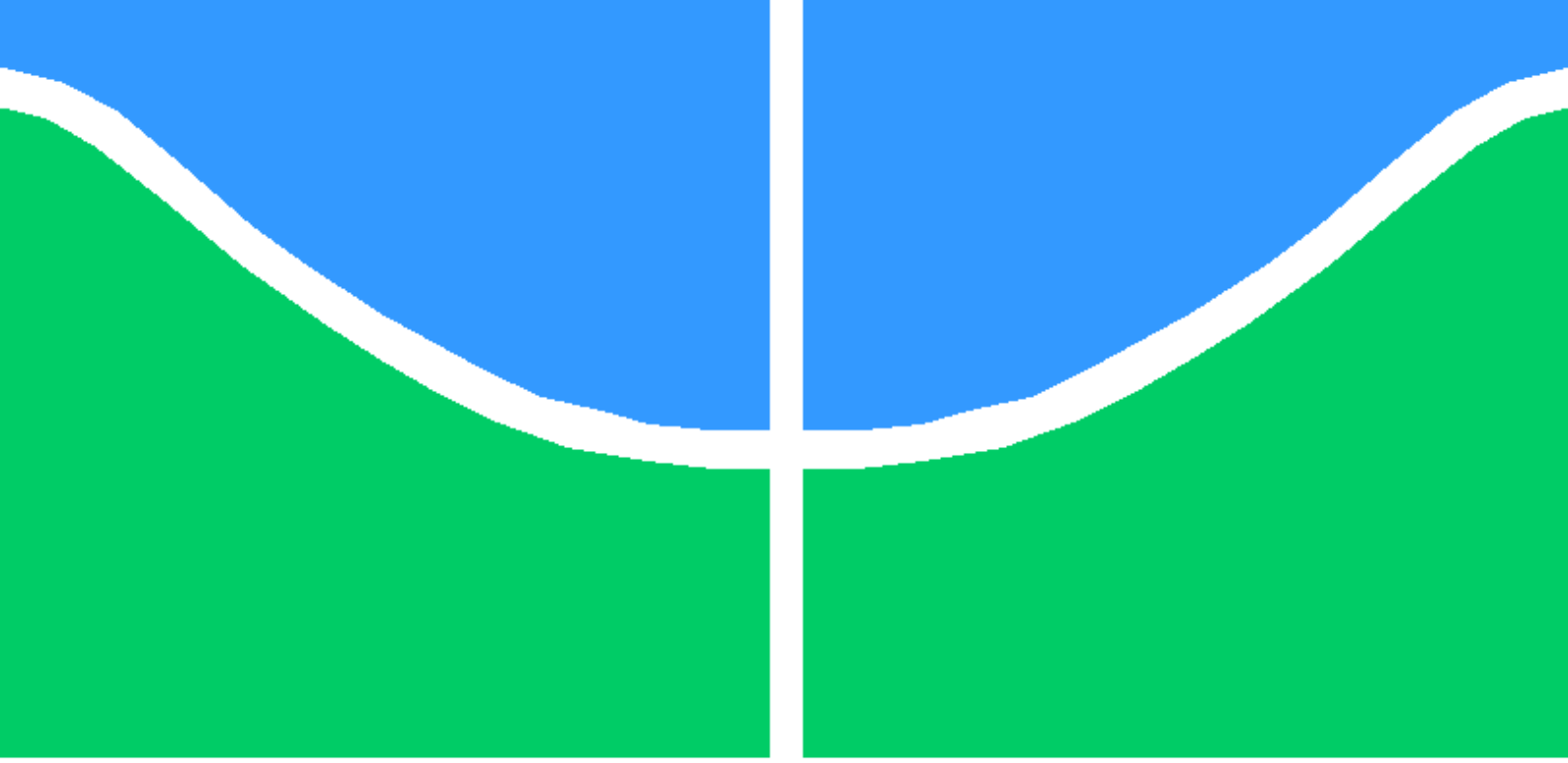

UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES

JOÃO JOSÉ COSTA GONDIM

TESE DE DOUTORADO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES

JOÃO JOSÉ COSTA GONDIM

Orientador: PROF. DR. ANDERSON CLAYTON ALVES NASCIMENTO, ENE/UNB

Co-orientador: PROF. DR. ROBSON OLIVEIRA DE ALBUQUERQUE, ENE/UNB

TESE DE DOUTORADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO PPGENE.TD - 118/2017
BRASÍLIA-DF, 5 DE JULHO DE 2017.

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES

JOÃO JOSÉ COSTA GONDIM

TESE DE DOUTORADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR.

APROVADA POR:

Anderson C A Nascimento

ANDERSON CLAYTON ALVES NASCIMENTO, Dr., ENE/UNB
(ORIENTADOR)

RAFAEL TIMÓTEO DE SOUSA JÚNIOR, Dr., ENE/UNB
(EXAMINADOR INTERNO)

ANDRÉ RICARDO ABED GRÉGIO, Dr., UFPR
(EXAMINADOR EXTERNO)

ANDRÉ COSTA DRUMMOND, Dr., CIC/UNB
(EXAMINADOR EXTERNO)

Brasília, 05 de julho de 2017.

FICHA CATALOGRÁFICA
JOÃO JOSÉ COSTA GONDIM
UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES
2017xv, 149p., 201x297 mm
(ENE/FT/UnB, Doutor, Engenharia Elétrica, 2017)
Tese de Doutorado - Universidade de Brasília
Faculdade de Tecnologia - Departamento de Engenharia Elétrica

REFERÊNCIA BIBLIOGRÁFICA

JOÃO JOSÉ COSTA GONDIM (2017) UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES. Tese de Doutorado em Engenharia Elétrica, Publicação 118/2017, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 149p.

CESSÃO DE DIREITOS

AUTOR: JOÃO JOSÉ COSTA GONDIM
TÍTULO: UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES.
GRAU: Doutor ANO: 2017

JOÃO JOSÉ COSTA GONDIM
Prédio CIC/EST, Campus Darci Ribeiro

Agradecimentos

Ao longo deste processo, gostaria de registrar a colaboração e apoio recebidos. Inicialmente agradeço ao meu orientador e amigo Anderson Nascimento pela confiança, compreensão e paciência com relação a mim. Agradeço também ao meu também amigo e coorientador Robson Albuquerque pelo apoio, principalmente nesta fase final do processo. Registro as discussões com Célia Ralha no que tange a modelagem de processos. Devo tmbém registrar o suporte recebido de Max Xavier e Tiago Medeiros.

Agradeço também aos profissionais atuantes na área de segurança, tanto no setor público como privado, companheiros de trincheiras, Robson Albuquerque, Otávio Cunha, Evandro Hora, Marco "Kiko" Carnut e Aldo Albuquerque Segundo pelas discussões cheias de *insights* e trocas de experiências que levaram a uma percepção mais ampla e realista da área de segurança como um todo.

Por fim, para não cometer alguma omissão, agradeço a todos que de alguma forma me ajudaram neste processo.

Dedicatória

Aos meus filhos Luiza, Isaac e Gabriel

ABSTRACT

A METHODOLOGICAL APPROACH FOR ACTIVE VULNERABILITY ASSESSMENT

Author: João José Costa Gondim

Supervisor: Anderson Clayton Alves Nascimento
Co-Supervisor: Robson Oliveira de Albuquerque

Programa de Pós-graduação em Engenharia Elétrica

Brasília, July of 2017

Penetration Tests (pentest) methodologies are the basis for actively prospecting vulnerabilities (PAV) in computer systems. However, there are several conceptual gaps in current pentest most widely used methodologies. They either lack methodological rigor or are designed as part of other information security processes, like risk assessment ou security audit. They are also either limited in their scope or in their process definition. They either do not define test execution at all, or do it with impacting shortcomings. With those motivations, DOTA, a decision oriented tool agnostic pentest methodology is developed and presented. It aims to define standardized procedures for vulnerability assessment based on the strategy and the decision flow during execution of pentests. The methodology interconnects planning procedures with the execution of computational scenarios through the composition of six closely adjacent phases. Tests (attacks) are modelled using a general decision cycle, so that DOTA gains in generality and aplicability. The methodology was applied for intrusion and unavailability, with full runs on controlled real world scenarios. Results obtained were more complete, precise and repeatable.

RESUMO

UMA ABORDAGEM METODOLÓGICA PARA PROSPECÇÃO ATIVA DE VULNERABILIDADES

Autor: João José Costa Gondim

Supervisor: Anderson Clayton Alves Nascimento
Co-Supervisor: Robson Oliveira de Albuquerque

Programa de Pós-graduação em Engenharia Elétrica

Brasília, Julho de 2017

Metodologias de Testes de Penetração (*Penetration Tests - pentest*) são a base para a prospecção ativa de vulnerabilidades (PAV) em sistemas computacionais. Entretanto, há várias lacunas conceituais nas metodologias mais amplamente utilizadas. As lacunas podem ser por falta de rigor metodológico ou por serem projetadas como parte de processos mais geria em segurança da informação, como avaliação de risco ou auditoria de segurança; ou por limitações de escopo ou na definição de seus processos; ou por não definir e modelar a execução dos testes, ou ainda por fazê-lo com limitações. Com tais motivações, DOTA, uma metodologia de *pentest*, tendo por objetivo definir procedimentos padronizados para prospecção de vulnerabilidades baseada na estratégia e no fluxo das decisões durante a execução de *pentests*. A metodologia agrega ações de planejamento com a execução de testes em diferentes cenários pela composição de seis fases sequenciais. Os testes (ataques) são modelados segundo um ciclo de decisão geral, de forma que DOTA ganha em generalidade e aplicabilidade. A metodologia foi aplicada com finalidades de intrusão e indisponibilidade, envolvendo cenários reais. O s resultados obtidos foram mais completos, precisos e repetíveis.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE TERMOS E SIGLAS

AR-DDoS	DDoS por reflexão amplificada
DDoS	DoS distribuído
DoS	*Denial of Service*
DOTA	*Decision Oriented and Tool Agnostic*
DSS	*Data Secutity Standard*
ENISA	*European Union Agency for Network and Information Security*
IoT	Internet das Coisas
ISP	Provedores de Serviços de Internet
NAT	*Network Address Translation*
NIST	*National Institute of Standards and Technology*
NVD	National Vulnerability Database
OADA	Orientar-Analisar-Decidir-Agir
OODA	*Observe-Orient-Decide-Act*
OSSTMM	*Open Source Security Test Methodology Manual*
OWASP	*Open Source Application Security Project*
PAV	Prospecção Ativa de Vulnerabilidades
PCI-DSS	*PCI Data Secutity Standard*
PCI-SSC	*Payment Card Industry Security Standards Council*
PDCA	*Plan-Do-Check-Act*
PIN	*Personal Identification Number*
PTES	*Penetration Testing Execution Standard*

RoE	*Rules of Engagement*
SCADA	*Supervisory Control and Data Aquisition Systems*
SDLC	*Software Development Life Cycle*
WSN	*Wireless Sensor Network*
XSS	*Cross Site Scripting*
OpSec	Segurança Operacional

Capítulo 1

Introdução

A disseminação do uso das tecnologias empregadas na Internet trouxe consigo um dilema:

- se por um lado estas tecnologias viabilizam a possibilidade de oferecer serviços e recursos a baixo custo, acarretando grande economia de escala, não podendo ser desprezadas;
- por outro, estas tecnologias não foram concebidas com preocupações de segurança, sendo frequentes os relatos de incidentes envolvendo comprometimento de sistemas e das informações lá presentes.

Mesmo antes do último boom de expansão da Internet, no início dos anos 90, os incidentes de segurança já ocorriam, e.g. [Spafford 1989] e [Stoll 1989], se agravando com o seu crescimento e o consequente incremento da disponibilidade de aplicações nas áreas comerciais e de governo que elevam as preocupações com as informações e sistemas disponibilizados. Se, por um lado, as organizações não podem se furtar dos benefícios da Internet, por outro elas temem que seus sistemas sejam invadidos, adulterados ou atacados.

Várias organizações entenderam que uma forma eficiente de avaliar os riscos e ameaças a suas informações e sistemas seria aplicar as mesmas técnicas, procedimentos e ferramentas usadas pelos atacantes para testar e avaliar, sob consentimento e em condições controladas, o grau de insegurança de seus sistemas.

Existem várias formas de avaliação de segurança de sistemas de informação. Podem ser citadas avaliações de vulnerabilidades que consistem de testes para identificação de vulnerabilidades, ou as varreduras de rede. Tipicamente estes procedimentos produzem listas identificando sistemas e componentes em uma rede e enumerando suas vulnerabilidades.

Entretanto, tais abordagens não incluem o ponto de vista do atacante para uma avaliação mais precisa. Uma diferença importante entre estas outras formas de avaliação de segurança de sistemas de informação e a Prospecção Ativa de Vulnerabilidades (PAV), que inclui os

testes de penetração, está no fato de que seu foco não se restringe apenas à coleta de listas de vulnerabilidades de sistemas de informação, com eventuais recomendações para sua correção. Na verdade, o esforço está na confirmação da efetiva presença das vulnerabilidades e subsequente exploração para uma avaliação mais precisa das implicações dessa presença no contexto do ambiente sob exame. Em resumo, o foco está em se ter uma visão clara do significado das vulnerabilidades encontradas durante a execução do processo de testes de penetração aplicadas ao contexto da segurança da informação da organização. Com estas motivações e preocupações, foi desenvolvida a metodologia apresentada nesta tese. Esta metodologia é motivada pela ênfase na repetibilidade e no fluxo de decisões ao longo do processo, se distanciando da descrição operacional de técnicas e procedimentos, sendo autocontida e independente de outras metodologias.

Um outro ponto importante diz respeito à dualidade dos procedimentos, técnicas e ferramentas que dão suporte a uma metodologia para PAV. Essa dualidade se expressa no fato que este arcabouço técnico aqui usado com fins de melhoria da proteção pode ser empregado em contexto ofensivo, o que é relativamente óbvio, pois é nesse contexto que esse ferramental técnico se origina. Apesar de tal emprego estar fora do escopo de interesse, deve-se registrar que há relatos de várias nações estado se utilizando de tais técnicas para fins militares e de inteligência.

De qualquer forma, a possibilidade de emprego dual de uma metodologia para PAV vem só reforçar o seu desenvolvimento.

1.1 Contribuições

De maneira geral, as principais contribuições deste trabalho são:

- suprir uma lacuna conceitual das metodologias existentes no que concerne os aspectos de decisão nos processos de *pentests*;
- propor e apresentar uma metodologia de *pentests* focada em fluxo de decisão e repetibilidade, que foi testada e validada em situações controladas e próximas de ambientes reais;
- modelar, de forma não procedimental, a execução de testes dentro de um processo de *pentest*, entendidos como ataques cibernéticos controlados e sob consentimento, que é ao mesmo tempo geral, aplicável e capaz de produzir resultados úteis;
- abranger testes e ataques contra a disponibilidade, sob a perspectiva do atacante.

1.2 Objetivos

O objetivo principal deste trabalho é desenvolver uma metodologia para realização de processos de prospecção ativa de vulnerabilidades que dentro da abordagem adotada para sua elaboração, estão presentes as seguintes características:

- explicitação dos critérios de decisão e escolha de procedimentos, técnicas e ferramentas;
- generalidade de escopo, incluindo testes de ameaças à disponibilidade;
- flexibilidade de aplicação;
- independência de ferramentas;
- modelagem geral da execução de testes e ataques.

Sob o foco da construção da metodologia em si, tem-se por objetivos específicos:

- Definir e orientar os processos de execução de testes de penetração em sistemas de informação.
- Padronizar o modus operandi da condução dos testes, produção e apresentação dos resultados.
- Aplicar a metodologia em cenários reais de aplicação, inclusive com testes que incluam indisponibilidade.
- Analisar os resultados e a própria metodologia desenvolvida.

1.3 Motivação e Justificativa

É consenso na comunidade de segurança que ataques constituem uma séria ameaça, havendo várias propostas de como lidar com a situação. Entretanto, não é incomum a avaliação de tais ameaças estar mais na forma de inferência que da constatação empírica de sua gravidade, baseada em fatos concretos que evidenciam danos e impactos.

Normalmente, a forma como a avaliação das ameaças é feita não leva em consideração a perspectiva do atacante (*e.g.* [ABNT 2005]). Dessa forma, os resultados obtidos serão no mínimo incompletos. Assim, é necessário incluir este ponto de vista para uma avaliação mais precisa. Essa é a motivação para uma metodologia de avaliação de vulnerabilidades que não só identifique mas também tente ativamente explorar as vulnerabilidades encontradas. A revisão bibliográfica aponta que existem diversas metodologias que definem e guiam o processo da prospecção ativa de vulnerabilidades (PAV). Entretanto, por suas características,

existem lacunas conceituais que fazem com que estas metodologias, por vezes confundam método com procedimento. Como será visto, o foco operacional atrelado a conjuntos de ferramentas pode afetar o ciclo de vida da metodologia.

O aspecto importante a ser considerado é a própria natureza dos chamados testes de penetração (*pentests*). No contexto aqui considerado, os testes de penetração são ataques cibernéticos realizados contra os recursos computacionais de uma organização demandante, sob sua ciência e seu consentimento, para fins de identificação de vulnerabilidades e avaliação de possíveis impactos de sua exploração [SANS Inst. 2003]. Por sua vez, entende-se por ataque cibernético o abuso de infraestrutura, recurso ou aplicação computacional, normalmente realizado remotamente, para fins de acesso e manipulação de informação, tomada de controle ou interrupção de serviço. Assim, um teste realizado em um processo de *pentest* segue as características de um ataque cibernético (como em [Zhuang et al. 2015] e [MI5 2016]).

Deve-se notar que nas metodologias vigentes, a modelagem da execução dos testes ou, como descritos acima, ataques se dá na forma de descrições de procedimentos ou seguindo o *modus operandi* de *malware* e ferramentas de ataque. Essas abordagens, além de restritivas e incompletas, só reforçam a necessidade de realizar tal modelagem.

Há pelo menos dois aspectos determinantes da dinâmica de um teste devem ser considerados na modelagem do processo de execução. O primeiro diz respeito à temporização de sua execução. As ações são executadas em pequenos intervalos de tempo, normalmente frações de segundo, com respostas sendo geradas na mesma escala de grandeza.

O segundo aspecto que está relacionado com as iniciativas de ações e papéis desempenhados pelas partes envolvidas. Normalmente, tem-se um lado atacante, que tem a iniciativa das ações, e um lado defensor, que reage mecanicamente aos estímulos do atacante.

Estes dois aspectos devem ser considerados na modelagem da execução do teste ou ataque pois esta percepção guia o processo de decisão do executor ou atacante: ações rápidas, respostas vegetativas, sem retaliação.

A explicitação de critérios de escolha e decisão, e o próprio encaminhamento destas são pontos extremamente importantes e negligenciados pelas metodologias atuais. Além desses, há uma certa falta de foco e flexibilidade nas metodologias, que não contemplam ataques de negação de serviço.

Assim, o objetivo deste trabalho, que diz respeito à formulação de uma metodologia para testes de penetração, é razoável e oportuno.

1.4 Publicações

Entre os resultados deste trabalho, foram geradas publicações.

A principal publicação se deu em um periódico A1 (Qualis 2013-2016 Eng. IV e

Comp.)sendo diretamente relacionada a esta tese, apresentando um resumo dos suas contribuições e resultados.

Costa Gondim, J. J., de Oliveira Albuquerque, R., Clayton Alves Nascimento, A., García Villalba, L. J., Kim, T. H. (2016). *A Methodological Approach for Assessing Amplified Reflection Distributed Denial of Service on the Internet of Things*. *Sensors*, 16(11), 1855.

Há também uma publicação em conferência A2 (Qualis 2013-2016 Eng. IV e Comp.) relacionada. Ela serviu como prova de conceito da metodologia aplicada à avaliação de impacto de ataque por meio de simulação:

Pacheco, L. A. B., Gondim, J. J., Barreto, P. A. S., Alchieri, E. (2016, October). *Evaluation of Distributed Denial of Service threat in the Internet of Things*. In *Network Computing and Applications (NCA), 2016 IEEE 15th International Symposium on* (pp. 89-92). IEEE.

1.5 Organização deste Documento

Este trabalho aborda as questões referentes aos aspectos metodológicos de processos de prospecção ativa de vulnerabilidades, propondo e aplicando uma metodologia centrada no fluxo de decisões relativas aos testes a se realizar e sua execução, onde são considerados diferentes cenários. A metodologia desenvolvida também é aplicada em situações em que deseja avaliar questões de disponibilidade.

Assim, este documento está organizado em capítulos. No Capítulo 2, é apresentado o estado da arte, onde os trabalhos relacionados são revisados. A descrição do problema, com o *rationale* para uma metodologia de *pentests*, é apresentada juntamente com a abordagem metodológica utilizada na sua construção no Capítulo 3. No capítulo 4, a metodologia em si é apresentada, e seus principais processos, juntamente com a modelagem são descritos. No Capítulo 5, as aplicações da metodologia, com os resultados obtidos, são apresentados e discutidos. A conclusão e as sugestões de trabalhos futuros encerram o documento no Capítulo 6.

Capítulo 2

Estado da Arte e Trabalhos Relacionados

Conforme apresentado no Capítulo 1, este trabalho apresenta dois conjuntos de contribuições: metodológicas e de aplicação da metodologia. Do ponto de vista metodológico, para fundamentar o desenvolvimento da metodologia apresentada à frente, são discutidos os aspectos relativos a metodologias de *pentest*. Do lado da aplicação, a metodologia foi utilizada para avaliar ameaças e possíveis impactos de ataques contra disponibilidade em ambiente da Internet das Coisas (*Internet of Things* - IoT). O presente capítulo aborda estes dois grandes temas sobre os quais este trabalho se fundamenta.

2.1 Testes de Penetração

Tradicionalmente, desde os primórdios da computação, sistemas computacionais são testados de várias formas, com os testes se concentrando em aspectos específicos do comportamento dos sistemas ([Musa 1975], [Hamlet 1977]). Por exemplo, realizam-se testes de funcionalidade, em que se avaliam se as funções do sistema são executadas conforme especificado; ou testes operacionais, que avaliam se os componentes do sistema operam como desejado; ou ainda testes de verificação, aplicados nas fases iniciais do desenvolvimento, visando estabelecer formalmente que a especificação está correta ([NIST 1980], [DoD 1985] e [Neugent et al. 1985]).

A ideia de realizar testes em sistemas computacionais voltados à detecção e identificação de falhas nos mecanismos dos sistemas responsáveis por funções de segurança que não tenham sido encontradas em outras formas de testes, considerando as ações tanto de um usuário autorizado como também de um sem autorização, constitui a motivação original para os testes de penetração. A ideia básica é, após testar as funcionalidades de segurança, estabelecendo sua correta implementação, ir além e tentar contorná-las. Estes testes, normalmente, deveriam ser realizados dentro do processo de desenvolvimento do sistema.

Em [Linde 1975] e [Attanasio, Markstein e Phillips 1976], já se encontram referências ao termo testes de penetração. Entretanto, dadas as características da época, os testes se

restringiam ao acesso local aos sistemas. [Neumann 1977], ainda neste contexto, propõe quatro passos em um teste de penetração:

1. entendimento acerca do sistema;
2. formulação de hipóteses sobre falhas;
3. testes para confirmação ou refutação das hipóteses;
4. extensão de testes com sucesso para formulação de novas hipóteses.

Estes passos com formulação de hipóteses e interação, levam a testes que requerem habilidade e experiência por parte de quem os executa.

Na verdade, [Neumann 1977] vai além e oferece um catálogo de ataques em objetos sensíveis de sistemas, *i.e.* aqueles que são a base da segurança e os controles a eles associados. [Bishop 1986] por sua vez propõe que se definam manuais de referência com as interações e efeitos esperados na interação entre módulos como auxílio aos testadores na seleção de condições de teste e na definição do que testar, uma vez que tais atividades são dificultadas pelas especificidades dos sistemas.

Cabe observar que esta abordagem de inventariar vulnerabilidades, ataques e situações estará presente na forma como os testes de penetração evoluíram.

[Pfleeger, Pfleeger e Theofanos 1989] é pioneiro em explicitamente propor uma abordagem metodológica para os testes de penetração, sendo motivado pelas dificuldades em definir o que deve ser testado. A metodologia proposta tem três passos:

1. identificação de objetos sensíveis;
2. determinação de possíveis pontos de vulnerabilidades para os objetos identificados;
3. realização testes de vulnerabilidades para verificar a adequação dos controles.

