X-Frame-Options: sameorigin Content-Type: text/html; charset=utf-8

Ataque:

<input type="password" name="password" id="password" size="15" value="" />

Solução indicada:

Deixar AUTOCOMPLETE nos formulário de entrada com a opção de senha AUTOCOMPLETE='OFF'.

URLs Testadas:

http://192.168.100.12/moodle/login/index.php http://192.168.100.12/moodle/login/index.php?lang=en http://192.168.100.12/moodle/login/index.php?lang=pt http://192.168.100.12/moodle/login/index.php?testsession=1

APÊNDICE 6: Casos de Abuso

Aplicação	Moodle						Versão:		2.7
Nome	*Cross-Site Request Forgery (CSRF)*								
Autor	Rodrigo Ronner						Data:		02/08/2014
Ferramenta	*WebScarab*								
Nível Alerta	Informação	X	Baixa		Média		Alta		Total = 0

Vulnerabilidade Encontrada, breve descrição:

Não foram encontradas falhas relacionadas à *Cross-Site Request Forgery (CSRF).*

APÊNDICE 7: Casos de Abuso

Aplicação	Moodle	Versão:	2.7
Nome	Utilização de Componentes Vulneráveis Conhecidos		
Autor	Rodrigo Ronner	Data:	02/08/2014

Ferramenta	*Spidering*								
Nível Alerta	Informação	X	Baixa		Média		Alta		Total = 0

Vulnerabilidade Encontrada, breve descrição:

Não foram encontradas falhas relacionadas à utilização de componentes vulneráveis conhecidos

APÊNDICE 8: Casos de Abuso

Aplicação	Moodle						Versão:		2.7
Nome	Redirecionamentos e Encaminhamentos Inválidos								
Autor	Rodrigo Ronner						Data:		02/08/2014
Ferramenta	*OWASP-ZAP*								
Nível Alerta	Informação	X	Baixa		Média		Alta		Total = 0

Vulnerabilidade Encontrada, breve descrição:

Não foram encontradas falhas relacionadas a Redirecionamentos e Encaminhamentos Inválidos

APÊNDICE 9: Casos de Abuso

Aplicação	Moodle						Versão:		2.7
Nome	Falta de Função para Controle do Nível de Acesso								
Autor	Rodrigo Ronner						Data:		02/08/2014
Descrição	*X-Frame-Options header not set*								
Ferramenta	*OWASP-ZAP*								
Nível Alerta	Informação		Baixa	X	Média		Alta		Total = 11

Vulnerabilidade Encontrada, breve descrição:

X-Frame-Options header não está incluído na resposta HTTP para proteger contra ataques '*clickjacking*'.

Requisição:

GET http://192.168.100.12/moodle HTTP/1.1

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0;)

Pragma: no-cache

Cache-control: no-cache

Content-Length: 0

Host: 192.168.100.12

Resposta:

HTTP/1.1 301 Moved Permanently

Date: Sat, 02 Aug 2014 22:13:18 GMT

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_auth_kerb/5.4 mod_fcgid/2.3.9 mod_nss/2.4.6 NSS/3.15.4 Basic ECC PHP/5.4.16 mod_wsgi/3.4 Python/2.7.5

Location: http://192.168.100.12/moodle/

Content-Length: 237

Content-Type: text/html; charset=iso-8859-12

Solução indicada:

A maioria dos navegadores modernos suportam *X-Frame-Options HTTP header*. Certifique-se de que está definido em todas as páginas retornadas pelo seu site (se você espera que a página a ser enquadrado apenas por páginas no servidor (por exemplo, é parte de um *FRAMESET*), então você vai querer usar *SAMEORIGIN*, caso contrário, você deve negar. *ALLOW-FROM* permite que sites específicos para enquadrar a página web em navegadores suportados).