Apesar de não ser uma metodologia completa na extensão dos testes, a principal contribuição desta abordagem, além do aspecto puramente metodológico, é a definição de critérios objetivos para definição de escopo e profundidade dos testes.

Deve-se também observar que os testes de penetração, como apresentados, estão inseridos no processo de desenvolvimento dos sistemas, com acesso das equipes de teste à documentação relevante [Pfleeger, Pfleeger e Theofanos 1989]. Esta situação é bem diversa da que evoluiu para o que se tem hoje com equipes de teste totalmente externas e sem informação específica sobre os sistemas testados.

Com o tempo, testes de penetração têm sido aplicados e combinados a várias técnicas e abordagens na segurança de sistemas e da informação. Alguns exemplos de como se deu essa evolução são ilustrados a seguir.

[Nyanchama 2005] relaciona testes de segurança com o gerenciamento de vulnerabilidades e o papel central destes na garantia da segurança corporativa. A motivação é usar os testes para reduzir o impacto de ataques e *malware* sobre as organizações, protegendo-as contra *hackers*, e minimizando o impacto de *malware* que exploram vulnerabilidades conhecidas.

[Miura-Ko e Bambos 2007] introduz um esquema para priorizar vulnerabilidades a serem corrigidas em sistemas de computação e redes. A priorização das vulnerabilidades e dos nós de rede a corrigir têm por base a percentagem de tempo que um invasor aleatório gastaria tentando explorá-los, levando em consideração a topologia da rede e as interações potenciais do nó no cálculo do risco relativo e da prioridade, definindo duas métricas para a segurança de uma rede. Os autores acreditam que essa abordagem pode ser usada como ponto de partida para explorar as estratégias de defesa faz sentido, dadas a topologia da rede e a estratégia de ataque.

[Ten, Liu e Manimaran 2008] estuda o impacto de um ataque cibernético sobre sistemas de controle de supervisão e aquisição de dados (Supervisory Control and Data Aquisition Systems, SCADA), propondo um quadro de avaliação de vulnerabilidade para avaliar sistematicamente as vulnerabilidades dos sistemas SCADA em três níveis: sistema, cenários e pontos de acesso. O método proposto pode ser visto como testes de penetração tendo por alvo os sistemas cibernéticos incorporados com os modelos de *firewall* e senha, o principal modo de proteção na indústria de energia quando da realização do estudo. O impacto de uma potencial intrusão eletrônica é avaliado pela sua potencial perda de carga no sistema de potência e contramedidas são identificadas para melhorar a segurança cibernética.

[Wang e Guo 2009] propõe uma abordagem ontológica para capturar, via testes de segurança, e utilizar os conceitos fundamentais na segurança da informação e sua relação, recuperando dados de vulnerabilidade e raciocínio sobre a causa e o impacto das vulnerabilidades. A ontologia proposta para gerenciamento de vulnerabilidades foi preenchida com todas as vulnerabilidades então presentes na National Vulnerability Database, NVD [O'Reilly 2009], com regras de inferência adicionais, representação de conhecimento e mecanismos de mineração de dados. A ontologia proposta integra vulnerabilidades comuns e seus conceitos relacionados, como ataques e contramedidas.

[Liu et al. 2012] faz uma revisão sobre técnicas de descoberta de vulnerabilidades de software, incluindo análise estática, fuzzing e testes de penetração. Também são abordados modelos de descoberta de vulnerabilidade como um exemplo de métodos de análise de vulnerabilidade de software que complementam técnicas de descoberta de vulnerabilidade, e se analisam as vantagens e desvantagens de cada técnica apresentada. Para os testes de penetração, são apresentadas como vantagens: não geram falso positivos; igualam a descoberta de vulnerabilidade a sua exploração; os resultados se baseiam em ambientes reais de usuário; e expõem vulnerabilidades que outras técnicas ou ferramentas não detectam. Entre as desvantagens, estão: resultados fortemente dependentes da habilidade, experiência e conhecimento dos testadores; e podem causar danos aos sistemas testados.

[Yeo 2013] considera os testes de penetração como uma atividade para determinar se a informação está devidamente protegida, usando as mesmas ferramentas e técnicas que os atacantes, mas de forma controlada com a permissão expressa do alvo. O esboço de uma metodologia refletindo as práticas de mercado mais usadas é apresentada.

[Mainka, Somorovsky e Schwenk 2012] desenvolveu uma ferramenta automatizada de teste de penetração para Web Services chamada WS-Attacker, apresentando uma visão geral de decisões de design e avaliando quatro *frameworks* de Serviços Web e sua resistência contra ataques de *spoofing* WS-Addressing e SOAPAction.

[Marback et al. 2013] propõe uma abordagem de testes de segurança baseada em modelos de ameaça que consiste em três macro atividades: a construção de modelos de ameaça tendo como base árvores de ameaça; a geração de sequências de testes de segurança a partir das árvores de ameaças; e a criação de casos de teste executáveis considerando entradas válidas e inválidas. Para apoiar a abordagem proposta, foram implementadas técnicas de geração de testes de segurança. Também realizou-se um estudo empírico para avaliar a eficácia da abordagem, demonstrando ser eficaz na exposição de vulnerabilidades.

[Großmann et al. 2014] propõe a integração sistemática da análise de risco e testes de segurança para otimizar o processo de teste, bem como a própria avaliação de risco. Assim, pretende-se que o resultado da avaliação de risco, *i.e.*, as vulnerabilidades identificadas, os cenários de ameaça e os incidentes indesejáveis possam ser usados para orientar a identificação do teste e possivelmente complementar os resultados da engenharia de requisitos com informações sistemáticas sobre as ameaças e vulnerabilidades de um sistema e suas probabilidades e consequências. Na verdade, estima-se que a abordagem de teste baseada em risco possa ajudar a otimizar a própria avaliação de risco, obtendo conhecimento empírico sobre a existência de vulnerabilidades, a aplicabilidade e as consequências dos cenários de ameaças e a qualidade das contramedidas.

[Tang 2014] aborda o esforço das organizações em proteger seus ativos mais críticos e identificar e responder a ataques assim que detectados, notando que a melhor maneira de fazê-lo é avaliar as medidas de segurança sob o ponto de vista de um *hacker*, realizando testes de penetração regulares, e segue com o processo de planejamento e execução de um teste e como garantir que ele produza resultados significativos.

[Kim et al. 2013] apresenta um método para construção e gerenciamento de um banco de dados de vulnerabilidades que serviria de base para um sistema gerenciando as vulnerabilidades e avaliando sua gravidade. O sistema produziria insumos para a realização de testes de penetração, bem como dos processos de remediação e resposta.

[Botella et al. 2014] apresenta uma abordagem de teste de segurança original guiada pela avaliação de risco, por meio de cobertura de risco, para executar e automatizar testes de vulnerabilidade para aplicativos web. A abordagem, denominada "Teste de Vulnerabilidade Baseado em Risco", adapta técnicas de Testes Baseados em Modelos, que são usadas atualmente para abordar recursos funcionais. A abordagem adotada também estende as técnicas

de Teste de Vulnerabilidade Baseado em Modelos, conduzindo o processo de teste usando padrões de teste de segurança selecionados a partir de resultados de avaliação de risco. A adaptação de tais técnicas para Testes de Vulnerabilidade Baseados em Risco define novas características neste domínio de investigação. Na abordagem, o modelo utilizado para a geração automatizada de testes capta alguns aspectos comportamentais das aplicações web, mas também inclui os objetivos de testes de vulnerabilidade para guiar o processo de geração de teste.

[Salas e Martins 2014] propõe o uso de duas técnicas de teste de segurança, ou seja, testes de penetração e injeção de falhas (*fault injection*), a fim de emular o ataque XSS (*Cross Site Scripting*) contra *Web Services*. Essa tecnologia, combinada com WS-Security (WSS) e Security Tokens, pode identificar o remetente e garantir o legítimo controle de acesso às mensagens SOAP trocadas. No teste de penetração, se utiliza o *scanner* de vulnerabilidades *soapUI*, que é uma das ferramentas mais populares em testes de penetração para aplicações do tipo. Para a injeção de falhas se desenvolveu a ferramenta WSInject, que apresenta falhas ou erros nos *Web Services* para analisar o comportamento em um ambiente não robusto.

Testes de penetração também foram incorporados ao conceito de exercício cibernético por entidades como MITRE [Jason 2014] e *European Union Agency for Network and Information Security*, ENISA, [ENISA 2011]. Os exercícios cibernéticos têm por finalidade avaliar o nível de prontidão e capacidade de resposta de equipes de operação de segurança de sistemas com respeito a incidentes cibernéticos. Entre outros aspectos, são avaliados a capacidade de comunicação e articulação, e a preparação e treinamento da equipe. Como base para o exercício, está a simulação de situações reais motivadas por ameaças. O exercício em si é executado por quatro equipes: o grupo de controle; a equipe sob avaliação ou treinamento; os atacantes (que utilizam-se de técnicas de testes de penetração); e os monitores, que observam as equipes de ataque e defesa com respeito às ações tomadas, documentando resultados e levando feedback sobre essas atividades para o grupo de controle.

2.2 Metodologias de Testes de Penetração

Há várias metodologias que tem por finalidade a sistematização do processo de prospecção e avaliação de vulnerabilidades, tendo como foco em testes de penetração (*pentests*) e *hacking* ético, que aqui são referenciados de forma genérica como prospecção ativa de vulnerabilidades (PAV). De forma geral, as abordagens adotadas pelas metodologias aqui descritas estão voltadas aos aspectos operacionais da realização dos testes, por vezes envolvendo procedimentos detalhados do uso de ferramentas. Por outro lado, apesar da maioria se propor a ser aplicável de forma geral, algumas tem escopo reduzido ou ainda foco em sistemas específicos.

A seguir, as metodolgias mais relevantes por serem padrões de fato ou de direito, ou ainda por refletirem melhores práticas de mercado bem fundamentadas são apresentadas.

2.2.1 NIST SP800-115

O *National Institute of Standards and Technology* (NIST), tem sua metodologia de *pentest*. O documento NIST Special Publication SP800-115 "Technical Guide to Information Security Testing and Assessment"[Scarfone et al. 2008] faz parte do conjunto de normas e diretivas sobre avaliação de risco, vulnerabilidades e segurança de sistemas, sendo assim subjacente aos documentos NIST Special Publication SP800-53 Rev. 4 [NIST 2015], que recomenda controles de segurança, e NIST Special Publication SP800-53A [NIST 2014], que define diretrizes para avaliação dos controles de segurança.

NIST SP800-115 [Scarfone et al. 2008] é uma metodologia amplamente adotada, sendo frequentemente exigida como requisito de conformidade. Na verdade, é um *framework* para Análise de Risco e provê diretrizes técnicas em avaliações de segurança da informação. Quanto ao seu foco, também é operacional, com um processo definido e detalhados procedimentos e técnicas. Entretanto, seu foco não é a realização dos testes de penetração, que são vistos como mais um recurso no ferramental de gerenciamento de risco.

O primeiro, NIST Special Publication SP800-53, Rev.4, abrange as etapas da Estrutura de Gerenciamento de Risco que tratam da seleção de controles de segurança para sistemas de informação federais (nos EUA) de acordo com os requisitos de segurança do FIPS 200 [FIPS 2006]. Isso inclui selecionar um conjunto inicial de controles de segurança baseados na análise de impacto de pior caso como na FIPS 199 [FIPS 2004], adequando os controles de segurança de linha de base e complementando os controles de segurança com base em uma avaliação organizacional de risco. As regras de segurança abrangem várias áreas, incluindo controle de acesso, resposta a incidentes, continuidade de negócios e recuperação de desastres.

Enquanto o último, NIST Special Publication SP800-53A fornece um conjunto de procedimentos para realizar avaliações de controles de segurança e controles de privacidade empregados dentro de sistemas de informação federais e organizações. Nele, os procedimentos de avaliação, executados em várias fases do ciclo de vida do desenvolvimento do sistema, são consistentes com os controles de segurança e privacidade em NIST SP800-53, Rev.4. Estes procedimentos são personalizáveis e podem ser facilmente adaptados para proporcionar às organizações a flexibilidade necessária para realizar avaliações de controle de segurança e avaliações de controle de privacidade que suportam processos de gerenciamento de riscos organizacionais e que estejam alinhados com a tolerância de risco declarada da organização. Informações sobre a construção de planos eficazes de avaliação de segurança e planos de avaliação de privacidade também são fornecidas juntamente com orientações sobre a análise dos resultados da avaliação.

Estes dois documento por sua vez se inserem no contexto do NIST *Risk Management Framework* (RMF), descrita em NIST Special Publication 800-37 Revision 1, *Guide for Applying the Risk Management Framework to Federal Information Systems: a Security Life Cycle Approach*, que é uma metodologia para implementação de gerenciamento de risco no

nível dos sistemas de informação [NIST 2014]; e NIST SP 800-39, *Managing Information Security Risk* [NIST 2011], que define o gerenciamento de riscos como "o programa e os processos de apoio para gerenciar o risco de segurança da informação para operações organizacionais (incluindo missão, funções e reputação)".

O NIST SP800-115 [Scarfone et al. 2008] inicia definindo a avaliação da segurança da informação como o processo de determinar a eficácia com que uma entidade sob avaliação (por exemplo, *host*, sistema, rede, procedimento, pessoa - o objeto de avaliação) atende aos objetivos e requisitos específicos de segurança e se constitui um guia para os aspectos técnicos básicos da realização de avaliações de segurança da informação, onde são apresentados os aspectos técnicos dos testes, juntamente com os métodos e técnicas de exame e análise que uma organização pode usar como parte do processo de avaliação, oferecendo *insights* aos avaliadores sobre a sua execução e o impacto potencial que eles podem exercer em sistemas e redes.

Além dos aspectos técnicos, antes mesmo dos aspectos metodológicos, demonstra-se uma preocupação com os aspectos gerenciais dos testes, alertando-se que para que uma avaliação seja bem sucedida e tenha um impacto positivo na postura de segurança de um sistema (e, em última instância, de toda a organização), elementos que vão além da Testes e exames devem apoiar o processo técnico.

Os processos e orientações técnicas apresentados visam habilitar as organizações sob teste:

1. no desenvolvimento de políticas de avaliação da segurança da informação, metodologia e papéis e responsabilidades relacionadas com seus aspectos técnicos;

2. no planejamento com precisão de uma avaliação técnica da segurança da informação, provendo orientações sobre que sistemas avaliar e a abordagem de avaliação, abrangendo as questões logísticas e de elaboração um plano de avaliação e ainda levando em consideração as implicações legais e de conformidade;

3. na execução de uma avaliação técnica de segurança da informação de forma segura e eficaz, utilizando métodos e técnicas, capaz de responder a quaisquer incidentes que possam ocorrer durante a avaliação;

4. na manipulação adequada dos dados técnicos (coleta, armazenamento, transmissão e destruição) durante todo o processo de avaliação;

5. e na realização de análises e elaboração de relatórios para traduzir os resultados técnicos em ações de tratamento de risco que venha a melhorar a postura de segurança da organização.

Claramente, há uma preocupação com a inserção das atividades de teste no contexto da análise e avaliação de risco. E nesse contexto, as informações apresentadas destinam-se a serem usadas para uma variedade de propósitos de avaliação. Por exemplo, algumas avaliações

podem se concentrar em verificar se um determinado controle (ou conjunto de controles) de segurança satisfazem os requisitos, enquanto outros pretendem identificar, validar e avaliar um sistema com respeito à possibilidade de exploração de suas fraquezas. As avaliações também podem ser realizadas para aumentar a capacidade de uma organização defender sua rede de computadores. Por fim, ressalva-se que as avaliações não devem substituir a implementação de medidas de controle e manutenção da segurança do sistema.

Mantendo o foco nos aspectos gerenciais, há uma série de recomendações para as organizações, com o objetivo de fazer com que a realização de avaliações técnicas de segurança seja aproveitados ao máximo. Assim, tem-se como objetivos:

1. Estabelecer uma política de avaliação da segurança da informação.
2. Implementar uma metodologia de avaliação repetida e documentada.
3. Determinar os objetivos de cada avaliação de segurança e adequar a abordagem de acordo.
4. Analisar os achados e desenvolver técnicas de mitigação de risco para corrigir as deficiências.

Do ponto de vista específico de metodologia, o referido documento em sua Seção 2.1 *Information Security Assessment Methodology*, enumera os benefícios de uma metodologia de avaliação de segurança repetível e documentada, a saber: prover consistência e estrutura aos testes de segurança visando minimizar os riscos que estes podem gerar; agilizar a capacitação de novas equipes de teste; e apontar limitações de recursos associados com as avaliações de segurança. A metodologia em si é apresentada sucintamente na sequência.

Recomenda-se uma metodologia de avaliação da segurança da informação em fases pois sua estrutura é fácil de seguir, e fornece pontos de ruptura naturais para a transição do pessoal, devendo conter no mínimo as seguintes fases:

- **Planejamento**
  Sendo crítica para uma avaliação de segurança bem-sucedida, na fase de planejamento são coletadas as informações necessárias para a execução da avaliação, tais como: os ativos a avaliar, as ameaças de interesse sobre os ativos e os controles de segurança a serem usados para mitigar essas ameaças. Uma avaliação de segurança deve ser tratada como qualquer outro projeto, com um plano de gerenciamento de projeto para abordar metas e objetivos, escopo, requisitos, funções de equipe e responsabilidades, limitações, fatores de sucesso, suposições, recursos, cronograma e entregáveis.

- **Execução**
  As metas desta fase são identificar vulnerabilidades e validá-las se for o caso. Esta fase deverá abordar as atividades associadas aos métodos e técnicas de avaliação. Embora as atividades específicas para esta fase dependam do que se avalia e como, após

sua conclusão os avaliadores terão identificado o sistema, rede, e vulnerabilidades do processo organizacional.

- **Pós-Execução**
  A fase de pós-execução centra-se na análise de vulnerabilidades identificadas e determinação das causas raiz, estabelecendo recomendações de tratamento e elaborando um relatório final.

Estas fases são discutidas, e em seção específica, é apresentada a metodologia para testes de penetração, cujos passos são ilustrados na figura a seguir (Figura 2.1).

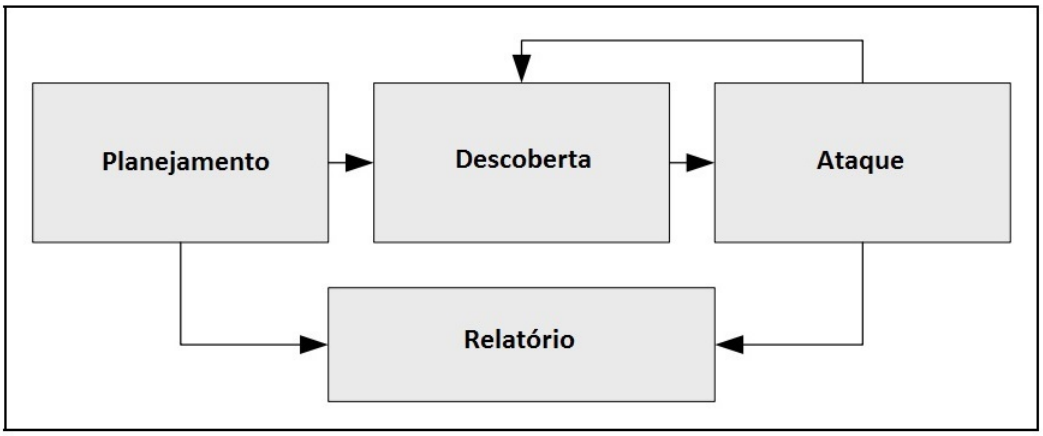

Figura 2.1: Metodologia de Pentest NIST (adaptado de [Scarfone et al. 2008])

Na fase de planejamento, as regras são identificadas, e seguindo a aprovação da gerência, as metas de teste são definidas. Nenhum teste real ocorre nessa fase.

A fase de descoberta do teste de penetração inclui duas partes. A primeira parte é o início de testes reais, abrangendo a coleta de informações e realização de varreduras. A segunda parte da fase de descoberta é a análise de vulnerabilidade, que envolve a comparação dos resultados da coleta de informações sobre os serviços, aplicativos e sistemas operacionais de *hosts* contra bancos de dados de vulnerabilidades (um processo automático para scanners de vulnerabilidade) e o conhecimento dos testadores sobre vulnerabilidades.

A fase de execução de ataques é o cerne de qualquer teste de penetração. A Figura 2.2 representa os passos individuais da fase de ataque, aqui entendido como o processo de verificação de vulnerabilidades potenciais previamente identificadas e a tentativa de sua exploração. Se um ataque for bem sucedido, a vulnerabilidade é verificada e as salvaguardas específicas são identificadas para mitigar a exposição de segurança a ela associada. Em muitos casos, os *exploits* que são executados não concedem o nível máximo de acesso a um invasor. Quando isso ocorre, são necessárias análises e testes adicionais para determinar o verdadeiro nível de risco para a rede, como identificar os tipos de informações que podem ser

colhidas, alteradas ou removidas do sistema. Caso um ataque a uma vulnerabilidade específica seja impossível, o testador deve tentar explorar outra vulnerabilidade descoberta. Se os testadores conseguirem explorar uma vulnerabilidade, eles poderão instalar mais ferramentas no sistema ou rede de destino para facilitar o processo de teste. Essas ferramentas são usadas para obter acesso a sistemas ou recursos adicionais na rede e obter acesso a informações sobre a rede ou organização. Testes e análises em múltiplos sistemas devem ser realizados durante um teste de penetração para determinar o nível de acesso que um adversário poderia ganhar. Este processo é representado no *loop* de realimentação na Figura 2.1 entre a fase de ataque e descoberta de um teste de penetração. A fase de relatório ocorre simultaneamente com as outras três fases do teste de penetração.

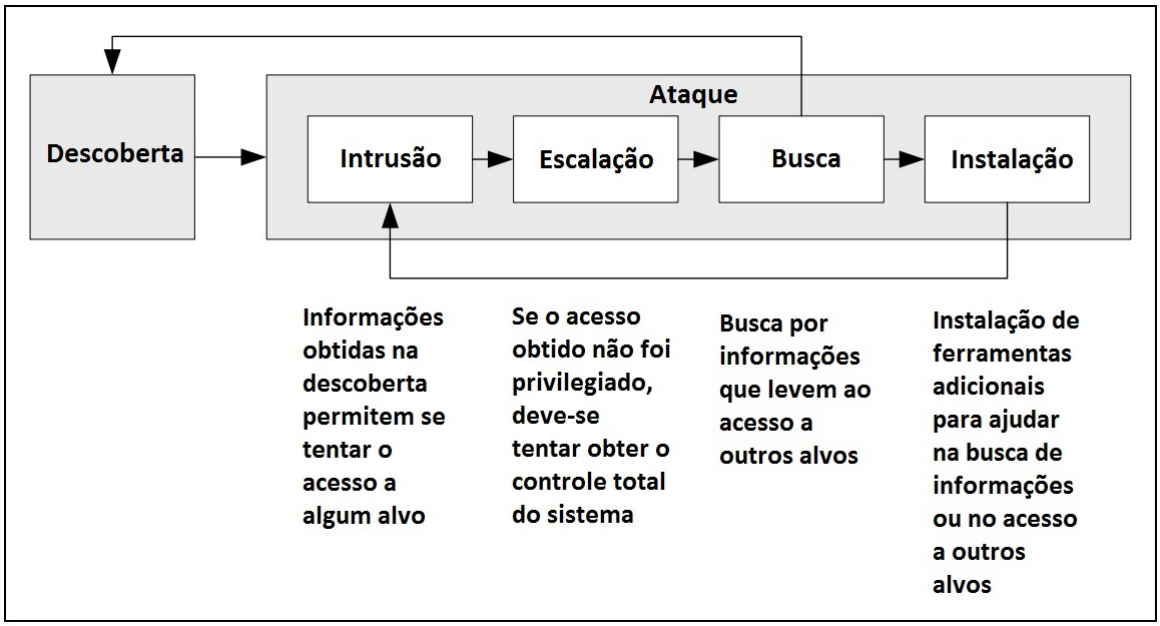

Figura 2.2: Passos do Ataque e Ciclo Realimentação - Metodologia NIST (adaptado de [Scarfone et al. 2008])

Além das técnicas de testes de penetração, são sugeridos ataques de força bruta para quebra de senhas e engenharia social.

2.2.2 *Open Source Security Test Methodology Manual*

A *Open Source Security Test Methodology Manual* (OSSTMM) [Herzog 2015] é quase um padrão de fato em termos de metodologia de testes de penetração: é modular, com processos e atividades bem definidos, sendo bem completa. Entretanto, é muito complexa tentando abranger todos os aspectos de segurança da informação. Talvez seu aspecto mais relevante seja que, apesar de sua ênfase em processos e procedimentos, o seu foco se mantém nos aspectos operacionais. Nas suas primeiras versões, incluía um rol de ferramentas recomendadas.

A OSSTMM, foi criada inicialmente com a finalidade de recomendar normas, metodologias e ferramentas para a comunidade de segurança. Com o tempo, OSSTMM evoluiu para se tornar um padrão de fato para testes de segurança, que fornece subsídios para fundamentar metodologias de testes de segurança, voltada aos profissionais da área. Os princípios que norteiam a OSSTMM são:

- uso de métricas quantificáveis;
- valorização da competência do executor;
- consistência e repetibilidade;
- completude e abrangência;
- persistência de resultados;
- conformidade com as normas de diversos países.

No seu repertório de técnicas utilizadas estão, em ordem crescente de custo e complexidade: varreduras de vulnerabilidade e segurança; testes de penetração e *hacking* ético; avaliação de risco; e auditoria e testes de segurança.

Na sua versão 3, a OSSTMM tem por objetivo determinar o nível da Segurança Operacional (*OpSec*) referente ao escopo de um alvo. *OpSec* é definida como a combinação de "separação e controles sem limitações". Trata-se essencialmente de uma forma de quantificar a proteção entre ativos, utilizando uma fórmula com uma abordagem para identificar e categorizar os controles (medidas de segurança) e limitações (fraquezas ou vulnerabilidades). O resultado final que se deseja obter como caracterização do nível de *OpSec* é a medida, ou estimação, da "Superfície de Ataque"de um determinado alvo, com o objetivo de identificar deficiências nas medidas de proteção implementadas. Por superfície de ataque, entende-se o balanço entre as vulnerabilidades e fragilidades encontradas e os controles efetivamente empregados. Apesar de ser um instrumento usado em, e também em grande parte guiado por, uma metodologia de análise de risco, OSSTMM se apresenta como produzindo subsídios visando auxiliar na tomada de decisões de risco, provendo uma abordagem sistemática para coletar e analisar dados para gerar resultados suficientes. No OSSTMM, risco é considerado um conceito subjetivo, assim este seria um insumo para definir e medir consistentemente o estado de segurança operacional para que as decisões sobre risco possam ser feitas com base em dados quantificados (medidos sistemática e coerentemente), em vez de experiências passadas, preferência de produtos ou outros insumos humanos que podem ser imprecisos. OSSTMM também não se apresenta como metodologia de "Análise de Ameaças", focando apenas na Superfície de Ataque, e tentando identificar e medir deficiências (limitações) na proteção de ativos. O processo definido é repetível e pode ser usado como ferramenta gerencial para avaliar a evolução operações de segurança. OSSTMM se baseia em quatro conceitos chave:

1. Regras de Engajamento (*Rules of Engagement* - RoE):
   As Regras de Engajamento abrangem cerca de 50 pontos individuais que vão desde a abordagem de Vendas e Marketing, até a entrega final do relatório, que definem os parâmetros para a abordagem global de aplicação da metodologia em um contexto específico. Os tópicos elencados estão alinhados com outro conceito básico, o *Critical Security Thinking*, pensamento crítico de segurança e visam obter uma abordagem imparcial para a medição de *OpSec*. Muitas das regras são muito específicas para a notificação, permissão, contratos e realização da avaliação real, indo além dos aspectos puramente técnicos, como em outras metodologias.

2. Pensamento Crítico de Segurança (*Critical Security Thinking*):
   Este conceito foi explicitado na versão 3, apesar de já presente em versões anteriores, e consiste na prática de usar a lógica e fatos, ao invés da opinião, experiência ou parcialidade, para formar ideias sobre segurança. De acordo com o OSSTMM "o processo de pensamento crítico de segurança depende do analista ser capaz de discernir as declarações verdadeiras ou pelo menos reconhecer o grau de possível falsidade ou propriedades dinâmicas em uma declaração. Uma maneira de fazer isso é reconhecer a quantidade de confiança que se pode ter em um fato através do uso de métricas de confiança". O conceito é suportado por uma técnica de 6 passos que auxilia no processo e garante uma abordagem consistente para o pensamento crítico de segurança.

3. Análise de Confiança (*Trust Analysis*):
   Este conceito também foi incluído na versão 3. De acordo com o OSSTMM, a confiança é uma parte da porosidade (*porosity*) de um alvo. Entendendo-se a segurança é como uma barreira que separa as ameaças dos ativos, a confiança é um furo nessa barreira. É onde o alvo aceita a interação oriunda de outros alvos. No entanto, as pessoas tendem a usar controles operacionais impróprios ou incompletos com seus parceiros confiados e confiáveis (*trusts*), como autenticação que foi feita com identificação imprópria, como uma voz sobre um telefone, um cartão de visita, ou mesmo apenas a suposição de que porque uma pessoa está em uma sala que eles estão autorizados a estar lá. Isto tornaria as pessoas susceptíveis a serem enganadas. O uso de controles adicionais são necessários para garantir a integridade e resiliência da confiança. Embora o OSSTMM entre em grande detalhe descrevendo as interações entre os ativos e sua relação com a confiança com ou sem a implementação de certos controles (como autenticação), a análise de confiança acaba por se resumir a uma fórmula usando um conjunto de dez propriedades de confiança, que podem ser aplicadas a quase todas as situações para criar regras de confiança. Quanto ao modo como a análise de confiança se aplica diretamente ao processo de teste de segurança, o OSSTMM prossegue dizendo: "Os testes de segurança verificarão quais são os trusts operacionais, entretanto o uso de regras de confiança é necessário para saber se eles devem existir. Isto é determinado pelo uso das regras de confiança durante os testes de segurança. "

4. Defesa em Largura:

Este conceito vem complementar o comumente aplicado em segurança, a defesa em profundidade. O conceito de defesa em largura envolve a aplicação de vários controles (dez para ser exato) sobre cada vetor ou interação, ao invés de ver uma empresa como sendo protegida por camadas únicas que podem ser "descascadas". O objetivo é avaliar cada ativo (porta, endereço IP, aplicativo, qualquer que seja dada a definição do escopo) contra os dez controles definidos no OSSTMM e medir a deficiência (*OpSec*).

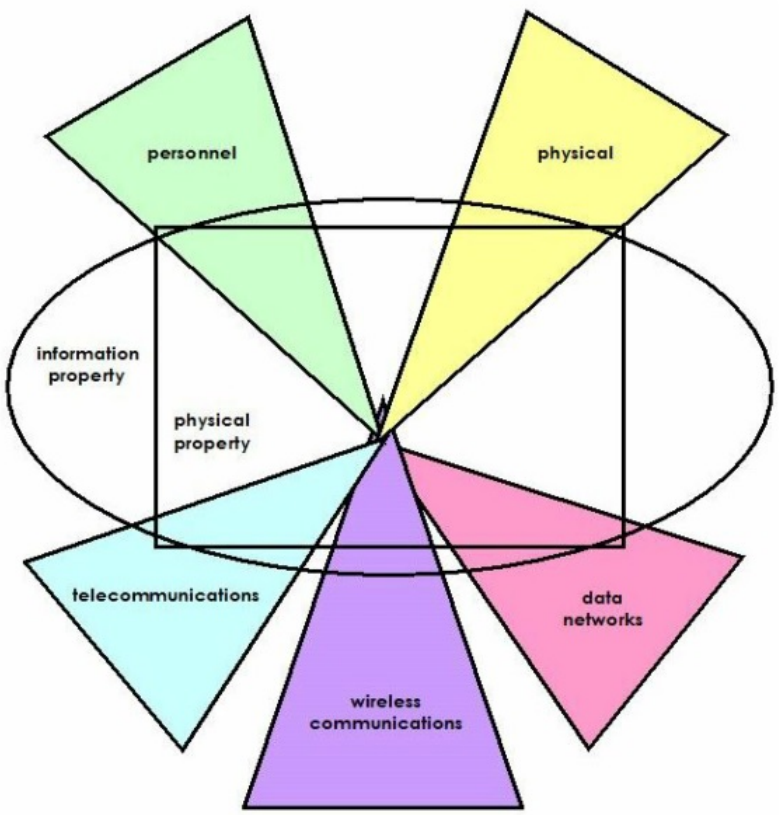

Figura 2.3: Canais da OSSTMM [Herzog 2015]

OSSTMM divide seu escopo de atuação em vários canais, conforme Figura 2.3, que provêm os meios de acesso aos recursos físicos e informacionais. Os canais englobam diferentes aspectos da segurança física:

- canal físico:
  abrangendo segurança física (acesso físico) e humana;
- espectro:
  que diz respeito às comunicações electrônicas, sinais e emanações sobre o espectro eletro magnético conhecido;
- comunicações:
  compreendendo redes de dados, sistemas eletrônicos e redes onde a interação se dá so-

bre meio cabeado, e telecomunicações, se referindo à infraestrutura de comunicações, possivelmente pública, que suporta as redes de dados e as comunicações em geral.

Em cada canal será executada a metodologia, onde seus módulos e tarefas são aplicadas de forma seletiva, seguindo as características do canal específico.

O processo da metodologia é descrito na Figura 2.4 abaixo.

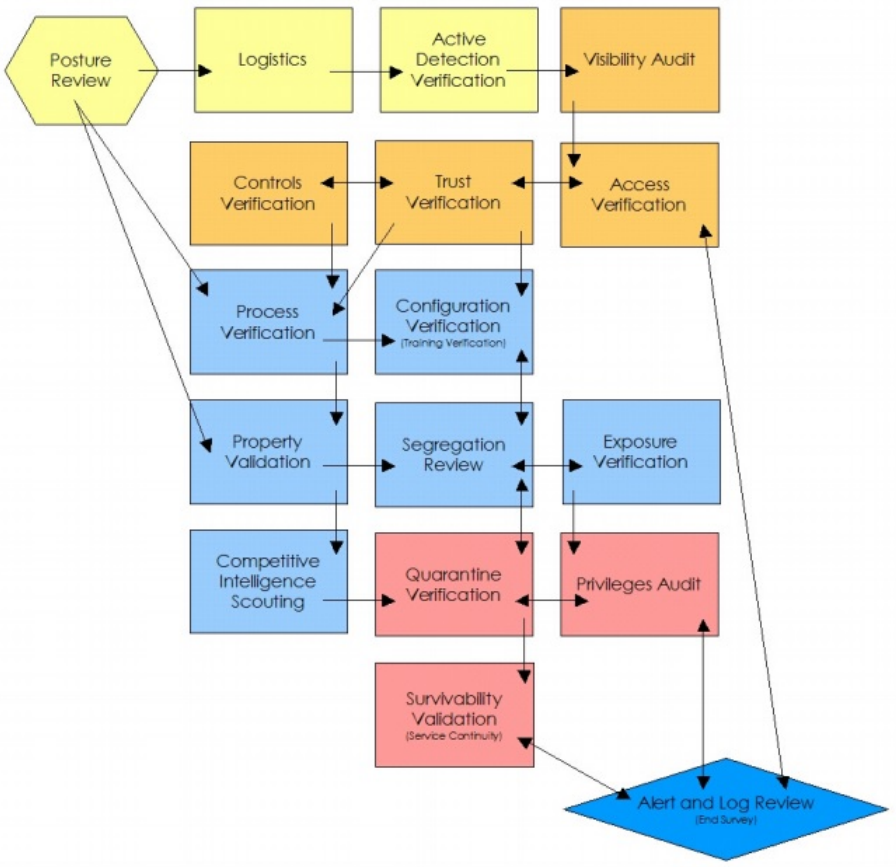

Figura 2.4: Processo da OSSTMM [Herzog 2015]

A OSSTMM se propõe ser uma metodologia de auditoria de segurança, assim o seu processo começa com uma revisão da postura do alvo. A postura é a cultura, regras, normas, regulamentação, legislação e políticas que definem o alvo. O processo termina com comparações de resultados para qualquer alarme, alertas, relatórios ou *logs* de acesso. Para o auditor, define-se o que se deve fazer, sua execução e verificação do resultado. Na metodologia, a separação entre a definição da tarefa e sua execução se dá pela relação entre os canais, que definem os módulos do processo e estes, por sua vez, as tarefas a serem executadas.

Cada módulo tem uma entrada e uma saída. A entrada é a informação usada na execução de cada tarefa, enquanto a saída é o resultado de tarefas concluídas. Essa saída pode ser tanto dados brutos como analisados e pode servir como entrada para outros módulos. Portanto, a falha em completar certos módulos ou tarefas pode limitar o sucesso na conclusão de outros módulos ou tarefas, restringindo o resultado final de uma auditoria.

Com relação aos dezessete módulos do processo, eles cobrem quatro fases na execução desta metodologia:

A. Fase Regulatória

B. Fase de Definições

C. Fase de Informação

D. Fase de Teste de Controles Interativos

Na Fase Regulatória, a auditoria se inicia com a compreensão dos requisitos, escopo e seus limites. Seus módulos estão descritos abaixo:

- Revisão de Postura:
  Consiste na revisão da cultura, regras, normas, marcos regulatórios e políticas aplicáveis ao alvo. Tem por objetivo definir escopo e testes a serem realizados.

- Logística:
  É a avaliação do impacto de distâncias, velocidade e falhas na precisão dos resultados. Seu objetivo é avaliar limitações da auditoria para melhoria de sua eficiência.

- Verificação de Impactos de Detecção:
  Verificação dos limitadores de interação, resposta e previsibilidade dos testes. O objetivo é definir restrições impostas aos testes interativos

Na Fase de Definições, se dá a definição e o detalhamento do escopo. Seus módulos são:

- Visibilidade da Auditoria:
  Neste módulo se determina dentro do escopo os alvos aos quais se aplicarão os testes, com visibilidade sendo entendida como presença. O objetivo é conhecer os alvos e como estes se relacionam no escopo.

- Verificação de Acesso:
  Neste módulo se avaliam a abrangência e a profundidade da interação de pontos de acesso do alvo e a autenticação necessária. Como o ponto de acesso é o ponto principal na interação com um ativo, a determinação de sua existência e forma de atuação são cruciais na avaliação do alvo.

- Verificação de Confiança:
  É a determinação das relações de confiança entre os alvos. O conhecimento das relações de confiança entre os alvos evidenciam a importância da interação entre eles e suas dependências.

- Verificação de Controles:
  Neste módulo se avaliam o uso e a eficácia de controles de perdas baseados no processo, tais como: não repúdio, confidencialidade, privacidade e integridade. Os processos são normalmente definidos em resposta a uma interação, persistindo mesmo ao seu fim ou mudança de estado, sendo necessário mapear os controles a eles associados.

A Fase de Informação tem por finalidade levantar informações, priorizando a busca por inconsistências. Seus módulos são:

- Verificação de Processo:
  Este módulo determina a existência e a efetividade do registro e manutenção dos níveis de segurança e também da diligência, determinada na revisão de posturas e controles. A ideia é se estabelecer não só a presença dos elementos do processo de segurança, mas também como estes estão operando.

- Verificação de Configuração:
  Aqui se avalia o estado normal de operação, segundo as condições normais, para se identificar problemas sem a aplicação dos testes. A compreensão das condições padrão sob as quais os alvos operam, não só auxiliam a justificar os alvos, mas também as condições de teste.

- Validação de Propriedades:
  Neste módulo se avaliam a abrangência e a profundidade do uso ilegal ou sem licença de propriedade intelectual ou aplicações nos alvos. O objetivo é estabelecer o estado da aplicação de direitos de propriedade.

- Revisão de Segregação:
  O módulo determina o nível de informação pessoal identificável definido pela revisão de postura. Assim, se conhecem os direitos de privacidade e a extensão das informações classificáveis não cobertas por tais requisitos.

- Verificação de Exposição:
  Neste módulo ocorre a busca por informação livremente acessível que descreve a visibilidade indireta de alvos ou ativos dentro do canal. A ideia é avaliar a exposição das informações sobre alvos e ativos em fontes abertas.

- Verificação de Inteligência Competitiva:
  Aqui se busca por informação livremente acessível que direta ou indiretamente pode vir a prejudicar o proprietário de alvos ou ativos dentro do canal. A ideia é avaliar a exposição das informações que isoladas ou agregadas podem influenciar decisões de competidores.

E por fim, a Fase de Teste de Controles Interativos, onde os testes são voltados à penetração e disrupção. Esta fase ocorre ao final dos testes para minimizar possíveis disrupções. Esta fase é composta pelos módulos descritos abaixo:

- Verificação de Quarentena:
  Aqui se determina e avalia o uso efetivo de quarentena para todos os acessos tanto para quanto do alvo. Assim, se determina a efetividade dos controles de autenticação e contenção.

- Auditoria de Privilégios:
  É o mapeamento e a avaliação do impacto do mau uso dos controles, credenciais ou privilégios na contenção e da escalação de privilégios, se detectada. A ideia é determinar a eficácia dos privilégios de autorização nos controles de autenticação e contenção.

- Validação de Sobrevivência:
  O módulo determina e avalia a resistência do alvo a estímulos adversos, onde controles de continuidade e resiliência podem ser afetados. Assim, determina-se a efetividade dos controles de continuidade e resiliência verificando a ocorrência de negação de serviço.

- Revisão de Alertas e *Logs*:
  Aqui ocorre a revisão das atividades de auditoria realizadas com sua verdadeira profundidade, conforme registrado pelo alvo. Desta forma, se consegue determinar que partes da auditoria deixaram uma trilha usável e confiável.

Cada um destes módulo é também detalhado em termos das tarefas a serem executadas. Como exemplo, temos as seguintes tarefas referente à varredura de rede do módulo de Auditoria de Visibilidade:

11.4.2 Enumeração

...

(l) Verificar as respostas dos fragmentos TCP SYN para as portas 0, 21, 22, 23, 25, 53, 80 e 443.

(m) Verificar as respostas de todas as combinações de flags TCP para as portas 0, 21, 22, 23, 25, 53, 80 e 443.

...

A OSSTMM tem como objetivo final calcular o *Risk Assessment Value* (RAV) como forma de se obter um valor quantitativo e não subjetivo para avaliar a superfície de ataque oferecida pela organização sob teste. O RAV consiste em um balanço entre os controles que deveriam estar implementados, os implementados, sua eficácia/eficiência e as vulnerabilidades encontradas. Para o cálculo, controles e vulnerabilidades são classificados para a devida ponderação, segundo os critérios da metodologia. A ideia é que com a aplicação periódica da metodologia, e o subsequente cálculo do RAV, possa se acompanhar e avaliar a efetividade de medidas de segurança implementadas na organização.

2.2.3 *Penetration Testing Execution Standard*

O *Penetration Testing Execution Standard* (PTES) [PTES 2009], é um padrão projetado para fornecer tanto empresas e prestadores de serviços de segurança uma linguagem comum e um escopo para realização de testes de penetração, aí entendidos como avaliações de segurança). Começou no início de 2009 na sequência de uma discussão que desencadeou entre alguns dos membros fundadores sobre o valor (ou falta de) de testes de penetração na indústria, sendo elaborado por um grupo de Profissionais de segurança de todas as áreas da indústria (isto é, instituições financeiras, provedores de serviços, fornecedores de segurança).

Apesar de não prover diretrizes técnicas sobre a execução dos testes em si, vem com um guia técnico que consiste em um catálogo de ferramentas e técnicas para a execução das fases de seu processo. Apesar da proposta ter um viés metodológico, ela finda por se caracterizar por seus aspectos operacionais e de habilidades no uso das ferramentas que constam no guia. Desta forma, apesar da separação em fases, o aspecto metodológico se enfraquece ante o apelo operacional das ferramentas.

PTES não inclui todos os possíveis cenários de *pentest*, mas destina-se a definir uma base para o mínimo necessário para um *pentest*, assim como vários níveis que fornecem atividades mais abrangentes necessárias para organizações com maiores necessidades de segurança. Os diferentes níveis também seriam definidos de acordo com a indústria em que serão aplicados os testes.

O público-alvo inclui duas comunidades principais: empresas que demandam o serviço e prestadores de serviços. Para as empresas o objetivo é permitir que eles exigem uma linha de base específica de trabalho como parte de um *pentest*. Para os prestadores de serviços, o objetivo é fornecer uma linha de base para os tipos de atividades necessárias, o que deve ser levado em conta como parte da abrangência do escopo através de relatórios e entregas.

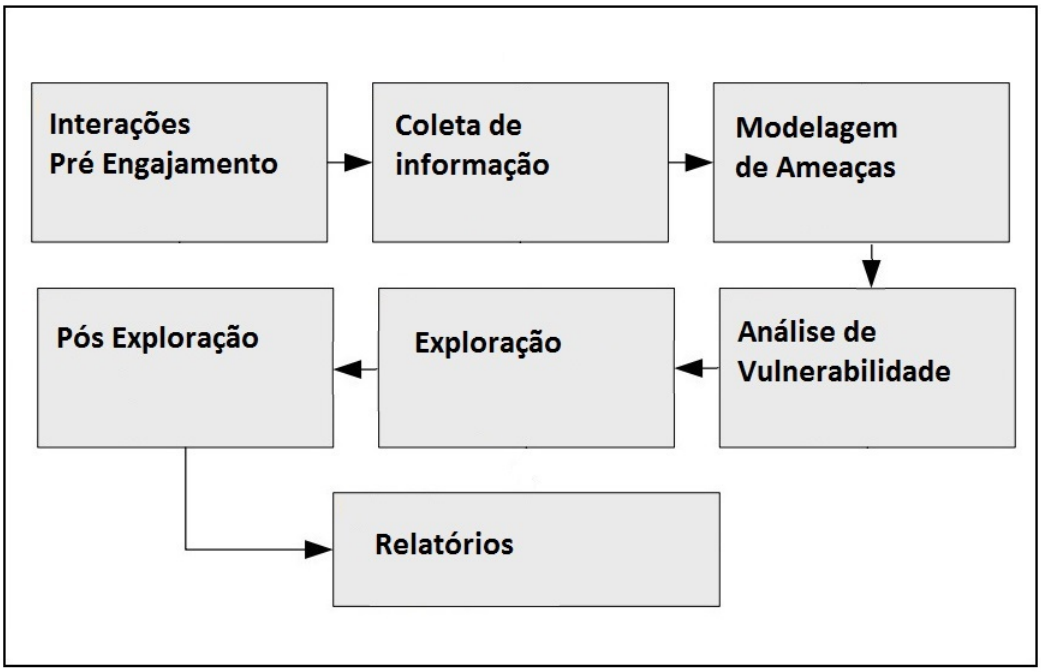

Figura 2.5: Metodologia PTES (adaptado de [PTES 2009]

PTES consiste em sete seções principais. Estes abrangem tudo relacionado a um teste de penetração, desde a comunicação inicial e raciocínio por trás de um *pentest*, através das informações coletadas e modelagem de ameaça, a fim de obter uma melhor compreensão da organização testada; exploração e pós-exploração, onde o conhecimento técnico de segurança dos testadores vem se agregar com a compreensão do negócio do envolvimento e, finalmente, para o relatório, que captura todo o processo, de forma que faça sentido para o cliente e lhe seja mais útil.

A norma prescreve um processo que serve de base para a execução do testes de penetração, conforme a Figura 2.5:

1. Pré-engagemento

   Nesta fase se define o escopo, bem como os objetivos dos testes, e se estima o tempo que será gasto executando-os. Da estimativa de tempo se deriva o valor a se remunerar pelos testes. Para auxiliar neste processo de definição, são apresentados questionários. Estes enfatizam aspectos quantitativos dos ativos envolvidos nos testes.

2. Coleta de Informações

   Esta fase utiliza fortemente técnicas de pesquisa em fontes abertas e tem por objetivo subsidiar a seleção de alvos. Assim, a Coleta de Informações está dividida em três categorias, descritas a seguir:

- Nível 1
  Neste nível, a informação pode ser obtida quase inteiramente por ferramentas automatizadas.
- Nível 2
  Este nível combina o uso de ferramentas automatizadas como no nível 1 e análises manuais. O objetivo é conseguir uma boa compreensão do negócio, incluindo informações como localização física, relações comerciais, organogramas, etc.
- Nível 3
  Aqui se considera toda a informação obtida nos níveis 1 e 2, juntamente com intensa análise manual. A ideia é se obter uma compreensão profunda de relações comerciais, muito provavelmente um grande número de horas para realizar a coleta e correlação.

Para a seleção de alvos, combinam-se as informações coletas com os resultados de varreduras de redes e aplicações.