URls Testadas:

http://192.168.100.12/moodle
http://192.168.100.12/moodle/calendar/export_execute.php?authtoken=c75ba0e368def19ab4220fd65c2adae80bc92148&cal_d&cal_m&cal_y&generateurl=Get+calendar+URL&preset_time=weeknow&preset_what=all&userid=0
http://192.168.100.12/moodle/calendar/export_execute.php?authtoken=c75ba0e368def19ab4220fd65c2adae80bc92148&cal_d&cal_m&cal_y&generateurl=Obter+URL+do+calend%C3%A1rio&preset_time=weeknow&preset_what=all&userid=0
http://192.168.100.12/moodle/course/view.php?id=1
http://192.168.100.12/moodle/lib/javascript.php/1406061308/lib/javascript-static.js
http://192.168.100.12/moodle/theme/javascript.php/clean/1406061309/footer
http://192.168.100.12/moodle/theme/javascript.php/clean/1406061309/head
http://192.168.100.12/moodle/theme/styles.php/_s/clean/1406061309/all/chunk0
http://192.168.100.12/moodle/theme/yui_combo.php?rollup/1406061308/mcore-min.js&rollup/3.15.0_1/yui-moodlesimple-min.js
http://192.168.100.12/moodle/theme/yui_combo.php?rollup/3.15.0/yui-moodlesimple-min.css

http://192.168.100.12/moodle/login/index.php

APÊNDICE 10: Casos de Abuso

Aplicação	Moodle					Versão:		2.7	
Nome	Cross-Site Scripting (XSS)								
Autor	Rodrigo Ronner					Data:		02/08/2014	
Ferramenta	OWASP-ZAP								
Nível Alerta	Informação		Baixa		Média		Alta	X	Total = 2

Vulnerabilidade Encontrada, breve descrição:

O ataque de *Cross-site scripting (XSS)* consiste em uma vulnerabilidade causada pela falha nas validações dos parâmetros de entrada do usuário e resposta do servidor na aplicação web. Este ataque permite que código HTML seja inserido de maneira arbitrária no navegador do usuário alvo.

Através de um XSS, o cracker injeta códigos *JavaScript* em um campo texto de uma página já existente e este *JavaScript* é apresentado para outros usuários, porque persiste na página.

O impacto da vulnerabilidade de XSS é principalmente sua imagem e a possibilidade de utilização da falha para a distribuição de *phishing* e facilitação de fraudes.

Dentre as principais consequências para o usuário afetado, incluem:

- Sequestro de sessão de usuários;
- Alteração do código HTML do aplicativo (visível somente do lado do cliente);
- Redirecionar o usuário para sites maliciosos;
- Alteração do objeto DOM para captura de dados ou envio de *malware*.

Requisição:

GET http://192.168.100.12/moodle/mod/url/view.php?id=4

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0;)

Pragma: no-cache

Cache-control: no-cache

Host: 192.168.100.12

Resposta:

HTTP/1.1 200 OK

Date: Sat, 02 Aug 2014 23:05:50 GMT

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_auth_kerb/5.4 mod_fcgid/2.3.9 mod_nss/2.4.6 NSS/3.15.4 Basic ECC PHP/5.4.16 mod_wsgi/3.4 Python/2.7.5

X-Powered-By: PHP/5.4.16

Set-Cookie: MoodleSession=d0re6ebdforfjucc3sbdblm2j7; path=/moodle/

Cache-Control: private, pre-check=0, post-check=0, max-age=0, no-transform

Pragma: no-cache

Content-Language: pt

Content-Script-Type: text/javascript

Content-Style-Type: text/css

X-UA-Compatible: IE=edge

Accept-Ranges: none

X-Frame-Options: sameorigin

Content-Type: text/html; charset=utf-8

Ataque:

javascript: (‘XSS Diga Olá!’);