3. Análise de Ameaça

Esta fase define uma abordagem de modelagem de ameaças para a correta execução de um teste de penetração. Apesar de não utilizar um modelo específico, exige que o modelo utilizado seja consistente em termos de representação de ameaças, suas capacidades e qualificações conforme a organização que está sendo testada e a capacidade de ser repetidamente aplicada a testes futuros com mesmos resultados.

O padrão centra-se em dois elementos-chave da modelagem de ameaças tradicionais - ativos e atacante (agente ou comunidade de ameaça). Cada um é, respectivamente, dividido em ativos de negócios e processos de negócios e as comunidades na ameaça e suas capacidades.

No mínimo, os quatro elementos devem ser claramente identificados e documentados em cada teste de penetração. Ao modelar o lado do atacante, sobre a comunidade de ameaça e seus recursos, aspectos adicionais de modelagem de motivação também devem ser fornecidos.

A fase de modelagem de ameaças de qualquer processo de testes de penetração é crítica para os testadores, bem como para a organização, fornecendo clareza no que diz respeito ao apetite e priorização do risco da organização (quais ativos são mais importantes que outros, quais comunidades de ameaças são mais relevantes que outras?). Além disso, permite que o testador se concentre em entregar um engajamento que imite as ferramentas, técnicas, capacidades, acessibilidade e perfil geral do atacante, mantendo em mente quais são os alvos reais dentro da organização de tal forma que os controles, processos mais relevantes.

O processo de modelagem de ameaças em si, visto de alto nível, envolve:

(a) Reunir documentação relevante.

(b) Identificar e categorizar ativos primários e secundários.

(c) Identificar e categorizar ameaças e comunidades de ameaças.

(d) Mapear comunidades de ameaças contra ativos primários e secundários.

4. Análise de Vulnerabilidade

   O teste de vulnerabilidade é o processo de descobrir falhas em sistemas e aplicações que podem ser alavancadas por um invasor. Essas falhas podem variar em qualquer *host* ou serviço de configuração incorreta, ou ainda no design de aplicativo inseguro. Embora o processo usado para procurar falhas varie e seja altamente dependente do componente particular que está sendo testado, alguns princípios chaves aplicam-se ao processo.

   Ao realizar análises de vulnerabilidades de qualquer tipo, o testador deve avaliar adequadamente a profundidade e amplitude aplicáveis para atender às metas e requisitos do resultado desejado.

5. Exploração

   A fase de exploração de um teste de penetração centra-se unicamente no estabelecimento do acesso a um sistema ou recurso, a despeito das restrições de segurança. Se na fase anterior a análise de vulnerabilidade foi realizada corretamente, esta fase poderá ser planejada com precisão. O foco principal é identificar o ponto de entrada principal na organização e identificar ativos de valor alto.

   Se a fase de análise de vulnerabilidade foi devidamente concluída, alvos de alto valor devem ter sido comprometidos. Em última análise, o vetor de ataque deve levar em consideração a probabilidade de sucesso e o maior impacto na organização.

6. Posexploração

   O objetivo da fase pós-exploração é determinar o valor dos ativos comprometidos e manter o controle destes para uso posterior. O valor do ativo é determinado pela sensibilidade dos dados nele armazenados e pelo potencial das máquinas em comprometer ainda mais a rede. Nesta fase são apresentados métodos e técnicas, e sugeridas ferramentas que se destinam a ajudar o testador a identificar e documentar dados sensíveis, configurações, canais de comunicação e relações com outros dispositivos de rede que podem ser utilizados para obter mais acesso à rede e configurar um ou mais métodos acessando a máquina posteriormente. Nos casos em que estes métodos diferem das regras de compromisso acordadas, as regras de envolvimento devem ser seguidas. Ao final desta fase, caso o efeito de algum teste persista, deve-se realizar a restauração do estado original.

7. Relatórios

   Este documento destina-se a definir os critérios de base para o relatório de testes de penetração. O formato proposto prescreve duas partes no relatório: um sumário executivo e um relatório técnico.

Como o padrão não fornece diretrizes técnicas quanto à forma de executar um *pentest* real, há um guia técnico para acompanhar o próprio padrão. Neste guia técnico, *Technical Guidelines*, são descritas ferramentas e procedimentos.

2.2.4 *Payment Card Industry Security Standards Council Data Secutity Standard*

O *Payment Card Industry Security Standards Council* (PCI-SSC) também propõe uma metodologia estruturada, a DSS (*Data Secutity Standard*), baseada em processo e voltada a aplicações de comércio e segurança de transações com *Personal Identification Number* (PIN).

A metodologia definida pelo PCI-SSC é a *PCI Data Secutity Standard* (PCI-DSS). A definição da PCI-DSS está em *Information Supplement: Penetration Testing Guidance* [PCI 2015], e estabelece dirtrizes gerais para testes de penetração focadas em:

- Componentes de Teste de Penetração:
  a compreensão dos diferentes componentes que compõem um teste de penetração e como isso difere de uma varredura de vulnerabilidade, incluindo escopo, aplicativo e testes orientados a camadas de redes, verificações de segmentação e engenharia social.
- Qualificações de um testador de penetração:
  definindo as qualificações de um executor de teste de penetração, interno ou externo, através de sua experiência passada e certificações.
- Metodologias de Teste de Penetração:
  informações detalhadas relacionadas às três partes de um teste de penetração: pré-engajamento, engajamento e pós-engajamento.
- Diretrizes de Revisão de Testes de Penetração:
  com orientações para desenvolver um relatório de teste de penetração abrangente que inclua as informações necessárias para documentá-lo.

Inicialmente, definem-se os objetivos do teste de penetração, que seriam: determinar se e como um usuário mal-intencionado pode obter acesso não autorizado a ativos que afetam a segurança fundamental do sistema, arquivos, *logs* ou dados do titular do cartão; e confirmar

que os controles aplicáveis, como o escopo, gerenciamento de vulnerabilidades, metodologia e segmentação, exigido no PCI DSS estão corretamente empregados. Na sequência, faz-se uma diferenciação entre teste de penetração e varredura de vulnerabilidade, deixando clara a profundidade e completude do primeiro. Seguem definições acerca dos requisitos dos testes e qualificações dos executores, chegando finalmente à metodologia.

A Metodologia tem três fases: pré teste, teste e pós teste (Figura 2.6). No pré teste, antes mesmo do início do teste, recomenda-se que todas as partes envolvidas (organização, testador e, quando aplicável, o avaliador) sejam informados dos tipos de testes (isto é, internos, externos, camada de aplicação ou camada de rede) a ser executado, como serão realizados e quais serão os alvos. Ao coordenar esses detalhes em primeiro lugar, questões em que o escopo é definido incorretamente ou outros problemas que exigiriam um reteste podem vir ser evitados. Há uma série de recomendações quando à definição do escopo, documentação a ser levantada, regras de engajamento, e critérios de sucesso para os testes.

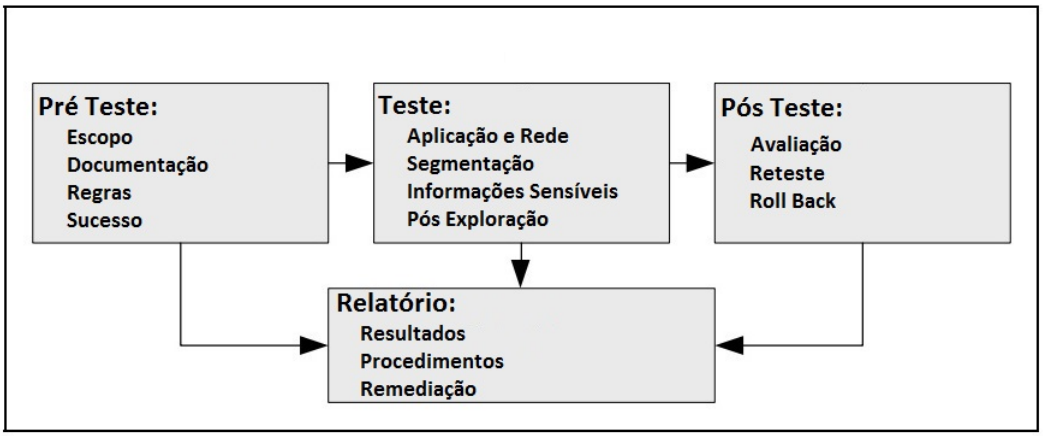

Figura 2.6: Metodologia PCI-DSS (adaptado de [PCI 2015])

O teste de penetração em si é visto essencialmente como um esforço manual, recomendando a adoção de ferramentas para auxiliar a sua realização. Desta forma, haveria uma redução do esforço com relação a tarefas repetitivas. Porém, se alerta que é necessário um julgamento na escolha de ferramentas apropriadas e também da existência de vetores de ataque que normalmente não podem ser identificados automaticamente. Entretanto, apesar de não haver indicação explícita de um conjuntos de ferramentas a serem usadas, não são feitas considerações acerca de como se deve proceder na escolha destas.

Ainda na fase de teste, há recomendações sobre testes de aplicação e as credenciais necessárias para sua execução; testes na camada de rede, sendo recomendado o uso de ferramentas automatizadas; testes avaliando a segmentação, que se recomenda executar entre as primeiras atividades; considerações sobre o manejo de informação sensível, especialmente informações de portador de cartão de crédito; e recomendação de persistência e posexploração do ataque.

No posteste, tem-se as diretivas sobre a avaliação das vulnerabilidade e a preparação de um relatório. Para a avaliação, faz-se referência a uma série de normas e padrões da indústria, e também se provê um *template* para apresentação dos resultados, incluindo descrição de procedimentos (lá referenciado como metodologia) e ferramentas utilizadas. Há ainda, a recomendação de melhores práticas de remediação, reteste, posremediação e *roll back* de qualquer operação persistente executada em preparação ou durante o teste. Assim, o posteste tem dois passos bem diferenciados: um de ordem técnica, outro voltado à documentação.

2.2.5 SANS

O SANS Institute também oferece uma metodologia com seu processo descrito em [SANS Inst. 2002] que segue um forte apelo operacional, que apresenta uma metodologia baseada em processo, mas com forte foco em ferramentas.

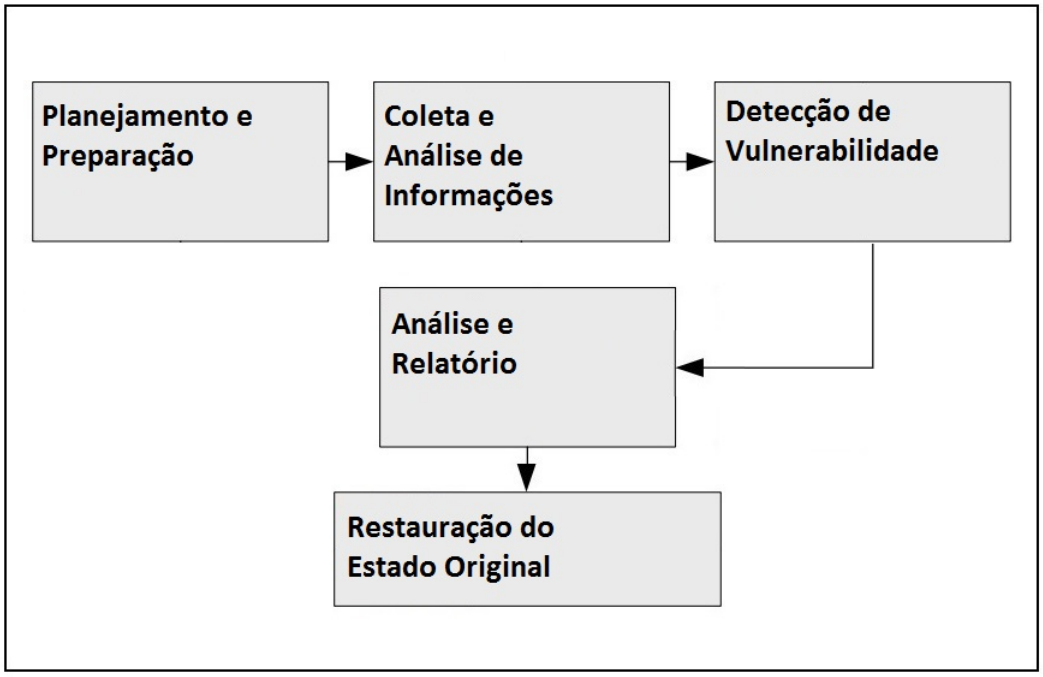

Figura 2.7: Metodologia SANS (adaptado de [SANS Inst. 2002]

O processo é definido em fases, conforme a Figura 2.7, sendo descrito a seguir:

- Planejamento e Preparação
  Neste passo deve-se definir o âmbito e o objetivo do teste de penetração, devendo haver um objetivo claro para o teste de penetração a ser conduzido. De acordo com esta metodologia, o objetivo de um teste de penetração é demonstrar que existem vulnerabilidades exploráveis na infraestrutura de rede de uma organização. O escopo do teste de penetração deve ser definido, identificando máquinas, sistemas e redes, requi-

sitos operacionais e pessoal envolvido. O formato no qual os resultados do teste serão apresentados devem também ser acordados.

- Coleta e Análise de Informações
  Deve-se coletar o máximo de informação possível sobre os sistemas e redes alvo. Há uma grande variedade de ferramentas disponíveis para esta fase.

- Detecção de Vulnerabilidade
  Ao fim da coleta e análise de informações, os executores do teste de penetração devem ter um conjunto de vulnerabilidades à sua disposição para serem testadas. Uma análise será feita sobre as informações obtidas para determinar qualquer possível vulnerabilidade que possa existir por meio de varredura manual de vulnerabilidades.

- Análise e Relatório
  Após a realização de todas as tarefas acima, a próxima tarefa é gerar um relatório para a organização. O relatório deve começar com uma visão geral do teste de penetração e a descrição do processo executado. Isto deve ser seguido por uma análise e comentários sobre vulnerabilidades que existem na rede ou nos sistemas. O relatório deve conter:

  - Resumo de todos os cenários de penetração bem sucedidos.
  - Lista detalhada de todas as informações coletadas durante os testes de penetração.
  - Lista detalhada de todas as vulnerabilidades encontradas.
  - Descrição de todas as vulnerabilidades encontradas.
  - Sugestões e técnicas para resolver vulnerabilidades encontradas.

- Restauração do Estado Original
  Caso tenha se alterado algum aspecto das redes e sistemas sob teste de alguma forma persistente, deve-se restaurar o estado anterior ao teste. O processo de limpeza é feito para remover qualquer situação que tenha sido resultado do teste de penetração.

2.2.6 *Open Source Application Security Project*

A *Open Source Application Security Foundation* propõe e mantém a metodologia *Open Source Application Security Project* (OWASP) [OWASP 2014] que é também um quase padrão de fato, sendo porém específico para o domínio de aplicações web. Entretanto, sua proposta não é ser uma metodologia de testes de penetração, mas integrar-se ao processo de desenvolvimento de aplicações, com ênfase no posdesenvolvimento e preprodução.

A OWASP periodicamente publica listas de vulnerabilidades mais frequentes com códigos de teste para sua identificação, juntamente com explicações sobre as vulnerabilidades e sugestão de formas de correção. Além disso OWASP define testes e provê ferramentas específicas para testes de segurança de aplicações.

Para OWASP, os testes de penetração são mais uma ferramenta no processo de melhoria de segurança de aplicações web, podendo ocorrer durante o desenvolvimento ou com sistemas já em produção. Na verdade, OWASP não define uma metodologia de teste de penetração. O *OWASP Technical Guide v4* recomenda, entre várias possíveis metodologias, as anteriormente descritas.

O OWASP é orientado ao ciclo de vida do desenvolvimento de software (*Software Development Life Cycle* (SDLC) e seu *framework* de teste (Figura 2.8) define atividade em cinco estágios do SDLC:

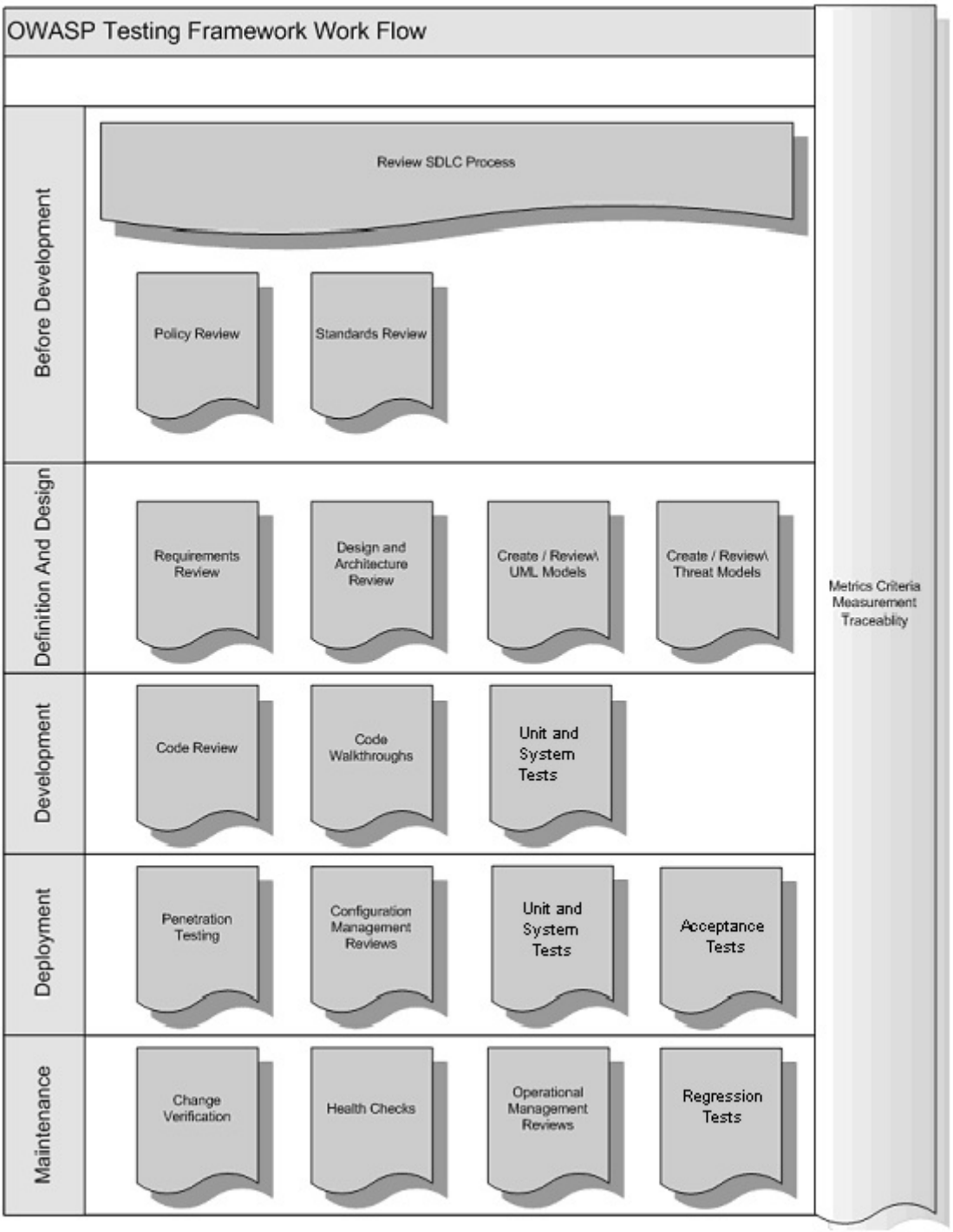

Figura 2.8: *Framework* de Teste OWASP [OWASP 2014]

- antes do início do desenvolvimento:
  neste estágio deve-se definir um SDLC em que a segurança seja inerente a suas fases, juntamente com a revisão de políticas e padrões, e o desenvolvimento de métricas e padrões voltados à rastreabilidade;
- durante a definição e projeto:
  este estágio envolve revisão de requisitos de segurança e do projeto, e da arquitetura, modelos UML e de ameaça;

- durante o desenvolvimento:
  a principal atividade neste estágio é a revisão de código, inicialmente em alto nível e depois em mais detalhe;

- durante a implantação:
  aqui ocorrem efetivamente os testes: de penetração e de gerenciamento de configuração;

- na manutenção e operação:
  neste estágio temos as revisões do gerenciamento operacional, checagens periódicas de segurança e a verificação de realização de mudanças.

2.2.7 Outros exemplos de metodologias

A título de exemplo, há também outras metodologias menos utilizadas, porém relevantes, ilustrando as tendências presentes nas metodologias acima.

Uma outra metodologia também restrita em escopo é a apresentada pelo Infosec Institute [Infosec Institute 2014] que abrange apenas aplicações web. Alisherov e Sattarova [Alisherov e Sattarova 2009] descrevem uma proposta de metodologia preocupada em atender a políticas e conformidade, com viés operacional porém sem especificar procedimentos.

Há ainda várias outras metodologias publicadas, voltadas aos mais variados aspectos, como: treinamento profissional ([DeFino 2010], [Wilhelm 2010], [Anonymous 2011], [Wilhelm e Andress 2011], [Basta 2013], [Engebretson 2013], [Neely, Hamerstone e Sanyk 2013], [Wilhelm 2013], [Baloch 2014], [Weidman 2014]); certificação ([Engebretson e Broad, CISSP. 2011], [Tiller 2012]); tutorial de uso de ferramentas ([Broad e Bindner 2014]); uso de técnicas de engenharia social ([Watson 2014]); dispositivos móveis ([Bergman et al. 2013]); dispositivos de baixa potência ([Polstra e Ramachandran 2015]); e auditoria de rede ([Jackson 2010]).

Apesar de recentes, por serem fortemente orientadas a técnicas e ferramentas, em contextos tecnológicos específicos que evoluem, estão quase todas defasadas com relação ao nível das medidas de proteção e às técnicas e ferramentas mais recentes. A Tabela 2.1 resume as metodologias e características aqui apresentadas.

Tabela 2.1: Quadro Demosntrativo das Metodologias Citadas na Seção 2.2.7

Metodologia	**Foco**	**Abordagem**	
		Procedimento	**F**erramental
[Infosec Institute 2014]	aplicações web	✓	
[Alisherov e Sattarova 2009]	conformidade	✓	
[DeFino 2010]	treinamento	✓	✓
[Wilhelm 2010]	treinamento	✓	✓
[Anonymous 2011]	treinamento	✓	✓
[Wilhelm e Andress 2011]	treinamento	✓	✓
[Basta 2013]	treinamento	✓	✓
[Engebretson 2013]	treinamento	✓	✓
[Neely, Hamerstone e Sanyk 2013]	treinamento	✓	✓
[Wilhelm 2013]	treinamento	✓	✓
[Baloch 2014]	treinamento	✓	✓
[Weidman 2014]	treinamento	✓	✓
[Engebretson e Broad, CISSP. 2011]	certificação	✓	✓
[Tiller 2012]	certificação	✓	✓
[Broad e Bindner 2014]	tutorial	✓	✓
[Watson 2014]	engenharia social	✓	
[Bergman et al. 2013]	disp. móveis	✓	✓
[Polstra e Ramachandran 2015]	disp. baixa potência	✓	✓
[Jackson 2010]	auditoria de rede	✓	

2.3 Ciclos de Decisão

Por falta de termo melhor, a expressão *ciclos de decisão* está sendo aqui usada para se referir a *meta* descrições de ciclos planejamento, decisão, execução e avaliação comumente presentes no contexto da gestão. Como será visto adiante, a abordagem adotada na formulação da metodologia proposta consiste em adotar um ciclo de tomada de decisão genérico o suficiente para acomodar a execução dos ataques. Desta forma, são consideradas duas abordagens: o ciclo PDCA e o ciclo OODA.

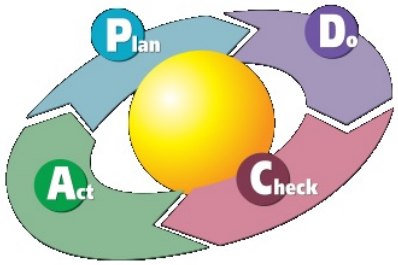

Figura 2.9: Ciclo PDCA

O ciclo PDCA *Plan-Do-Check-Act*, [Shewhart e Deming 1939] [Deming 1986] (Figura 2.9), foi originalmente proposto no contexto da gestão de controle de qualidade em processos industriais. Posteriormente, foi sendo adaptado e aplicado a diversas outras áreas. No

contexto de nosso interesse, o ciclo PDCA se apresentou como opção uma vez que serve de base para todo o *framework* de gestão de segurança da informação. Entretanto, o ciclo PDCA está normalmente associado a processos perenes, de longa duração, com fins de melhoria contínua. O ciclo define claramente as fases de planejamento, execução e avaliação. Assim, o ciclo PDCA tem sido apropriadamente aplicado em situações onde a dinâmica das ações é relativamente lenta, sendo medida em meses ou anos. Dessa forma, ainda que o PDCA se adequasse como base para estruturação das fases da metodologia, não dispunha do ferramental para ciclos rápidos de decisão e execução.

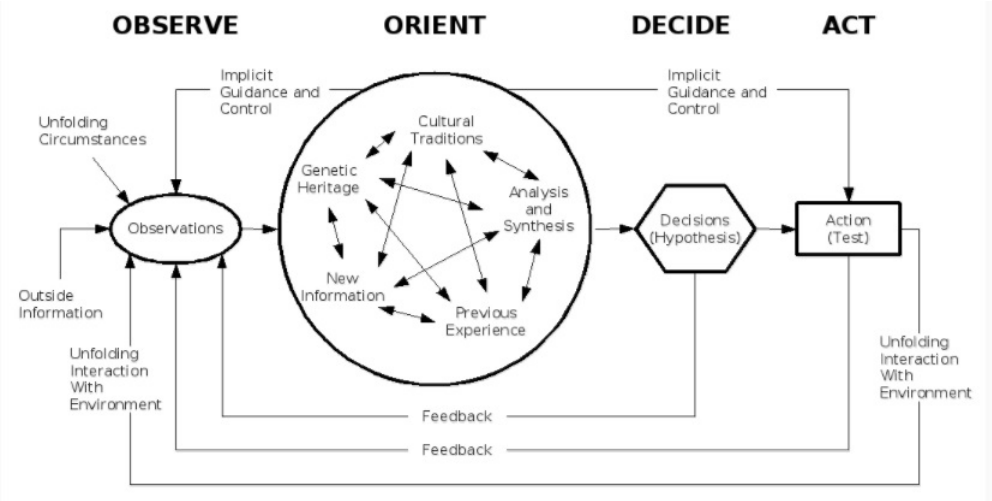

Figura 2.10: Ciclo OODA [Boyd 1976]

Já o ciclo OODA *Observe-Orient-Decide-Act* (Figura 2.10) foi concebido tendo em mente ambiente conflitivo com rápidas interações [Boyd 1976], [Boyd 1995]. Na sua origem, o ciclo OODA se inspirou no combate aéreo entre MIG-15 e F-86 Sabre durante a Guerra da Coréia. A tomada de decisão posterior à observação e a sua imediata avaliação estão mais próximas do que acontece em um ataque cibernético. Mesmo assim, a provisão da fase de orientação, que além da análise dos resultados da observação, inclui ações defensivas e potencialmente agressivas no sentido de manter o atacante em condições de não ser atacado e ao mesmo tempo dispor de opções ofensivas. Na verdade, pode-se incluir nessa fase de orientação a simulação do ciclo OODA que o adversário estaria executando.

Este ciclo se adequa à modelagem de situações conflitivas altamente interativas, provendo a dinâmica necessária para representar os papéis de atacante e atacado ainda que estes venham a exercer funções duais de ataque e defesa, inclusive simultaneamente.

2.4 Ataques de Negação de Serviço

Ataques de negação de serviço (*Denial of Service*, DoS) tem por finalidade exaurir recursos de dispositivos ou de infraestrutura para causar a disrupção de serviços e conse-

quentemente levar a indisponibilidade. Em sua forma volumétrica, o ataque basicamente consiste no envio de grandes quantidades de *requests* que podem se sobrepujar ao servidor que tenta tratá-los ou exceder os limites de tráfego congestionando enlaces de acesso [McDowell 2009]. Uma forma comum de implementar e potencializar ataques DoS é pelo uso de técnicas de distribuição (DoS distribuído, DDoS), em que vários nós enviam tráfego de forma coordenada, resultando em maior intensidade no ataque e complexidade de mitigação, dada a frequente ofuscação da origem.

Além dos ataques volumétricos, um ataque DoS também pode se viabilizar por abuso de protocolo. Este caso diz respeito a ataques de baixo volume, com taxa lenta onde *probes* com tráfego legítimo exploram características e facilidades específicas, ou detalhes de implementação de um protocolo que levam à exaustão de algum recurso do alvo de modo que *requests* legítimos não conseguem ser atendidos apropriadamente. Assim, consegue-se efeito idêntico ao caso volumétrico, onde se excede a capacidade de processamento ou a banda dos enlaces.

Ataques DoS por abuso de protocolo podem ser separados em dois grupos: por exploração de protocolo, quando o atacante usa maliciosamente alguma característica ou funcionalidade; e por exploração de implementação. Como exemplo do primeiro caso, tem-se os ataques *Slowlorris*, em que HTTP GETs são enviados sem completar o endereço do recurso solicitado, e *RUDY*, onde HTTP POSTs com um só caracter são enviados [Damon et al. 2012]; No segundo caso, pode-se citar o clássico e histórico ataque *Ping of Death* [Kenney 1996].

Os ataques volumétricos também podem ser divididos em ataques por *flooding* e por reflexão amplificada. Em ambos os casos, tenta-se sobrecarregar algum recurso do alvo, normalmente banda, pelo envio de grandes volumes de tráfego. Em ataques por *flooding*, nós já comprometidos e controlados pelo atacante enviam tráfego diretamente para o alvo, enquanto em ataques por reflexão nós intermediários, refletores, são usados para o mesmo fim. Para o atacante, o refletor é qualquer nó que envia um datagrama IP em resposta a outro recebido anteriormente. Nos ataques DDoS por reflexão amplificada (AR-DDoS, que serão detalhados na Seção 2.4.1), os refletores de interesse são aqueles que amplificam o tráfego, *i.e.* sua resposta produz mais bytes ou pacotes, ou ambos, que o datagrama orginal de *probe*. Este comportamento é caracterizado por um fator de reflexão, que indica quanto tráfego é gerado pelo refletor. Assim, a amplificação no refletor potencializa o tráfego gerado pelo atacante [Paxson 2001].

Um ingrediente final comum à maioria dos ataques DoS, inclusive AR-DDoS, é a técnica conhecida como *IP spoofing*, que consiste no uso de endereços de origem arbitrários em datagramas IP. Em ataques DoS, os atacantes usam essa técnica para ofuscar a origem real do ataque. Caracteristicamente, em ataques AR-DDoS os atacantes enviam tráfego aos refletores usando o endereço do alvo como origem [Ali 2007]. O protocolos mais comumente usados em AR-DDoS são DNS, NTP, SNMP e SSDP. Eles são *challenge-response* e produzem significativa amplificação em suas respostas [Rossow 2014].

Entre os ataques DoS/DDoS, os ataques por reflexão amplificada são certamente os mais

interessantes sob a perspectiva de um atacante. O principal motivo está no menor esforço, tanto de preparação quanto de execução. A preparação, que tradicionalmente levava à necessidade de se comprometer um número considerável de *hosts*, manualmente ou por uso de uma ferramenta (como em *botnets*), agora se reduz à identificação dos refletores. A ferramenta de identificação de refletores consiste basicamente em construir um *probe*, enviá-lo e monitorar a resposta que, se houver, permite o cálculo do fator de amplificação. Para o ataque, basta enviar o mesmo *probe* usando como endereço de origem o do alvo.

2.4.1 DDoS por Reflexão Amplificada

Como indicado, ataques DDoS constituem uma questão importante na segurança IoT. Antes de definir e discutir os ataques DDoS por reflexão amplificada (AR-DDoS), algumas definições são necessárias, tendo em mente que os casos de interesse são ataques que ocorrem em ambientes de rede. Um ataque de rede é uma ação hostil, agressiva ou mal-intencionada executada usando a rede ou contra a sua infraestrutura ou serviços. Um invasor é uma entidade que executa um ataque. Um ataque de negação de serviço (DoS) é um ataque onde o atacante tenta causar interrupção completa do serviço para que os usuários legítimos não possam acessá-lo. Um ataque DoS em uma de suas formas mais básicas consiste em inundar o serviço com enormes quantidades de tráfego, como solicitações de serviço, ou explorar alguma fragilidade do protocolo, que pode ser motivada pelo design ou pela implementação. No primeiro caso, os ataques são chamados de ataques volumétricos, e os últimos são chamados de baixo volume e taxa lenta (*low volume and slow rate, Low and Slow - L&S*).

Um ataque distribuído de negação de serviço (DDoS) é um ataque DoS onde as ações do ataque são distribuídas entre várias entidades que atuam de maneira coordenada contra o alvo [Nagpal et al. 2015]. À medida em que a infraestrutura evoluiu, com o aumento das capacidades dos enlaces e o amadurecimento do software dos servidores, juntamente com a disseminação de melhores práticas, os ataques DoS evoluíram tornando-se distribuídos. Assim, ao invés de um único atacante atingir o alvo, grupos de *hosts* agem em conjunto para esse fim. Entretanto, a maior eficiência de ataque requer uma extensa preparação, uma vez que uma arquitetura completa se faz necessária, com o atacante passando a ter que comprometer e controlar vários *hosts* antes que o ataque possa vir a ser executado. Estes *hosts* comprometidos formam as chamadas redes zumbis, onde um *host* (o mestre) controla o restante dos *hosts* (os escravos), estando o mestre está sob controle direto do atacante. Esta arquitetura oferece algumas vantagens ao atacante: a identificação da origem do ataque é muito mais difícil; mitigação também é muito mais difícil, pois há mais de uma origem de tráfego de ataque. O próximo passo na evolução do DDoS foi reduzir o esforço de preparação, usando *malware* para infectar, propagar e recrutar *hosts*. Do ponto de vista da arquitetura, os *hosts* comprometidos (*bots*) se reportam a um servidor de comando e controle, sob controle do atacante [Arukonda e Sinha 2015].

Um ataque de negação de serviço distribuído por reflexão amplificada, AR-DDoS, é um

ataque DDoS volumétrico onde uma infraestrutura é abusada a fim de potencializar e redirecionar o tráfego para o alvo. A potencialização é conseguida através da amplificação, enquanto que o redirecionamento proporciona a reflexão. A reflexão é obtida enviando *probes* sobre um protocolo não orientado à conexão que gera uma resposta. Como o protocolo é não orientado à conexão (*connectionless*), os endereços de origem podem ser falsificados (*spoofed*) e a resposta é assim redirecionada para o destino. A amplificação ocorre quando a resposta é maior do que a *probe* de estímulo. Além da elevada eficiência e simplicidade de execução, o que torna o AR-DDoS ainda mais atraente para os atacantes é que seu esforço de preparação pode ser reduzido à detecção de potenciais refletores, que estão disponíveis em grande número.

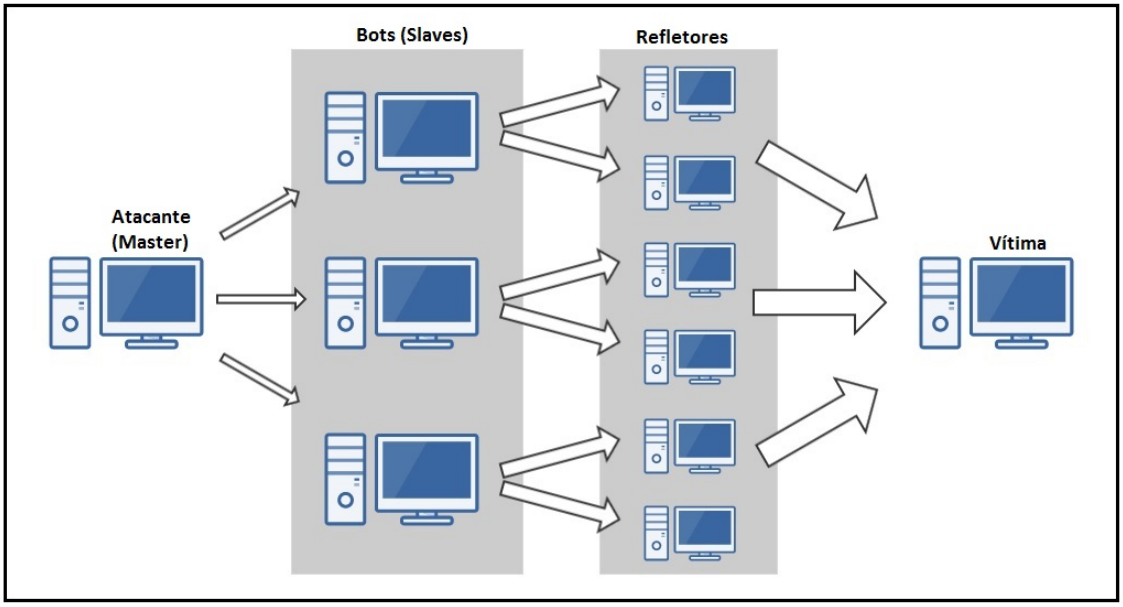

Figura 2.11: Arquitetura de um ataque AR-DDoS

A Figura 2.11 ilustra uma arquitetura típica para um ataque AR-DDoS. Basicamente, é idêntica à habitual de um ataque DDoS, com os refletores antes da vítima para amplificar o tráfego. Como mostrado, há duas camadas entre o atacante e vítima: os refletores, que efetivamente enviam tráfego amplificado diretamente para a vítima; e um conjunto de *hosts* comprometidos (escravos ou *bots*) que são controlados pelo atacante (mestre) e enviam *probes* para os refletores. A rigor, esta camada intermediária não é necessária, pois o atacante pode executar eficientemente um ataque enviando *probes* diretamente para os refletores. Esta camada intermediária cumpre duas funções: contribui para aumentar a escala do ataque e fornece ofuscação de sua origem, protegendo o atacante. Além disso, já está presente na arquitetura DDoS típica.

Em termos de execução do ataque, o mestre envia informações de controle para os *hosts* da camada intermediária para coordenar o ataque. Em seguida, esses *hosts* enviam *probes* montadas com o endereço da vítima como endereço de origem para os refletores. Quando um refletor recebe um desses *probes*, ele envia uma resposta amplificada à vítima.

Em termos de proteção e defesa, os ataques DDoS volumétricos têm algumas melhores práticas já bem estabelecidas. Eles podem ser prevenidos com medidas *anti-spoofing* em redes de acesso. As regras de filtragem de entrada e de saída para evitar o tráfego com endereços forjados é uma prática comumente recomendada. No entanto, nem sempre são implementadas. Quanto à mitigação, para se defender contra um ataque DDoS em execução, a abordagem usual é filtrar o tráfego de ataque o mais próximo possível de sua verdadeira origem. A dificuldade de implementar tal filtragem é que o tráfego de ataque normalmente contém endereços de origem falsificados. Assim, a filtragem baseada em endereço tende a ser ineficiente, filtrando a origem errada e afetando endereços que não estão envolvidos no ataque. A estimativa e atribuição eficientes da origem pode ser obtida através da análise dos fluxos à montante na rede e aplicação de filtros a prováveis endereços de origem. Isso, no entanto, exige uma estreita coordenação entre os Provedores de Serviços de Internet (ISPs). Existem também serviços que oferecem depuração de tráfego, *scrubing*, para um determinado domínio ou serviço. Os serviços a serem protegidos estão conectados ao *scrubber*, que geralmente tem grande largura de banda e capacidade de filtragem, e recebe fluxos limpos sem tráfego de ataque.

No caso particular de AR-DDoS, o tráfego de refletores para o alvo normalmente não tem endereço de origem falsificado e a filtragem de endereço de origem pode ser mais precisa e eficiente se os filtros forem colocados próximos aos refletores.

Alguns protocolos normalmente usados em IoT são vulneráveis a AR-DDoS. As características relevantes destes para AR-DDoS são discutidas na Seção 5.2.1. No caso de IoT, no entanto, existem duas formas como os nós são posicionados que poderiam inerentemente protegê-los de AR-DDoS. Primeiro, uma fração significativa das suas redes não estão ligadas à Internet, e os incidentes são reduzidos a ações internas. O outro fator é que outra fração significativa de redes conectadas à Internet usa Network Address Translation (NAT), que oculta a topologia interna e atua como um filtro de acesso.

Essas duas proteções são contornadas na arquitetura da Figura 2.11. Ao explorar vulnerabilidades no software IoT, como o uso de serviços de login inseguros com credenciais padrão, um invasor pode facilmente implementar uma *botnet* com um número potencialmente enorme de *bots*.

2.5 Internet das Coisas

A primeira referência à expressão Internet das Coisas (*Internet of Things* - IoT), está em [Ashton 2009] que descreve o uso de dispositivos rastreados por RFID na Internet para o controle logístico. [Atzori, Iera e Morabito 2010] oferece uma definição para IoT onde pode-se identificar os três vetores de IoT. Primeiramente, a dimensão de rede e conectividade, evidenciada pela Internet; Segundo, a dimensão dos dispositivos (*Things*, coisas) que abrange um leque maior de objetos genéricos. As diferenças conceituais em relação à IoT, decorrem dos

interesses específicos de quem a observa, que pode ter uma visão mais orientada à conectividade ou mais orientada aos dispositivos. Porém, ao se unir os dois conceitos, forma-se outro que semanticamente significa "uma rede global de objetos conectados, unicamente endereçáveis, baseados em protocolos padrão de comunicação" [Atzori, Iera e Morabito 2010]. A terceira dimensão é a visão semântica da IoT, que trata da visualização, interpretação, busca, endereçamento e armazenamento de todos os dados. Ao se unir essas três visões, obtém-se uma visão macro que compõe a Internet das Coisas, conforme mostra a Figura 2.12.

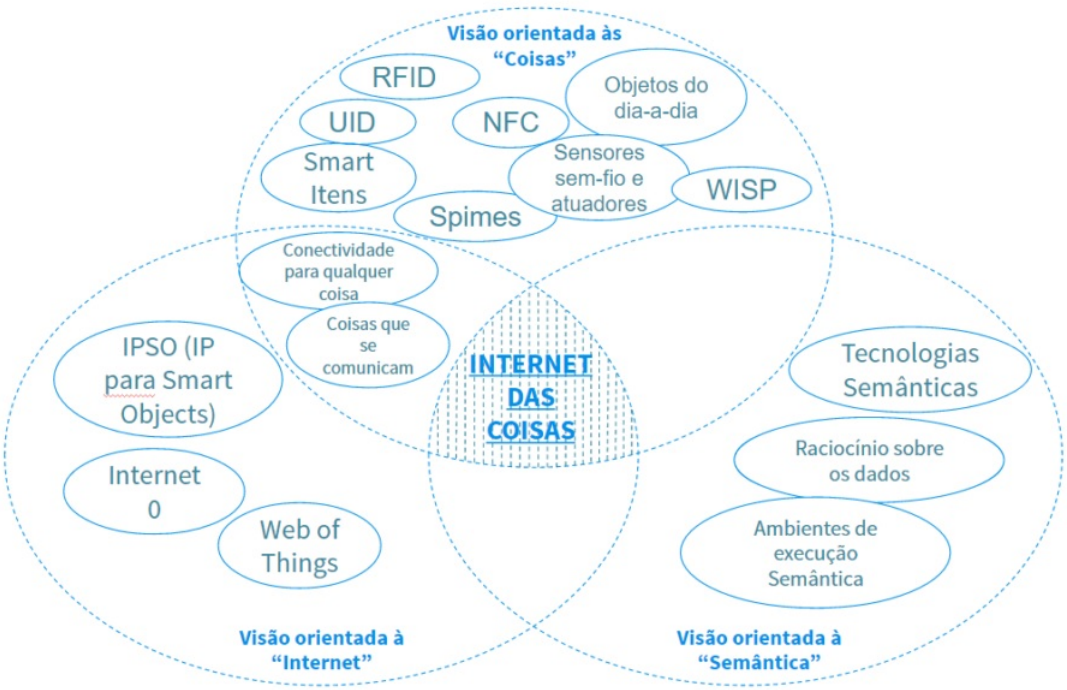

Figura 2.12: As três visões de IoT (adaptado de [Atzori, Iera e Morabito 2010]

[Borgia 2014] define IoT como "computação ubíqua, computação pervasiva embarcados, unidos para formar um sistema em que o mundo real e digital se encontram em uma interação contínua e simbiótica." Apesar das divergências na definição exata do que a IoT representa, alguns pontos são comuns à maioria das definições: trata-se da inclusão de dispositivos, com capacidade de interagir com um meio físico e múltiplos usuários, para obtenção de dados. Além disso, possui em algum estágio comunicação com a Internet e alguma "inteligência" para a análise e utilização dos dados.

2.5.1 Segurança em IoT

A discussão dos aspectos de segurança tem sido intensa. Kumar, Vealey and Srivastava [Kumar, Vealey e Srivastava 2016] oferecem uma *survey* descrevendo possíveis ataques nas várias camadas que formam a infraestrutura de IoT. Yu et al. [Yu et al. 2015] também apresentam uma resenha dos principais desafios de segurança com um possível *road-*

map para abordá-los. Elkhodr, Shahrestani e Cheung [Elkhodr, Shahrestani e Cheung 2016] revisam alguns dos principais temas que desafiam a adoção generalizada da IoT, discutindo interoperabilidade, gerenciamento e gestão, segurança e privacidade. Cvitić, Vujić and Husnjak [Cvitić, Vujić e Husnjak 2016] consideram as camadas que formam IoT e abordam os riscos relativos a cada uma delas. Xylogiannopoulos, Karampelas e Alhajj [Xylogiannopoulos, Karampelas e Alhajj 2016] usam técnicas de mineração de dados para propor sistemas de detecção precoce (*early warning detection*) em infraestruturas de IoT. Yin et al. [Pa et al. 2016] desenvolveram um *honeypot* para IoT com a finalidade de capturar *malware* específicos daquele ambiente.

No tópico específico de ataques DDoS em IoT, o debate também é prolífico. Arış, Oktuğ e Yalçın [Arış, Oktuğ e Yalçın 2015] fazem uma *survey* sobre ataques DoS que podem vir a ter ambiente de IoT como alvo, avaliando sistemas que tentam detectar e mitigar tais ataques. Pras et al. [Pras et al. 2016] argumentam que a análise de ataques recentes demonstra que será relativamente fácil construir as ferramentas de ataque da próxima geração, que afirmam serão capazes de gerar ataques com potencial de intensidade de mil a um milhão de vezes maior que o observado nos ataques atuais. Sonar e Upadhyay [Sonar e Upadhyay 2016] apresentam uma solução baseada em agente para proteger e tratar de ataques DDoS em IoT. Zhang e Green[Zhang e Green 2015] propõem e testam um algoritmo defensivo *lightweight* para ataques DDoS em redes IoT. Singh e Panda[Singh e Panda 2015] também propõem um *framework* de defesa para detecção antecipada de ataques DDoS volumétricos, em que se tenta minimizar a degradação de desempenho da rede. Hu [Hu 2016] descreve ataques DDoS em arquiteturas IoT. Sgouras, Birda e Labridis [Sgouras, Birda e Labridis 2014] avaliam o impacto de ataques DoS/DDoS sobre a confiabilidade de *smart grids* pela análise qualitativa de índices de confiabilidade baseada em simulação.

Em IoT as preocupações de segurança normalmente se voltam aos aspectos relacionados a privacidade, integridade e controle de acesso. Aqui, o interesse recai sobre a disponibilidade, em particular os ataques DDoS que a afetam.

A percepção geral é que o impacto devido às falhas de segurança será maior, seja pelos dispositivos desempenharem funcionalidades críticas, ou pelo seu grande número, ou ambos. Uma das grandes preocupações é o abuso da infraestrutura de IoT por ataques DDoS, tanto para ser alvo ou atacar outras infraestruturas ([Jackson 2013], [Cox 2013]). Há relatos de ocorrência de ambas as formas de incidentes na Internet ([Sharon 2015], [Toms 2016]). Entretanto, a avaliação da ameaça se dá mais baseada em inferência que evidência empírica.

2.5.2 Indisponibilidade e IoT

Ainda entre os objetivos deste estudo, há uma referência explícita quanto a aplicabilidade da metodologia proposta a ataques contra a disponibilidade. Esta avaliação da aplicabilidade a ataques de indisponibilidade se deu em ambiente IoT.

A segurança é uma preocupação profunda no contexto das redes de computadores e da IoT, com a sua discussão focada na privacidade, integridade, controle de acesso, disponibilidade e resiliência. Com relação à disponibilidade, os ataques DDoS estão no topo da lista, visando a infraestrutura e os dispositivos IoT ou abusando deles para atingir terceiros. O consenso na comunidade de segurança é que ataques DDoS representam uma séria ameaça e já se tem propostas para tentar lidar com as ameaças. No entanto, além da inferência, a avaliação qualitativa e quantitativa de possíveis impactos de ameaças à IoT é necessária para o tratamento adequado.