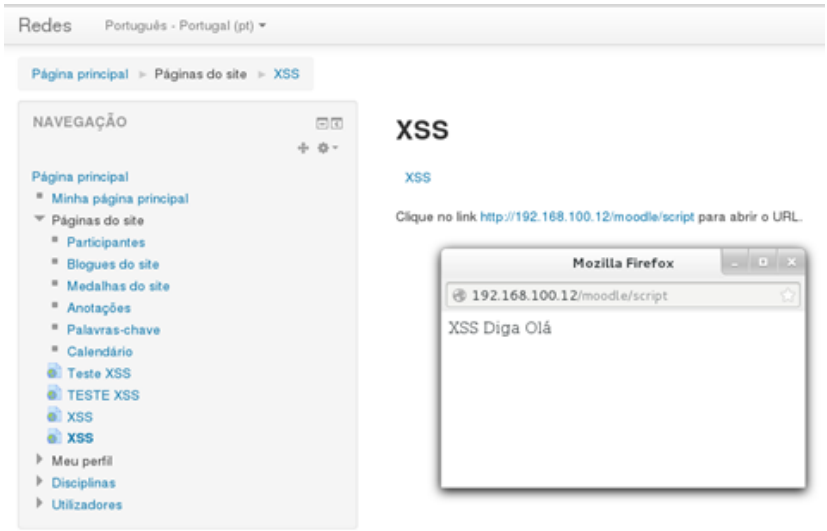


Figura 21 - Execução código em *javascript* por meio de uma vulnerabilidade.

Solução indicada:

Fases: Análise e Especificação

Entender o contexto em que serão utilizados os seus dados e a codificação. Isto é especialmente importante quando a transmissão de dados ocorre entre componentes diferentes, ou ao gerar saídas que podem conter várias codificações, ao mesmo tempo, como páginas da Web ou mensagens de correio de várias partes.

Para cada página da Web que é gerado deve-se usar e especificar uma codificação de caracteres, tais como ISO-8859-1 ou UTF-8. Quando uma codificação não for especificada, o navegador Web pode escolher uma codificação diferente, adivinhando qual codificação está realmente sendo usado pela página web. Isso pode fazer com que o navegador Web possa tratar certas sequências como especial, a abertura do cliente para ataques sutis XSS.

Para ajudar a atenuar ataques XSS contra cookie de sessão do usuário, defina o cookie de sessão para ser *HttpOnly*. Em navegadores que suportam o recurso *HttpOnly* (como versões mais recentes do Internet Explorer e Firefox), este atributo pode impedir cookie de sessão do usuário de ser acessível para os scripts do lado do cliente mal-intencionados que usam *document.cookie*. Esta não é uma solução completa, desde *HttpOnly* não é suportado por todos os navegadores. Mais importante ainda, tecnologias *XMLHTTPRequest* e outro navegador poderoso fornecer acesso de leitura a cabeçalhos HTTP, incluindo o cabeçalho *Set-Cookie* em que a bandeira *HttpOnly* está definido.

Certifique-se de executar a validação de entrada em interfaces bem definidas dentro do aplicativo. Isso ajudará a proteger o aplicativo, mesmo se um componente é reutilizado ou transferi para outros locais.

Fase: Desenvolvimento

Uma biblioteca ou *framewor*k que não permitisse que essa fraqueza pudesse ocorrer. Exemplos de bibliotecas e *frameworks* que tornam mais fácil gerar a saída corretamente codificada incluem biblioteca da *Microsoft Anti-XSS*, o módulo de codificação OWASP ESAPI e *Apache Wicket*.

Se possível, sugere-se uso mecanismos estruturados que aplicam automaticamente a separação entre dados e código. Estes mecanismos podem ser capazes de fornecer o relevante citar, codificação e validação automaticamente, em vez de depender do desenvolvedor para fornecer essa capacidade em cada ponto onde a produção foi gerada.

URls Testadas:

http://192.168.100.12/moodle/mod/url/view.php?id=4
http://192.168.100.12/moodle/mod/url/view.php?id=6