A maneira usual de avaliar a disponibilidade não leva em conta explicitamente as condições de um ataque. Assim sendo, se os testes são para avaliar impactos de ataque, eles têm que incluir o ponto de vista do atacante para uma avaliação precisa. Esta é a motivação para uma metodologia de avaliação de vulnerabilidades que não só as identifica, mas tenta ativamente explorá-las.

A IoT, juntamente com os serviços e dispositivos e sensores, estão na superfície de ataque oferecida e estão expostos a várias ameaças, sendo as mais relevantes as voltadas à disponibilidade, como ataques de negação de serviço (DoS). Os ataques DoS visam esgotar recursos de dispositivos ou de infraestrutura, a fim de causar interrupção de serviço e portanto a indisponibilidade. Em sua forma volumétrica, os ataques consistem basicamente em enviar um grande número de solicitações que sobrepujam algum recurso de um serviços quando este tenta processá-los, ou exceder os limites de tráfego [McDowell 2009]. Uma forma comum de implementar e potencializar ataques DoS está na sua distribuição (DoS distribuído, DDoS), onde vários nós enviam tráfego de forma coordenada, resultando em maior eficiência de ataque e mais demandando um esforço de mitigação mais complexo dada a ofuscação da fonte.

A título de ilustrar, em termos práticos e realistas, o tipo de ameaça que se colocava à IoT, dois casos recentes são apresentados. O primeiro é um ataque AR-DDoS de grande volume e o outro um ataque DDoS que envolveu mais de 145 mil dispositivos IoT.

a. **Ataque de 300 Gbps por Amplificação usando DNS**:

   Em março de 2013 um grande ataque AR-DDoS por amplificação de DNS foi dirigido contra um serviço *antispam* de renome na Internet [Bright 2013]. O incidente foi tratado por um dos principais serviços de *scrubbing* de tráfego que hospedava o serviço [Prince 2013]. Nesse tipo de ataque, a amplificação é alcançada abusando resolvedores de DNS, que estão abertos para consultas recursivas. Especificamente, uma consulta do tipo **ANY** é enviada para o resolvedor de DNS. Ao tratar uma consulta **ANY**, o resolvedor responde, enviando informações correspondentes ao domínio consultado. Para reflexão, a consulta falsifica o endereço de origem usando o da vítima.

   O ataque usou a arquitetura AR-DDoS usual. Estima-se que 30.000 servidores DNS distintos foram usados como refletores. Isso representou cerca de 0,1 por cento de todos os potenciais refletores disponíveis na Internet momento do ataque. Como taxa

típica amplificação do DNS atinge entre 28 e 54 vezes [US-CERT 2014], sendo o fator médio de 40 vezes, a camada intermediária de *bots* precisaria gerar cerca de 750 Mbps, o que é possível com uma *botnet* de pequeno porte (menos de mil *bots*).

O ataque começou em 18 de março com cerca de 10 Gbps. Em 19 de março, escalou para 90 Gbps, e em 22 de março atingiu 120 Gbps. Em 27 de março, o ataque chegou a 300 Gbps de tráfego. Este foi o maior ataque DDoS relatado até então.

O ataque foi tratado corretamente pelo *scrubber* do tráfego, com a cooperação de ISPs a montante.

b. **Ataque DDoS envolvendo mais de 145.000 dispositivos IoT**:

Em 19 de setembro de 2016, houve relatos de um ataque massivo com tráfego da ordem de 1,1 Gbps, que se estabilizou em 900 Gbps, contra um serviço de hospedagem na Internet [Goodin 2016]. Os ataques duraram mais de uma semana, com taxas que variaram de 100 Gbps a 800 Gbps. Durante a resposta ao ataque, foi identificado que ele partiu de uma *botnet* consistindo de câmeras conectadas à Internet e gravadores de vídeo digitais (DVRs), típicas em aplicações de *smart city* e *smart home*, comprometidas. Estima-se que cada *bot* contribuiu com algo entre 1 Mbps e 30 Mbps, com a capacidade máxima de ataque atingindo 1,5 Tbps.

Aparentemente, esta enorme coleção de *bots* foi comprometida, explorando vulnerabilidades em dispositivos *smart*, como credenciais default e chaves criptográficas SSH (*Secure Shell*) codificadas em hardware.

Houve relatos no passado de abuso de 25.000 câmeras de circuito fechado de TV conectadas à Internet (CCTV) em ataques DDoS, e até mesmo refrigeradores conectados à Internet enviando *spam*. Contudo, o tráfego gerado foi modesto quando comparado às taxas relatadas.

Um aspecto relevante do ataque descrito é o fato de que não envolveu amplificação. Ou seja, o volume de tráfego foi produzido basicamente pelo grande número de dispositivos comprometidos. Assim, é de se esperar ataques de maior volume no futuro, quando as técnicas de amplificação forem incorporadas aos vetores de ataque.

2.6 Resumo do Capítulo 2

Conforme indicado, neste capítulo foram apresentados os conceitos e as metodologias de testes de penetração. Estas foram discutidas e comparadas visando fundamentar a elaboração da metodologia proposta. Na sequência, foram também apresentados os conceitos de IoT e o debate sobre sua segurança, especialmente os ataques de negação de serviço.

Capítulo 3

Descrição do Problema

Conforme revisado na Seção 2 2.2, existem várias metodologias que definem e orientam o processo de avaliação ativa de vulnerabilidades. Como ficou evidente, as metodologias estudadas não atendem integralmente ao que se espera delas em relação a testes de penetração. Em alguns casos, apresentam deficiências e inconsistências.

Como já foi apresentado na Seção 1.2, o objetivo principal deste trabalho é desenvolver uma metodologia para realização de processos de prospecção ativa de vulnerabilidades. Para esse processo em si, as seguintes características são requisitos: explicitação dos critérios de decisão e escolha de procedimentos, técnicas e ferramentas; generalidade de escopo, incluindo testes de ameaças à disponibilidade; flexibilidade de aplicação; independência de ferramentas; modelagem geral da execução de testes e ataques; e por fim, consistência e estruturação.

Inicialmente, será feita uma análise, seguida de uma classificação, das metodologias estudadas no Capítulo 2. A seguir, cada um dos aspectos acima será abordado visando definir requisitos e guiar a construção da metodologia desenvolvida.

3.1 Análise das Metodologias Estudadas

As metodologias são consideradas são analisadas abaixo.

3.1.1 NIST

A impressão geral é que o documento mantém sua coerência e elenca diretrizes de planejamento para os testes, sempre com ênfase nos aspectos gerenciais. Do ponto de vista estritamente técnico, há descrições de procedimentos em nível razoável de detalhes, entretanto sem descer ao nível de aplicação de ferramentas. Há um apêndice dedicado a recomendar ferramentas para a realização dos testes de penetração, bem como de documentos auxiliares para a formulação e documentação de metodologias.

Resumindo, apesar de se apresentar como um guia técnico para avaliação de segurança de sistemas, o documento em questão se atém mais aos aspectos gerenciais do planejamento e execução dos testes. Não há, entretanto, diretivas específicas para a escolha dos alvos e a condução dos testes em si. É razoável estimar que isto decorre da inserção dos testes de penetração no contexto específico da análise de risco, que por sua vez deveria vir a prover o direcionamento e os subsídios para a condução dos testes.

A orientação quanto à tomada de decisão, como já indicado, é principalmente de ordem gerencial. Assim, é como se o processo do teste de penetração tivesse um objetivo genérico de validar riscos e vulnerabilidades identificados, avaliando a eficácia dos controles empregados, se presentes. Dessa forma, a condução dos testes fica ainda mais dependente da experiência do testador.

Outro aspecto a mencionar, é que caso se deseje atingir um alvo específico, a forma relativamente episódica como as vulnerabilidades são encontradas e testadas pode levar a que este se perca entre outros alvos e não seja atingido ou mesmo explorado devidamente.

Por fim, estas diretivas técnicas, aparentemente pela própria proposta do documento, não consistem em uma metodologia em si, mas em linhas metodológicas gerais e da descrição, também geral, de procedimentos e, finalmente, ferramentas que juntas dariam ao interessado condições de elaborar seus próprios procedimentos. Novamente, isto pode ser consequência do enquadramento dos testes de penetração como parte da análise de risco e não como um processo de per si.

3.1.2 OSSTMM

A metodologia apresentada é bem detalhada e complexa. Por se propor atuar no contexto da auditoria de sistemas, absorve toda um *framework* que lhe impõe um *overhead* de esforço considerável. Além disso, por quase definir seu próprio quadro conceitual, por vezes não totalmente aderente à cultura e à prática de testes de penetração, demanda a formação de habilidades por parte do executor dos testes que extrapolam o perfil técnico em detrimento do gerencial.

A definição das tarefas, com alto nível de detalhamento, leva à necessidade de ferramentas para sua execução. Estas, apesar de não estarem especificadas explicitamente, estão fortemente determinadas pelo detalhamento das tarefas. Em alguns casos, o detalhamento leva a testes que não são aplicáveis ou de grande relevância no contexto da avaliação.

Outro aspecto a se considerar é a proposta de se prover um resultado quantitativo para a avaliação de segurança. Não obstante a oportunidade em se ter medidas objetivas e numéricas, neste caso a operacionalização que se pretende dar às mesmas pode ser enganosa. O primeiro aspecto que se deve atentar é que, mesmo com ponderações e contra pesos, um número que reflete a contabilização de vulnerabilidades e controles enquanto conveniente pode não refletir aspectos importantes da avaliação de segurança. Assim, a proposta de servir

como suporte a decisões de investimento e acompanhamento levaria a erros de avaliação.

3.1.3 PTES

O padrão conforme descrito não apresenta diretrizes técnicas quanto à forma de executar um *pentest* real. entretanto, há um guia técnico acompanhando o padrão,*Technical Guidelines*, onde são descritas ferramentas e procedimentos específicos. Assim, o foco da execução dos testes em si recai sobre a aplicação de um conjunto de ferramentas.

Por vezes, o processo como um todo apresenta inconsistências. Por exemplo, como a seleção de alvos precede a análise de vulnerabilidades, que só ocorre após a execução dos testes, os alvos são selecionados sem seguir uma orientação clara de relevância com relação ao que faz sentido para os sistemas sendo testados. A impressão é que os alvos são selecionados na medida que são encontrados e então explorados para depois se avaliar os impactos da exploração.

Essa abordagem, além de enfraquecer o planejamento global do processo de testes de penetração, pode afetar a completude e a profundidade dos resultados.

3.1.4 PCI-SCC DSS

A metodologia, como descrita, estaria mais voltada para a definição de requisitos do que se deve abordar em um processo de auditoria do que realmente em algo específico voltado a testes de penetração. Pela abordagem demonstrada, os testes são o principal instrumento desse processo de auditoria. Entretanto, nas diretivas não há de forma clara orientações quanto às decisões a serem tomadas, que critérios devem guiar tais decisões, bem como indicações de como conduzir os testes de penetração.

Por sua ênfase na auditoria, pode-se afirmar que a metodologia PCI-DSS está mais preocupada com conformidade e gestão de riscos. Pode-se argumentar que tal enfoque seja consequência da especificidade dos sistemas ao qual a metodologia PCI se dedica. Contudo, tal especificidade deveria motivar critérios específicos para decisão, indo além de indicação de condições limitadoras na execução dos testes.

Deve-se também destacar que parece haver uma imprecisão conceitual quanto ao que seria a metodologia. Nas recomendações para o relatório, tem-se a recomendação para descrição da metodologia utilizada, pedem-se os detalhes das metodologias usadas para completar o teste e se indicam explicitamente: varredura de portas, NMAP, etc; e também a relação das ferramentas utilizadas. Assim, há uma confusão entre procedimento e metodologia.

A consequência da aplicação desta metodologia, que é padrão de mercado para um segmento crítico da atividade econômica, é a dependência da qualidade do teste e dos seus resultados no expertise do testador, não apenas no seu conhecimento particular de ataques e ferramentas, mas principalmente no seu discernimento quanto ao planejamento global dos

testes e sua condução.

3.1.5 SANS

Da forma como descrita, a metodologia é definida de forma relativamente vaga com indicações gerais das fases porém sem o necessário aprofundamento. Da forma como apresentadas, consiste em um conjunto de diretivas sem o devido detalhamento.

Ainda que as diretrizes sejam precisas e consistentes com o que se espera para um processo de testes de penetração, a principal consequência dessa abordagem é que a obtenção de resultados é dependente da experiência e conhecimento do executor dos testes. Assim, a completude, profundidade e repetibilidade dos resultados podem ser fortemente comprometidas.

3.1.6 OWASP

Conforme descrito, OWASP apresenta uma metodologia detalhada e abrangente porém voltada a aplicações web, sendo assim de escopo restrito. Seu foco não está nos testes de penetração mas no desenvolvimento de aplicações web seguras. Com respeito aos aspectos metodológicos de testes de penetração, estes não são definidos ficando apenas uma indicação de possíveis metodologias que podem ser utilizadas (coincidentemente, as que foram estudadas no Capítulo 2).

Assim, OWASP não se qualifica como uma metodologia de testes de penetração.

3.2 Classificação das Metodologias Estudadas

O que caracteriza as metodologias acima discutidas é a ênfase nos aspectos procedurais dos *p*entests. Em alguns casos, a apresentação dos procedimentos é de forma didática, como se também tivesse por finalidade a formação de executores de tais testes, com o esforço de formação no domínio das ferramentas.

3.2.1 Quanto à Finalidade

Do ponto de vista da finalidade, as metodologias divergem com respeito ao foco em testes de penetração. Apenas as metodologias PTES e SANS são dedicadas exclusivamente aos testes de penetração em si. A metodologia do NIST insere os testes de penetração no contexto maior da análise e gerenciamento de risco, enquanto a OSSTMM os insere no ferramental de auditoria de segurança para avaliação sob a perspectiva operacional pela estimação da superfície de ataque oferecida. A metodologia PCI-DSS também instrumentaliza os testes

de penetração dentro do contexto da auditoria de risco. E, por fim OWASP também os considera instrumentais para o desenvolvimento de aplicações web seguras conforme Tabela 3.1(nesta e nas demais Tabelas N/A significa "‘Não se aplica"’).

Tabela 3.1: Quadro Comparativo das Metodologias - Finalidade

Metodologia	**Finalidade**
NIST	gerenciamento e análise de risco
OSSTMM	auditoria de segurança
OWASP	desenvolvimento seguro de aplicações web
PTES	testes de penetração
PCI-DSS	auditoria de risco
SANS	testes de penetração

3.2.2 Quanto ao Escopo

Com respeito ao escopo de aplicação, há um predomínio das metodologias de espectro geral, como NIST, OSSTMM, PTES e SANS; enquanto PCI-DSS (aplicação PIN) e OWASP (aplicações web são voltadas a domínios específicos (vide Tabela 3.2).

Tabela 3.2: Quadro Comparativo das Metodologias - Escopo

Metodologia	**Escopo**
NIST	geral
OSSTMM	geral
OWASP	aplicações web
PTES	geral
PCI-DSS	aplicações PIN
SANS	geral

3.2.3 Quanto ao Processo

A metodologias NIST e OSSTMM são muito complexas, porém sob diferentes aspectos: a primeira por sua ênfase em detalhamento de aspectos gerenciais; a última por definir todo um quadro conceitual próprio para então definir processos e procedimentos. As metodologias PCI-DSS, PTES e SANS definem o processo, e não são detalhadas (PCI-DSS apresenta um pouco mais de detalhes que PTES, que é mais detalhado que SANS). OWASP não define o processo, nem o detalha: apenas indica possíveis metodologias para testes de penetração - coincidentemente, as aqui apresentadas (vide Tabela 3.3).

Tabela 3.3: Quadro Comparativo das Metodologias - Processo

Metodologia	Processo		
	Definido	Detalhado	Complexo
NIST	✓	✓	✓
OSSTMM	✓	✓	✓
OWASP	N/A	N/A	N/A
PTES	✓		
PCI-DSS	✓		
SANS	✓		

Do ponto de vista descritivo, as metodologias acima se utilizam da descrição operacional dos procedimentos, recomendando fortemente o uso de ferramentas (algumas listam as ferramentas sugeridas). A única metodologia que apresenta um modelo de execução de ataque é a NIST. Entretanto, a modelagem é restrita pois contempla apenas ataques por infecção, com ciclo intrusão-escalação-persistência, típico de *malware* e ferramentas de ataque (vide Tabela 3.4).

Tabela 3.4: Quadro Comparativo das Metodologias - Descrição do Processo

Metodologia	Descrição do Processo		
	Operacional	Ferramentas	Ataque
NIST	✓	✓	✓
OSSTMM	✓	✓	
OWASP	N/A	N/A	N/A
PTES	✓	✓	
PCI-DSS	✓	✓	
SANS	✓	✓	

Sob a perspectiva da organização do processo de teste de penetração, as metodologias apresentadas não explicitam os processos de decisão necessários para seleção de alvos e condução de testes. Na metodologia PCI-DSS, a organização de fases apresenta uma inconsistência uma vez que a análise de vulnerabilidades ocorre após os testes. Essa inconsistência se agrava por não haver uma descrição clara do *rationale* das decisões que devem ser tomadas ao longo do processo de testes de penetração. Deve-se ressaltar que essa ausência da descrição do *rationale* é comum entre as metodologias discutidas (vide Tabela 3.5).

Tabela 3.5: Quadro Comparativo das Metodologias - Organização do Processo

Metodologia	Organização do Processo		
	Consistente	Separação	Decisão
NIST	✓	✓	
OSSTMM	✓	✓	
OWASP	N/A	N/A	N/A
PTES	✓		
PCI-DSS			
SANS	✓		

Outra inconsistência, também na PCI-DSS, é confundir o método com o procedimento. OSSTMM faz uma separação clara dos conceitos, no que é seguida por NIST.

Como se vê, as metodologias revisadas se orientam a processos, focando em operações e com forte suporte de ferramentas, que de certa forma determinam seus procedimentos.

3.3 O Aspecto Metodológico

A palavra metodologia é derivada de três outras de origem grega: *metha*, "para além de"; *odos*, caminho; e *logus*, estudo. Assim, seria o estudo dos métodos, aí entendidos como as formas de se construir o caminho para se realizar algo. No contexto da ciência, seria a investigação, organização e sistematização dos métodos de investigação que permitem atingir resultados que refletem objetivos. Ainda no contexto da investigação científica, a metodologia seria o conjunto dos métodos que regem uma investigação. Note-se que a metodologia é assim parte do processo de investigação pela sistematização de métodos e técnicas. Além disso, é etapa específica anterior à investigação que considera os critérios de escolha e seleção de métodos para investigação.

Assim, o foco da metodologia é o suporte às decisões que orientam a condução do processo de investigação, provendo os critérios para escolha e seleção de métodos, procedimentos e ações específicos. De forma resumida, pode-se afirmar que o método está para os procedimentos, enquanto a metodologia guia todo o processo de investigação, principalmente as decisões referentes à sua condução.

Retornando às metodologias estudadas na Seção 2.2, observa-se que dentre as que foram estudadas a maioria não se encaixa no que se espera de uma metodologia. Apenas as metodologias do NIST e a OSSTMM estariam aderindo ao conceito, por suprir critérios e guiar decisões, enquanto as outras são bem focadas nos aspectos procedurais, ainda que sob o formato de métodos.

Dessa forma, justifica-se propor uma metodologia com forte ênfase em sistematizar o processo de testes de penetração como uma investigação sobre a presença, abrangência e impactos específicos de vulnerabilidades, provendo critérios e guiando decisões na forma de escolhas e seleções.

3.4 Finalidade

Observando-se as metodologias estudadas na Seção 2.2, fica evidente que nem todas tem por finalidade a execução de testes de penetração. Apenas, PTES e SANS são específicas para testes de penetração enquanto as outras os consideram como um elemento de um processo mais amplo. Recordando, NIST tem por finalidade gerenciamento e análise de risco; OSSTMM e PCI DSS tem por objetivo auditoria de segurança e de risco, respectivamente;

OWASP é voltado ao ciclo de vida desenvolvimento de aplicações web, e sequer define metodologia de testes de penetração específica.

O principal efeito desta falta de foco nos testes de penetração é que o detalhamento apresentado, como na metodologia do NIST ou OSSTMM, não é voltado aos testes de penetração, mas às suas áreas finais, e no caso destas, a análise de risco.

Assim, há a necessidade de uma metodologia de testes de penetração que lhes seja específica, seguindo o aspecto metodológico descrito anteriormente, com nível de detalhamento apropriado.

3.5 A Modelagem da Execução dos Testes ou Ataques

O cerne de qualquer metodologia de testes de penetração é a execução dos testes ou ataques. Entretanto, o recurso utilizado pelas metodologias estudadas para modelar este aspecto crucial não se apresenta satisfatório. Apenas a metodologia do NIST arrisca uma modelagem explícita que se pretende geral para a execução dos testes. Outras recorrem à descrição de procedimentos ou prescrição de conjuntos de ferramentas.

Na modelagem adotada pelo NIST, há um ciclo de ataque (Figura 2.2) que segue as etapas que um *malware* seguiria em um ataque por infecção. O primeiro passo é uma intrusão, seguida de uma escalação de privilégios e uma busca por novos alvos (movimento lateral). Da busca, pode-se seguir para a instalação de mecanismos que levem à persistência do ataque, ou um retorno à descoberta de vulnerabilidades e alvos para intrusão.

Não obstante o mérito em se ter uma descrição que não desce aos detalhes operacionais das ações, esta modelagem do ciclo de execução de ataque é restritiva, pois exclui outras formas de ataques que não precisem de escalação ou busquem movimentação lateral. Além disso, a dupla realimentação com inícios e fins distintos aumentam a complexidade e possibilitam mais retrabalho na execução dos testes.

A lição que a metodologia do NIST nos ensina é que apesar da ideia do ciclo de ataque ser boa, o ciclo escolhido não atende aos requisitos de generalidade de uma modelagem uma vez que se restringe a ataques por infecção tipicamente implementados por *malware*.

Como subsídio para uma modelagem geral do ciclo de ataque foram considerados dois ciclos de decisão, conforme apresentados na Seção 2.3. Entretanto, estes não se mostraram adequados como base para a modelagem da execução de ataques.

Como se pode observar, os ciclos como apresentados não se mostram apropriados para a modelagem da execução de testes em *pentests*. O PDCA por sua dinâmica lenta com ênfase em longos ciclos de planejamento.

Já o OODA, apresenta um nível de sofisticação vai além dos que se observa em situações de ataques cibernéticos, em que os papéis de atacante e atacado são bem definidos,

lembrando ainda que os testes que serão executados são ataques cibernéticos consentidos e executados sob condições controladas. Dada a característica dos ataques cibernéticos pode-se até entender tal característica como um requisito para aplicação da metodologia.

Assim, ainda que o ciclo OODA ofereça a característica dinâmica necessária para a modelagem de ataques, sua sofisticação parece extrapolar o que seria necessário para modelar um ataque cibernético acrescentando uma complexidade que não redundaria em benefício. Isto é, como os defensores respondem de forma programada aos estímulos de ataque e medidas interativas, e respostas advindas de intervenção humana atuam primordialmente na contenção, normalmente sem retaliar, a fase de orientação se esvaziaria na forma de uma análise de estímulos e respostas, que pode ser antecipada em fases anteriores de planejamento.

Assim, o OODA, apesar de acomodar interações mais dinâmicas, insere um nível de complexidade que não acrescenta à modelagem.

Na metodologia proposta, a abordagem adotada foi, partindo-se do ciclo OODA, propor-se um ciclo mais simples e adequado para modelar a execução de ataques.

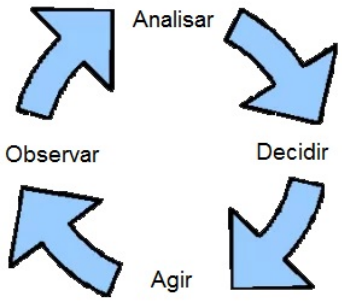

Figura 3.1: Ciclo OADA

O ciclo *Orientar-Analisar-Decidir-Agir* (OADA), ilustrado na Figura 3.1, simplifica as interações e realimentações do ciclo OODA, levando em conta a dinâmica das interações que ocorrem em testes e ataques, e lembrando que normalmente os papéis de atacante e atacado não se alternam entre as partes.

Os passos do ciclo OADA são descritos na tabela 3.6 abaixo.

Tabela 3.6: Etapas do Ciclo OADA

Etapa	Descrição
Observar	diz respeito ao monitoramento do sistema sob teste tanto antes quanto após sua aplicação.
Analisar	consiste no processamento das informações e do comportamento observado, avaliando o sucesso do teste quanto a objetivo e comportamento esperado e sugerindo possibilidades de ação posterior.
Decidir	envolve a escolha da direção a seguir, motivada pela formulação de uma hipótese sobre o objetivo e o resultado esperado do teste a ser executado.
Agir	onde efetivamente se dá a implementação da ação

Este ciclo é geral o suficiente para acomodar adequadamente diversas formas de ataque além da intrusão, inclusive ataques contra a disponibilidade. Também não se faz mais necessário explicitar os passos operacionais que o testador deverá executar. É sificiente que no planejamento dos testes se antecipe as situações e decisões que precisarão ser tomadas. Além disso, a modelagem é aderente ao conceito de metodologia, tendo como objeto da modelagem as decisões que devem ser tomadas quando da execução do ataque. Dessa forma, a metodologia será flexível o suficiente para absorver a evolução de procedimentos, técnicas e ferramentas, além de ter seu escopo expandido a ataques que as metodologias estudadas não contemplavam.

3.6 A Questão das Ferramentas

A forma adotada pelas outras metodologias estudadas recai na enumeração, seguida de descrição, dos procedimentos específicos referentes aos testes. A ênfase então se dá sobre os aspectos operacionais e procedurais dos testes e não nas decisões que guiam o processo. O primeiro grande problema de enumerar procedimentos e operações é sua incompletude e perda de generalidade. Se surge uma nova técnica de teste ou ataque, se esta não for explicitamente agregada à metodologia, não será executada. A consequência é a necessidade de atualização constante e o encurtamento do ciclo de vida da metodologia.

Assim, apesar de possivelmente definir processos e atividades, as metodologias que recorrem à enumeração e descrição, tendem por se concentrar em aspectos operacionais e comumente baseiam suas ações em conjuntos de ferramentas recomendadas ou pré definidas.

O uso de ferramentas não é um problema em si. Elas normalmente dão escalabilidade e repetibilidade ao processo de testes. Contudo, podem reduzir o escopo e a aplicabilidade de uma metodologia impactando ainda mais o encurtamento de seu ciclo de vida.

O cenário de segurança é extremamente dinâmico e uma metodologia acoplada a um conjunto de ferramentas gera uma destas três situações:

1. as ameaças evoluem e novas ferramentas juntamente com procedimentos devem ser incluídos na metodologia;

2. as medidas defensivas evoluem e as ferramentas tornam-se menos eficientes ou precisam ser atualizadas, e consequentemente também a metodologia;

3. sistemas se tornam obsoletos e a necessidade de evolução ocorre como nos casos anteriores. Ou seja, a obsolescência das ferramentas leva à obsolescência da metodologia.

Assim, as metodologias baseadas em ferramentas têm ciclos de vida curtos ou têm de evoluir continuamente. A abordagem para evitar esta situação é fazer com que a metodologia

venha a ser independente de ferramentas, ou agnóstica em relação a elas: em vez de ter suas atividades definidas pelos instrumentos adotados, deverá proporcionar os meios de definir os instrumentos a serem usados no processo. A metodologia então deve ser dedicar ao *rationale* que define ações (isto é, decisões), e deve fornecer orientações, critérios e diretivas sobre o que deve ser considerado para decidir o curso de ação durante a execução de um ataque.

Deve-se ressaltar ainda que as metodologias que recorrem à enumeração e descrição padecem do mesmo efeito, ainda que o encurtamento do ciclo de vida se dê de forma menos acentuada.

A conclusão é que o fator decisivo do processo não devem ser as ferramentas. Ao contrário, as ferramentas devem ser escolhidas, e se necessário até desenvolvidas, pelas necessidades específicas da execução dos testes de penetração. Este princípio estará presente na metodologia proposta.

3.7 Consistência e Organização do Processo

Um aspecto comum às metodologias estudadas é uma estrutura serial que segue o *template*: preteste, teste, posteste e relato, que será adotada como base para a metodologia proposta. Entretanto, alguns refinamentos serão necessários.

Antes de começar a execução de um ataque, uma série de ações preparatórias se faz necessária: a definição em alto nível do que deve ser testado e em que condições; a coleta de informação básica sobre os sistemas a serem testados; o planejamento do ataque, do nível estratégico aos detalhes executivos. Após o teste, vem toda a análise dos resultados e o relato e apresentação dos resultados. Permeando todo o processo, tem-se a geração sistemática de registros de decisões, justificativas, atividades executadas e seus resultados.

Assim, a estrutura básica da metodologia proposta segue a forma: especificação dos testes, levantamento inicial de informações, planejamento do *pentest*, execução dos testes, consolidação e análise e apresentação dos resultados.

Esses são os princípios que orientaram o desenvolvimento de uma metodologia, DOTA, para AVA baseada em uma abordagem orientada à tomada de decisão, agnóstica em relação a ferramentas. A formulação de DOTA tem por finalidade abranger vulnerabilidades em geral, sendo flexível o suficiente para acomodar não só ataques de intrusão, mas outras formas de vulnerabilidades, como DoS.

3.8 Resumo do Capítulo 3

Neste capítulo, foi apresentada a análise crítica e a classificação das metodologias estudadas no Capítulo 2 deixando clara a existência de lacunas conceituais e inadequações

nas diversas abordagens. Na sequência, foram discutidos os requisitos para a elaboração da metodologia desenvolvida neste trabalho e apresentada no próximo capítulo.

Capítulo 4

Metodologia para Prospecção Ativa de Vulnerabilidades

Este capítulo descreve a metodologia de testes de penetração, que visa prover definição e padronização dos procedimentos de identificação e avaliação de vulnerabilidades. A metodologia, por explicitar os critérios e processos de tomada de decisão e por ser independente de ferramentas específicas será referida como DOTA (de *Decision Oriented and Tool Agnostic*).

4.1 Estrutura Geral

A metodologia DOTA é composta por seis fases e pelos processos e atividades que compõem cada uma delas. A metodologia é sequencial, ou seja, os resultados de cada fase são insumos para as fases subsequentes.

As seis fases são:

I. Especificação do Teste de Penetração.

II. Levantamento Inicial.

III. Planejamento do *Pentest*.

IV. Execução dos Testes de Penetração.

V. Consolidação e Análise.

VI. Apresentação dos Resultados.

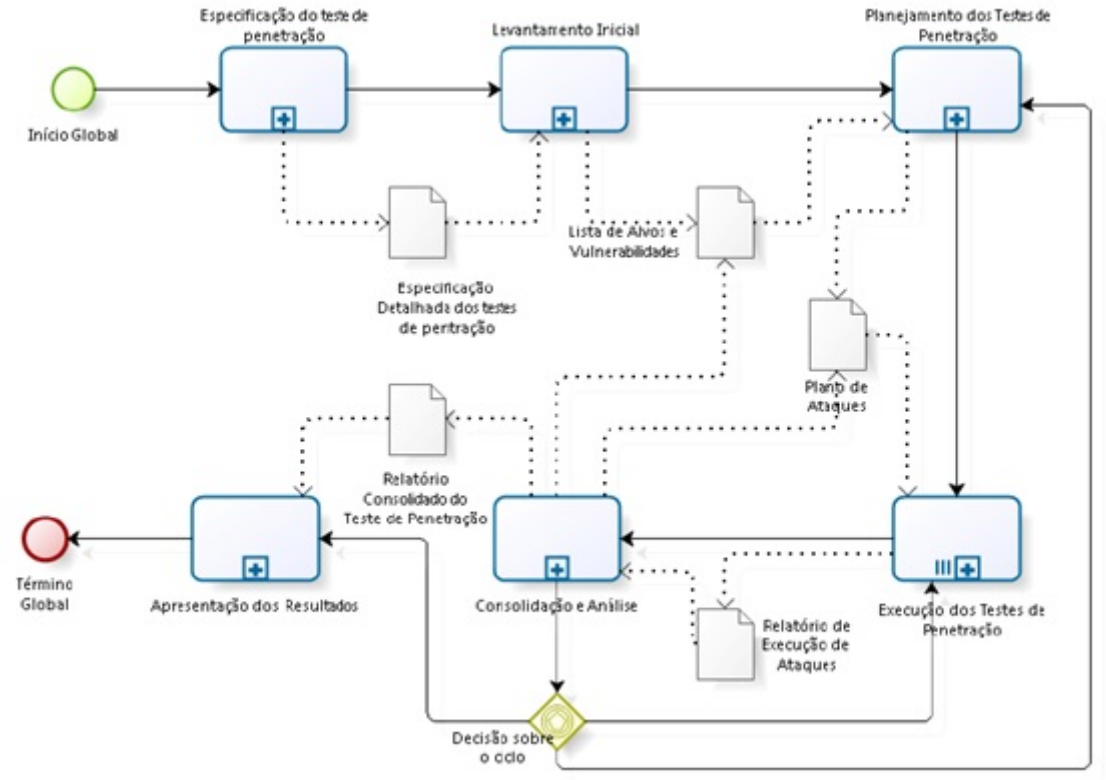

Figura 4.1: Diagrama de Fluxo Geral

A estrutura geral da metodologia DOTA é mostrada na Figura 4.1 sendo cada uma das fases brevemente descritas a seguir.

4.1.1 Fase I: Especificação do Teste de Penetração

Esta fase tem por finalidade definir os objetivos do teste de penetração e suas condições de execução. Nela serão definidos o escopo e o foco em que os testes serão executados. Também serão definidos as formas e limites de atuação da equipe de execução. Deve-se também definir como as informações obtidas durante os testes de penetração serão tratadas e a quem os resultados deverão ser apresentados. Ao seu final serão gerados documentos que consolidam estas informações. Deve-se também obter formalmente o conhecimento e consentimento da organização em que os testes serão realizados.

4.1.2 Fase II: Levantamento Inicial

Esta fase tem por objetivo realizar o reconhecimento e levantamento dos sistemas de informação a serem testados, bem como suas possíveis vulnerabilidades. Os resultados desta fase fornecerão o subsídio para a fase seguinte, na qual serão determinados os alvos. As varreduras são realizadas em vários níveis, começando pela rede, passando por sistemas e

serviços até aplicações, seguindo o escopo definido na Fase I.

4.1.3 Fase III: Planejamento do Teste de Penetração

O objetivo desta fase é definir o planejamento estratégico, tático e executivo dos testes de penetração. De posse dos possíveis alvos, os prioritários são definidos, sendo montados os planos de ataque. Nestes planos deve-se incluir a escolha específica das táticas bem como a forma de execução.

Com as informações acerca dos serviços e aplicações, bem como suas vulnerabilidades, pode-se definir a estratégia de abordagem dos sistemas de informação a serem testados, bem como seu planejamento tático e executivo, sendo seus principais resultados a escolha dos alvos segundo o critério de acesso à informação e os planos de teste (ataque).

Prioritariamente, tem-se como alvos os sistemas que normalmente hospedam informações da atividade núcleo da organização alvo, ou que são críticas para o cumprimento de sua missão. Tipicamente estes alvos são os bancos de dados corporativos, os serviços de informação e comunicação corporativa, como correio eletrônico e intranet (ainda que o teste de penetração seja executado externamente à organização). Também são alvos os sistemas de proteção e controle de acesso aos alvos prioritários.

4.1.4 Fase IV: Execução dos Testes

Esta é a principal fase do processo, em que os testes são de fato realizados. As fases anteriores são a preparação desta, e a seguinte a consolidação de seus resultados.

A execução dos teste é modelada segundo um ciclo de tomada de decisão geral o suficiente para acomodar os ataques sem que seja necessário entrar em seus detalhes operacionais. Desta forma, a metodologia se torna independente de ferramentas específicas.

4.1.5 Fase V: Consolidação e Análise

Nas fases anteriores são gerados documentos tais como: relatórios das varreduras, planos de ataque, resultados dos ataques, análises destes resultados e *log* das atividades. As informações contidas nestes artefatos precisam ser consolidadas para a análise e posterior apresentação dos resultados obtidos, bem como dos procedimentos utilizados.

Além da consolidação e análise dos resultados, também é necessário decidir se os testes serão encerrados ou se uma nova rodada será executada, tanto para atingir objetivos que não foram alcançados como para aprofundar e explorar caminhos que não foram previstos no planejamento.

4.1.6 Fase VI: Apresentação dos Resultados

A partir da consolidação dos resultados e sua análise, os achados e conclusões do projeto são apresentados ao demandante conforme as definições e os resultados obtidos.

Cada uma das fases acima compreende processos, que por sua vez agrupam atividades. Estes são detalhados na Seção seguinte.

4.2 Detalhamento das Fases

Cada uma das fases acima compreende processos, que por sua vez agrupam atividades gerando artefatos. Estes são detalhados na sequência. Contudo, para mera referência futura, ao fim deste capítulo (Seção 6.1) segue um quadro de resumo das fases, processos e atividades desta metodologia.

As fases são detalhadas a seguir.

4.2.1 Fase I: Especificação dos Testes de Penetração

Esta fase tem por finalidade definir os objetivos e escopo do teste de penetração e suas condições de execução.

Em termos de procedimento, podem ser realizadas reuniões, entrevistas ou formulários, cujos resultados devem ser consolidados. Devem estar envolvidos nesta fase representantes do executor, a equipe de testes, e da instituição demandante, responsável pelos sistemas de informação a serem testados. A ênfase não deve ser nos aspectos técnicos dos testes, mas no esclarecimento da função estratégica dos sistemas a serem testados no contexto do negócio da instituição. Assim, estes encontros devem incluir não só pessoal técnico mas também de nível gerencial.

Nesta fase podem ser apresentadas queixas específicas que deverão ser tratadas. Por exemplo, pode ser de interesse da organização em que serão executados os testes de penetração, averiguar especificamente um de seus sistemas de informação do qual já pode haver dúvidas quanto ao estado de sua segurança; ou checar se um dado sistema crítico está realmente seguro.

A ideia é que, de posse destas informações se definam o escopo e a profundidade com que ferramentas e técnicas serão lançadas sobre os sistemas alvo. Estas informações darão subsídio para a elaboração da especificação dos testes de penetração. No caso de haver implicações jurídicas ou regulatórias envolvendo a realização dos testes, recomenda-se também a elaboração de um termo de acordo entre a organização em que os testes de penetração serão realizados e os executores fixando os arcabouço legal que regerá a execução dos testes.

Ao final desta fase, a equipe de testes deve estar com as seguintes questões respondidas:

1. O teste de penetração será interno ou externo?
   Do ponto de vista da localização física em que estará a equipe que realizará os testes de penetração, ela poderá se situar no ambiente da organização, com acesso à sua rede interna corporativa, ou acessando de fora, possivelmente via Internet. O posicionamento da equipe é determinante das formas de execução dos testes: incluindo as técnicas e ferramentas a serem empregadas.

2. Sob que aspecto o teste de penetração será conduzido: confidencialidade, integridade, disponibilidade, autenticidade, etc?
   A segurança da informação visa atender requisitos de segurança, como: confidencialidade, integridade, disponibilidade, autenticidade, etc. Entretanto, estes requisitos são atendidos de acordo com o foco de atuação da organização. Por exemplo, em organizações militares tipicamente o requisito priorizado é a confidencialidade, enquanto que em um banco a prioridade seria a integridade. Assim no primeiro caso, um teste de penetração estaria focado em obter acesso indevido a informações, enquanto no último além do acesso indevido seria necessário demonstrar a possibilidade de alteração das informações.

   O aspecto de segurança a ser priorizado também é fator determinante dos testes a serem realizados.

3. Quais os *hosts*/servidores, e/ou os endereços que devem ser testados?
   Minimamente, é necessário saber a faixa de endereços de rede que precisam ser testados. Em alguns casos, pode-se ter especificamente uma lista de máquinas que a organização alvo considere prioritários ou já tenha alguma queixa ou suspeita, e neste caso estas máquinas terão um tratamento mais focado quando da realização dos testes de penetração.

4. Há algum serviço específico, crítico ou não, a ser testado?
   Pode haver algum serviço específico que a organização em que os testes serão realizados considere prioritários ou já tenha alguma queixa ou suspeita e assim o serviço terá um tratamento diferenciado quando da realização dos testes de penetração.

5. Há alguma aplicação específica, crítica ou não, a ser testada?
   Tem-se aqui uma situação análoga à da questão anterior, porém a priorização será dada a alguma aplicação ao invés de um serviço.

6. Há alguma informação específica que deve ser avaliada quanto a confidencialidade, integridade e controle de acesso?
   Pode haver alguma informação específica que a organização alvo considere prioritária e assim o serviço, aplicação ou máquinas a ela relacionada terão um tratamento específico quando da realização dos testes de penetração.

7. Deve-se avaliar a rede, serviços ou aplicações quanto à disponibilidade, i. e., robustez à negação de serviços?

A avaliação da disponibilidade normalmente acarreta interrrupção de serviços, o que normalmente não deve acontecer, especialmente durante testes de penetração. Entretanto, se explicitamente expresso pela organização em que os testes de penetração serão realizados, deve-se executá-los. Quando da definição deste aspecto, deve-se alertar quanto a possibilidade de interrupção de serviço.

8. Podem-se executar testes que levem à negação momentânea de serviços?
   Similarmente à situação da questão anterior, há testes de penetração que podem interromper serviços e aplicações, ainda que este não seja o objetivo. Quando esse for o caso, a organização demandante deve ser alertada. Normalmente, deve-se evitá-los. Entretanto, se for expressamente solicitado pela organização demandante, devem ser incluídos na suite de testes a ser realizados.

9. Os testes devem ser executados sem que sejam detectados?
   Um dos aspectos que podem ser avaliados durante testes de penetração é a capacidade de detecção e resposta da equipe responsável pela TI da organização alvo. Neste caso, não há a necessidade de se utilizar técnicas furtivas de teste. Entretanto, pode haver uma demanda específica do contrário. O requisito de execução dos testes de penetração utilizando técnicas furtivas é fator determinante na escolha destas e nas forma de abordar os sistemas que serão avaliados.

10. Qual o prazo de execução?
    O prazo de execução típico é de quatro a seis semanas, dependendo das informações obtidas, tamanho da rede e da equipe que realizará os testes de penetração (tipicamente de 2 a 4 componentes).

11. Quais os custos envolvidos (Recursos humanos, Financeiros, Disponibilidade, etc)?
    As questões anteriores darão subsídios para dimensionamento dos recursos necessários para realização dos testes. Por exemplo, testes internos podem implicar em custos com deslocamento e estadia da equipe.

Da lista acima, as questões que não podem deixar de ser respondidas são as três primeiras, que dizem respeito à forma e aos objetivos e escopo específicos dos testes de penetração, e as duas últimas, que tratam do dimensionamento dos recursos a serem empregados para a realização dos testes. Normalmente, as questões 4, 5 e 6 não têm resposta, a menos que a organização alvo já tenha alguma preocupação ou queixa específica em termos de segurança da informação

As questões 7 e 8 normalmente têm resposta negativa, ou seja, deve-se evitar a interrupção de serviços e aplicações, a menos que se deseje também avaliar a disponibilidade. Já a questão 9 terá resposta afirmativa quando se desejar avaliar a capacidade de detecção e resposta da instituição.

4.2.2 Atividades da Fase I

O objetivo da Fase I é produzir a especificação detalhada dos testes de penetração. Esta especificação constitui documento básico do projeto em que serão realizados os testes de penetração. A Figura 4.2 mostra as atividades da Fase I.

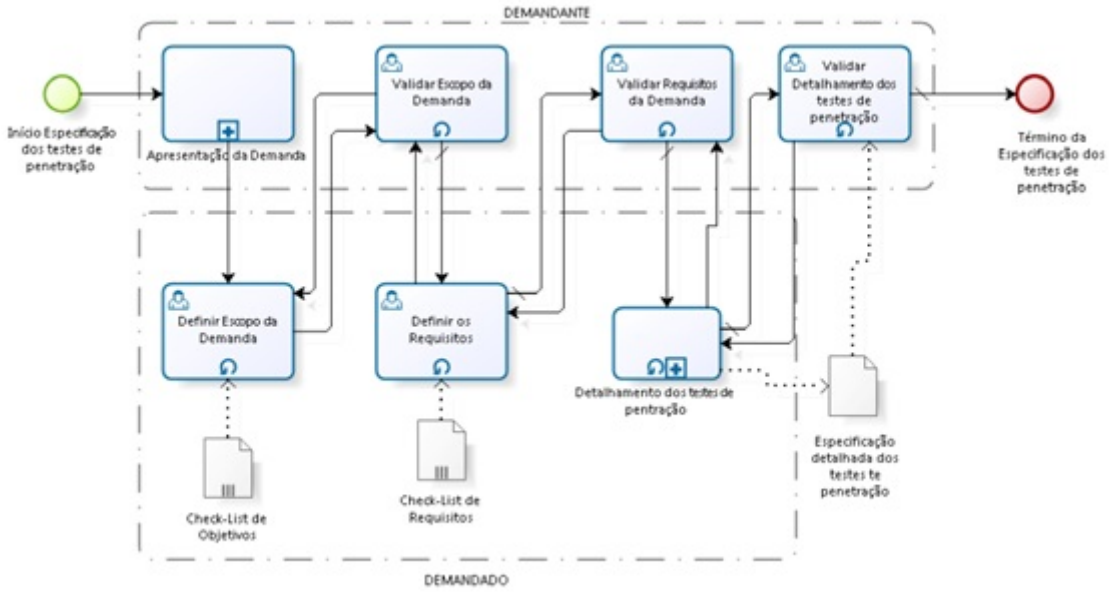

Figura 4.2: Fase 1 - Especificação dos Testes.

As atividades da Fase I se desenvolvem de forma interativa entre o demandante e o demandado sendo descritas a seguir:

- Apresentação da Demanda

  A apresentação da demanda, que pode tomar a forma de uma reunião inicial, é a primeira atividade deste processo e visa estabelecer as bases para a especificação dos testes de penetração. Alternativamente, a reunião pode ser substituída pelo envio de um questionário a ser respondido pelo demandante.

  A apresentação inicial visa fornecer uma primeira aproximação das informações que venham a responder a lista de questões apresentada anteriormente. A menos que estejam presentes membros da equipe técnica da organização alvo, estes aspectos não devem ser abordados explicitamente, devendo se evitar a solicitação de documentos, principalmente técnicos. Porém, se espontaneamente oferecidos, podem ser aceitos. É fundamental, no decorrer da reunião, passar aos representantes da organização a imagem de confiança e cooperação. Deve-se indagar aos representantes da organização da conveniência de elaborar, como um adendo à especificação, um termo de acordo, esclarecendo que a primeira será o documento formal definindo os aspectos técnicos, enquanto o último se atém aos aspectos administrativos e jurídicos.

- Definir Escopo da Demanda

  Após a apresentação da demanda, é preciso definir o escopo dos testes. Das questões sugeridas acima, juntamente com as respostas e demais informações passadas, deve-se extrair uma lista de objetivos e o escopo dos testes. Estes devem refletir as necessidades e demandas apresentadas segundo os critérios de segurança da informação.

  Para subsidiar a elaboração dos objetivos e do escopo, deve-se obter informações sobre a organização demandante. O objetivo principal é consolidar e detalhar as informações coletadas na apresentação. Esta atividade por sua vez também pode subsidiar a elaboração do termo de acordo para a realização dos testes, se necessário.

- Validar o Escopo da Demanda

  A lista de objetivos e o escopo devem ser validados pelo demandante. O demandante deve avaliar se suas necessidades estão devidamente capturadas na lista de objetivos e o escopo submetidos para apreciação. A consolidação destes será a base para a definição dos requisitos.

- Definir os Requisitos

  Uma vez validado o escopo e os objetivos, deve-se elaborar uma lista de requisitos. Estes ainda devem ser formulados de forma geral, sem necessariamente mapear as características específicas do demandante. Um passo que deve ser realizado é a consolidação das informações da listas de verificação resultantes das atividades anteriores.

  A lista de verificação básica para a reunião incluiu as onze questões já levantadas anteriormente. Ela deve ser cruzada com as informações obtidas no levantamento de informações realizado anteriormente para incluir aspectos específicos e relevantes da organização.

  Esta consolidação levará à lista de requisitos que será submetida para validação.

- Validar os Requisitos

  O requisitos definidos devem ser validados pelo demandante, onde este avalia a adequação do conjunto de requisitos, ainda não detalhados, para execução dos testes.

- Detalhamento dos testes de penetração

  Os requisitos validados são então detalhados. Esse detalhamento produzirá o documento base com a especificação detalhada dos testes de penetração, que deve ser também validado pelo demandante.

  A especificação detalhada é a peça fundamental para o início dos testes em si devendo ser elaborada segundo as informações obtidas. Nela estão os parâmetros que definirão os aspectos técnicos dos testes. Nela deve necessariamente estar explícito o que vai ser realizado, sob que aspecto vai ser testado, o escopo dos testes, suas condições limitadoras e de realização, bem como o que não vai, não deve e/ou não pode ser feito, como por exemplo, interrupção de serviços, alteração de informação, etc.

O termo de acordo, caso se julgue necessário elaborá-lo, é um adendo à especificação dos testes sempre que houver requisitos legais e regulatórios incorrendo sobre a realização dos mesmos e será o balizador jurídico e administrativo de toda sua execução. Seu principal objetivo é garantir o conhecimento e o consentimento explícito da organização acerca de como os testes de penetração serão executados e conduzidos.

Caso se opte por se elaborar um termo de acordo, este deve ser elaborado sob duas perspectivas:

- a administrativa: listando, para cada uma das partes, os recursos que deverão ser providos ou ter o acesso concedido, em que condições, cronograma, ônus de cada parte, etc; e
- a jurídica: que deve garantir que tudo que foi previsto dos pontos de vista técnico e administrativo está dentro da legalidade.

- Validar Detalhamento dos testes de Penetração

  O demandante deverá avaliar o detalhamento de forma global, levando em conta recursos, objetivos, escopo, prazos e condições de execução.

  Completada esta atividade, encerra-se esta fase.

- Detalhamento das Atividades e Artefatos da Fase I

  - Início Especificação do Projeto
    Início geral da especificação do projeto. Normalmente o início é em função de algum problema ou solução necessária de segurança da informação em alguma infraestrutura ou sistema.
  - Apresentação da Demanda
    O demandante apresenta suas necessidades acerca de informações que gostaria de obter e possíveis fontes de onde a informação provavelmente estaria disponível. O demandado interage com o demandante até o ponto de entendimento global da necessidade e quais os requisitos e objetivos mínimos.
  - Definir Escopo da Demanda
    Uma vez que se tenha o conjunto de requisitos, o demandado gera a lista de objetivos do projeto. Os objetivos do projeto devem ser focados nas necessidades e demandas apresentadas segundo critérios de segurança da informação.
  - Check-List de Objetivos
    Documento que subsidia o demandado a gerar a lista de objetivos do projeto. Este documento contém informações como:
    * Dados a serem exfiltrados.
    * Estabelecimento de alvos a serem comandados e controlados.
    * Dados, infraestrutura e serviços a serem monitorados.

    * Dados, informações e infraestrutura a serem destruídos ou incapacitados.
    * Interrupção de serviços e nível de aceitação da interrupção.
    * Definição do nível de dano aceitável em caso de destruição de dados e infraestrutura.

– Validar Escopo da Demanda
  O demandante deve validar os objetivos do projeto segundo suas necessidades e o entendimento global da demanda.

– Definir os Requisitos
  O demandado define um conjunto de requisitos em função do entendimento da demanda. O demandado analisa e detalha os requisitos em termos de ações, recursos e condições gerais de execução. Ao final da tarefa o demandado deve ter o conjunto de requisitos definido.

– Check-List de Requisitos
  Documento auxiliar que contém conjunto de informações que subsidiam o demandado a detalhar os requisitos do projeto. Fazem parte dos requisitos informações como:
    * Descobrir e avaliar capacidade de detecção e resposta.
    * Origem do teste interno ou externo ao alvo.
    * Se o teste é focado em infraestrutura ou aplicação.
    * Se pode causar interrupção ou não de serviços.
    * Se pode destruir dados ou não.
    * Definição de recursos e custos envolvidos no projeto.
    * Definição de prazos e janelas de execução.
    * Definição de protocolo de troca de informação.
    * Definição do foco de segurança da informação relacionado à confidencialidade, disponibilidade, integridade, nãorrepúdio, autenticidade, etc.

– Validar Requisitos da Demanda
  O demandante deve validar o conjunto de requisitos da execução do projeto mediante aprovação de minuta de requisitos.

– Detalhamento do Projeto
  O demandado mediante o entendimento geral do projeto, do escopo e dos objetivos cria o documento com a especificação detalhada da demanda.

– Validar Detalhamento do Projeto
  O demandante deve validar o projeto de maneira global, incluindo seus recursos, objetivos, escopo, prazos e condições de execução.

– Especificação Detalhada do Projeto
  Documento Executivo que contém objetivos, escopo, recursos, prazos, condições de execução relacionados ao projeto segundo entendimento do demandante e do demandado.

– Término da Especificação do Projeto
  Conclusão da especificação do projeto. O processo é finalizado com a criação de um artefato detalhado do projeto contendo a especificação detalhada do projeto.

4.2.3 Fase II: Levantamento Inicial

O objetivo desta fase é realizar o reconhecimento e levantamento dos sistemas de informação a serem testados, bem como suas possíveis vulnerabilidades. Os resultados desta fase servirão como subsídio para a fase seguinte em que serão determinados os alvos.

Esta fase deve ser realizada utilizando ferramentas de rede capazes de efetuar varreduras, testes e diagnósticos. Também serão utilizadas ferramentas potencialmente agressivas que investigam vulnerabilidades específicas de sistemas operacionais, serviços e aplicações.

As varreduras devem ser executadas seguindo critérios, conforme uma escala crescente de profundidade, grau de intrusão e detalhamento. Estes critérios são apresentados na Tabela 4.1.

Tabela 4.1: Critérios de Aplicação de Varredura

Critério	Comentário
Profundidade	O projeto dos sistemas e redes e as medidas adotadas para a sua proteção seguem uma arquitetura em camadas, ficando as informações mais sensíveis nas camadas mais internas. Assim, a profundidade diz respeito à camada que é alvo da varredura.
Grau de intrusão	As varreduras são baseadas em estímulos e respostas. Pode-se ter estímulos que causam mais efeitos que outros, e consequentemente são mais fáceis de ser detectados. Assim, o grau de intrusão diz respeito aos possíveis efeitos causados pela varredura.
Detalhamento	As varreduras produzem um volume razoável de informação. O detalhamento diz respeito à quantidade e especificidade das informações obtidas em uma varredura.

Assim, pode-se operacionalizar as varreduras em três níveis, ilustrados na Tabela 4.2.

Deve-se ressaltar que as varreduras de aplicação normalmente não são suportadas por ferramentas específicas exigindo um trabalho metódico e minucioso de examinar cada link das páginas web disponibilizadas, bem como o código fonte das mesmas. Consequentemente, as varreduras de aplicação são as que começam a exigir uma formação mais elaborada de quem vai executá-las, enquanto que nas anteriores basta disparar as ferramentas que mecanizam as varreduras. A formação do executor exige conhecimento acerca da implementação de aplicações web, bancos de dados e segurança de redes.

Tabela 4.2: Níveis de Varredura

Nível	Tipo	Descrição
1	Varreduras de Rede	Estas varreduras têm por foco a topologia e a infraestrutura da rede (roteadores, enlaces, servidores, etc), dos sistemas operacionais e dos serviços nela presentes. No caso dos serviços, devem-se também identificar os software que os implementam. Outros objetivos incluem detectar presença ou não de dispositivos como *firewall* ou *proxy*, identificando fabricante, versão e, se possível, política de filtragem.
2	Varreduras de Serviços	O foco destas varreduras está na identificação de *daemons*, da versão que está sendo executada e de possíveis vulnerabilidades. Uma vez identificados os serviços, em alguns casos, como o HTTP, de acordo com o software que os implementam podem sofrer varreduras focalizadas em vulnerabilidades específicas deste,utilizando ferramentas específicas.
3	Varreduras de Aplicação	Esta varredura busca por vulnerabilidades de aplicação, no caso as que implementam ou suportam sistemas de informação. Por exemplo, é comum uma rede disponibilizar aplicações em ambiente web, tais como serviços de busca, controle de acesso a informação, consultas a bancos de dados corporativos ou simplesmente páginas de conteúdo dinâmico. Todas as aplicações desta categoria devem ser testadas quanto às vulnerabilidades mais frequentes e conhecidas.

O fluxo geral da Fase II é mostrado na Figura 4.3.

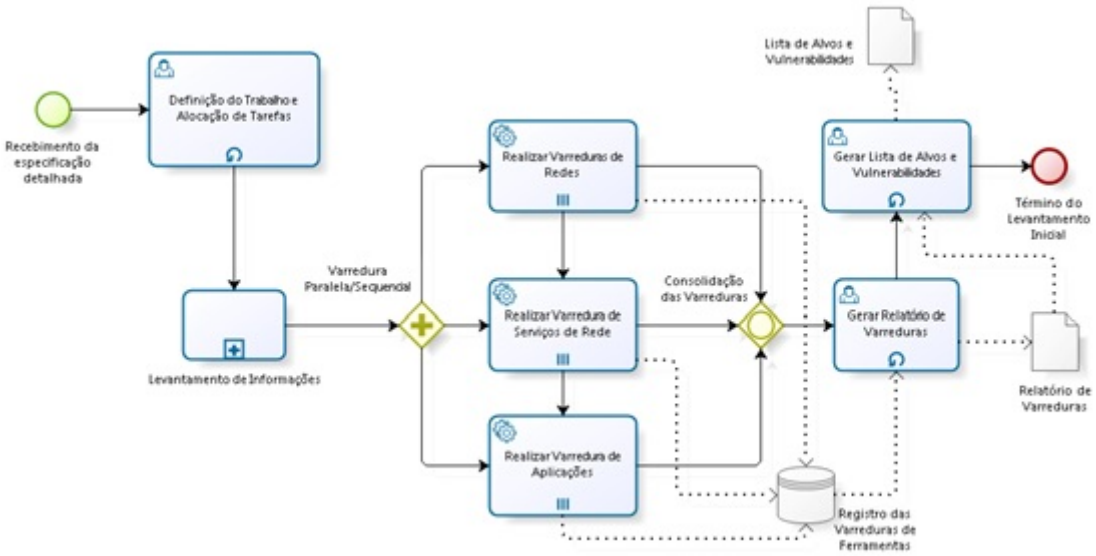

Figura 4.3: Fase 2 - Varreduras Iniciais.

4.2.4 Atividades da Fase II

A Figura 4.3 ilustra as possíveis sequências de varredura, que podem se repetir ocorrendo em nível de rede, vulnerabilidades de sistemas, serviços ou aplicações. As macro atividades da Fase II são descritas a seguir.

- Definição do Trabalho e Alocação de Tarefas

  A primeira atividade desta fase é a definição do trabalho e a alocação de tarefas. O subsídio básico é a especificação detalhada. Desta tem-se como extrair as tarefas. Por exemplo, se os testes têm por escopo a infraestrutura, excluindo aplicações, as varreduras estarão restritas à rede e serviços. A alocação de tarefas segue as competências dos membros da equipe de execução.

- Levantamento de Informações

  Esta atividade envolve o levantamento de informações sobre a organização em que se realizarão os testes. O objetivo do levantamento é propiciar um melhor entendimento da organização que terá os sistemas testados. Especificamente, devem-se ter os subsídios para realização das varreduras e posteriormente na seleção de alvos.

  O levantamento deve se focar em:

  – dados institucionais, principalmente a missão e o ramo de atividade da organização;

  – imagem da organização, como ela se apresenta ao público;

  – abrangência;

- dados financeiros.

- Realizar Varreduras de Rede

  As varreduras de rede têm por finalidade identificar os elementos da topologia da rede. Normalmente, esta tarefa é suportada integralmente por ferramentas. Eventualmente, para redes que utilizam serviços como *Network Address Translation* (NAT) técnicas adicionais podem vir a ser necessárias.

  As ações desta atividade são:

  - Preparação de varredura de rede:
    A varredura de rede é preparada tendo como subsídio o relatório da reunião já realizada, especificamente as informações constantes da lista de verificação respondida. Estas informações servirão de base para definir as ferramentas que serão utilizadas, bem como seu *modus operandi*. Por exemplo, se foi identificada a necessidade dos testes não serem detectados, as varreduras deverão ser realizadas por ferramentas que suportem operação em modo furtivo (*stealth*).

    Espera-se que dentre as informações obtidas estejam nomes de domínio e servidores. Minimamente, deve-se ter pelo menos uma faixa de endereços IP. As ferramentas tipicamente utilizadas nesta fase são conhecidas genericamente como *port scanners* (*e.g.* NMAP). No caso de redes com NAT, a identificação pode necessitar de recursos mais sofisticados, como o controle de parâmetros específicos dos estímulos utilizados na varredura. No caso de se necessitar controlar os parâmetros dos campos dos datagramas que implementam os estímulos (*probes*), certamente será necessário desenvolver uma aplicação que gere tais *probes* ou uma ferramenta capaz de forjar pacotes (*e.g.* *scapy*). Ao final desta atividade deve-se ter um roteiro das varreduras de rede, que deve incluir:

    * faixa de endereços a serem varridos;
    * técnicas de varredura a serem utilizadas;
    * ferramentas a serem utilizadas, juntamente com suas configurações;
    * sequência de ativação das ferramentas.

  - Execução de varredura de rede:
    Esta atividade compreende a execução da varredura em si, segundo o roteiro determinado anteriormente. Eventualmente, pode ser necessário rever o roteiro proposto à medida que os resultados são obtidos. Ao final desta ação estarão disponíveis os relatórios gerados pelas ferramentas utilizadas.

  - Compilação de resultados de varredura de rede:
    A compilação dos resultados compreende a análise, correlação e consolidação dos resultados obtidos na fase anterior. Os resultados são cruzados para identificação de inconsistências e corroborações. Eventualmente, resultados discrepantes podem levar a repetição de testes. Ao final desta ação deve-se ter uma tabula-

ção de máquinas identificadas, e os respectivos serviços identificados. Possivelmente, pode-se ter também a definição dos sistemas operacionais das máquinas identificadas.

- Realizar Varreduras de Serviços

  As varreduras de serviços aprofundam o processo de levantamento de informações em preparação aos testes de penetração propriamente ditos. Neste estágio, já se buscam informações sobre as funcionalidades disponíveis nas máquinas e sistemas identificados, focando na identificação de vulnerabilidades dos sistemas em execução nas máquinas identificadas.

  Estas varreduras ocorrem em duas etapas. A primeira tem por objetivo identificar os componentes de software (sistemas operacionais e servidores) que estão sendo executados nos *hosts* identificados. De posse dessas informações, os componentes são testados especificamente para as versões dos componentes que estão instalados e em execução.

  As atividades que compõem este processo são descritas a seguir. Para a identificação dos componentes de software, temos:

  - Preparação de varredura de serviços:
    As tabulações geradas nas varreduras anteriores fornecem os subsídios para a preparação da varredura de serviços. As informações sobre máquinas e sistemas identificados serão cruzadas com os dados de serviços de nomes (DNS, *Whois*). Além disso, serão realizadas varreduras sobre serviços para os quais existam ferramentas específicas de varredura. Ao final desta atividade deve-se ter um roteiro com:
    * os serviços específicos a serem varridos com os respectivos endereços;
    * técnicas de varredura a serem utilizadas;
    * ferramentas a serem utilizadas, juntamente com suas configurações;
    * sequência de ativação das ferramentas.
  - Execução de varredura de serviços:
    Similarmente às varreduras anteriores, esta ação compreende a execução da varredura de serviço em si, segundo o roteiro determinado anteriormente. Eventualmente pode ser necessário rever o roteiro proposto à medida que os resultados são obtidos. Ao final desta atividade estarão disponíveis os relatórios de saída das ferramentas utilizadas.
  - Compilação de resultados de varredura de serviços:
    Nesta atividade os relatórios de saída das ferramentas utilizadas são compilados e tabulados com as vulnerabilidades específicas dos serviços identificados.

  Na sequência, as varreduras são especificamente voltadas às possíveis vulnerabilidades das versões identificadas dos componentes em execução. Suas atividades são:

- Preparação de varredura de vulnerabilidades:
  A compilação de resultados vinda da varredura de rede será a base na qual as ferramentas, no caso conhecidas como *probe scanners*, serão selecionadas e configuradas. Tipicamente, um *probe scanner* realiza testes específicos averiguando a ocorrência de vulnerabilidades. Em alguns casos os testes podem ser disruptivos, de modo que é preciso se avaliar possíveis impactos antes da execução.

  Ao final desta ação deve-se ter um roteiro com:

  * os sistemas e serviços a serem varridos com os respectivos endereços;
  * técnicas de varredura a serem utilizadas;
  * ferramentas a serem utilizadas, juntamente com suas configurações;
  * sequência de ativação das ferramentas.

- Execução de varredura de vulnerabilidades:
  Identicamente à varredura de serviços, esta ação compreende a execução da varredura de vulnerabilidades em si, segundo o roteiro determinado anteriormente. Eventualmente pode ser necessário rever o roteiro proposto à medida que os resultados são obtidos. Ao final desta atividade estarão disponíveis os relatórios de saída das ferramentas utilizadas.

- Compilação de resultados de varredura de vulnerabilidades:
  Nesta atividade os relatórios gerados pelas ferramentas utilizadas são compilados e tabulados incluindo as vulnerabilidades específicas dos serviços identificados.

- Realizar Varreduras de Aplicações

  As varreduras de aplicações são as que têm maior valor agregado, se comparada com as anteriores. Normalmente as aplicações lidam diretamente com as informações relevantes à missão e negócio das organizações, podendo inclusive implementar processos críticos. Aplicações típicas têm a forma de interfaces que servem para acessar sistemas de bancos de dados. Assim, o caminho para as informações fica mais direto e rápido. O que torna estas varreduras mais difíceis é que enquanto sistemas operacionais e serviços apresentam razoável padronização, no que diz respeito à arquitetura e implementação, o mesmo não acontece com as aplicações. A grande dispersão de arquiteturas e implementações dificulta a disponibilização de ferramentas de suporte aos testes levando à necessidade de técnicas mais sofisticadas.

  Por exemplo, uma aplicação que intermedia o acesso a um banco de dados corporativo pode ser testada com a técnica de *SQL injection*. Esta técnica, que consiste em injetar fragmentos de código SQL (*Simplified Query Language*: linguagem de acesso padrão para bancos de dados relacionais) visando burlar os mecanismos de controle de acesso. O sucesso desta técnica está em inferir a estrutura das consultas usada pela aplicação e a partir daí explorá-la. Esta técnica, que na sua operacionalização é um processo de tentativa, erro e inferência, dispõe de suporte de ferramentas relativamente limitado.

- Preparação de varredura de aplicações:
  As tabulações geradas nas varreduras anteriores dão os subsídios para a preparação da varredura de aplicações. Entretanto, uma vez identificadas, é preciso levantar informações sobre aplicação e de sua arquitetura e implementação para que se possa decidir como abordá-la.

  Assim, antes da determinação de que testes devem ser realizados, deve-se realizar varredura manual de reconhecimento para cada aplicação identificada. Por exemplo, para uma aplicação web deve-se acessar todos os links e todos os campos de dados disponíveis. Nestes últimos, deve-se entrar com valores que provoquem mensagens de erro, que podem revelar detalhes de implementação. Neste processo, deve-se desativar a facilidade provida por alguns navegadores em que mensagens de erro são tratadas e apresentadas em um formato padrão, supostamente amigável.

  Ao final desta atividade deve-se ter um roteiro com:

  * as aplicações específicas a serem testadas com os respectivos endereços;
  * técnicas de varredura a serem utilizadas com sua ordem de utilização;
  * ferramentas a serem utilizadas, se houver, juntamente com suas configurações e sequência de ativação das ferramentas.

- Execução de varredura de aplicações:
  Identicamente às varreduras já realizadas, esta atividade compreende a execução das varreduras de aplicações em si, segundo o roteiro determinado anteriormente. Eventualmente pode ser necessário rever o roteiro proposto à medida que os resultados são obtidos. Ao final desta ação estarão disponíveis os resultados dos testes realizados e se houver ferramentas disponíveis, seus relatórios de saída.

- Compilação de resultados de varredura de aplicações:
  Nesta atividade os resultados dos testes realizados e, se houver, os relatórios de saída das ferramentas utilizadas são compilados e tabulados com as vulnerabilidades específicas das aplicações identificadas.

- Gerar Relatório de Varreduras

  Esta atividade tem por finalidade produzir o relatório de varreduras. Essencialmente, este relatório é gerado agregando as compilações dos resultados das varreduras realizadas.

- Gerar Lista de Alvos e Vulnerabilidades

  Nesta atividade, extrai-se do relatório de varreduras gerado anteriormente as informações relevantes com respeito a vulnerabilidades e as máquinas em que estas foram detectadas. Assim, para cada máquina identificada com vulnerabilidades estas são listadas.

  Esta atividade encerra a fase.

- Detalhamento das Atividades e Artefatos da Fase II

  - Recebimento da especificação detalhada
    O demandado entrega ao especialista em segurança da informação o projeto com a especificação detalhada do projeto.
  - Definição do Trabalho e Alocação de Tarefas
    O especialista em segurança deve criar um conjunto de tarefas e dividir as funções entre os integrantes da equipe de *pentest* de acordo com a finalidade do teste e equipe disponível para execução. O especialista deve conhecer o perfil técnico da equipe. É importante a definição de um cronograma básico de execução.
  - Levantamento de Informações
    Esse processo é o responsável pelo levantamento de informações em fontes abertas sem realizar testes de varreduras ativas. Esta etapa é considerada como um reconhecimento do ambiente onde o escopo e o objetivo do *pentest* serão realizadas.
  - Varredura Paralela/Sequencial
    Em função do detalhamento das atividades, reconhecimento, objetivos e escopo, são realizados pelo menos três tipos diferentes de varreduras. São iniciadas de maneira superficial e vão se aprofundando à medida que se levanta dados. Devem ser considerados aspectos de furtividade, onde as operações devem ou não ser realizadas de maneira furtiva em função do objetivo e escopo do projeto.
  - Realizar Varreduras de Redes
    De posse do uso de ferramentas de varredura de rede e das informações acerca do escopo e objetivos do projeto, são realizados os processos específicos de varredura com uso de ferramentas e os seus resultados são guardados para análise. O objetivo principal da varredura de rede é levantar a topologia de rede do alvo, identificando elementos ativos, seus fornecedores e versões de sistemas em operação.
  - Realizar Varredura de Serviços de Rede
    Etapa onde ocorre a busca e identificação de serviços de rede utilizando ferramentas específicas de acordo com o escopo e objetivo do projeto. O objetivo desta atividade é levantar possíveis serviços de rede, softwares, sistemas, versões, fabricantes e em última análise as vulnerabilidades que possam existir referentes ao serviço em questão. O processo deve ser documentado no registro de varredura para consulta e análise na medida do necessário.
  - Realizar Varredura de Aplicações
    Etapa onde é realizado o detalhamento de aplicações e sua arquitetura, softwares, sistemas e versões com respectivos detalhes sobre possíveis vulnerabilidades, mediante uso de ferramentas específicas. A varredura de aplicações tem por objetivo o levantamento de softwares em camada de aplicação que podem ser usados

como vetores de outros métodos de análise de vulnerabilidade e escalação de privilégio. Toda a execução e resultados devem ser armazenados no registro de varreduras para posterior análise e consulta.

- Registro das Varreduras de Ferramentas
  Repositório com os resultados das varreduras e seu detalhamento específico. Este repositório deve ter uma linha de tempo para avaliar possíveis mudanças de objetivos, contexto e do próprio escopo do projeto.
- Consolidação das Varreduras
  Agregação dos resultados de cada técnica para consolidar o relatório de varredura.
- Gerar Relatório de Varreduras
  Etapa onde ocorre análise e correlação dos resultados das varreduras. Esta tarefa inclui a indicação de possíveis caminhos de exploração das vulnerabilidades identificadas. Gera como saída um relatório detalhado de varredura.
- Relatório de Varreduras
  Documento que contém:
  - a lista de varredura de redes e seu detalhamento.
  - a lista de serviços de redes e seu detalhamento.
  - a lista de aplicações e seu detalhamento.
- Gerar Lista de Alvos e Vulnerabilidades
  Etapa que usa como insumo o relatório de varreduras para criar a lista de alvos e vulnerabilidades e possíveis vetores de exploração. Gera o relatório de alvos e vulnerabilidades e suas respectivas oportunidades de exploração.
- Lista de Alvos e Vulnerabilidades
  Documento que contém a lista de alvos e vulnerabilidades consolidados mediante o resultado das tarefas de varredura de redes, serviços e aplicações. Este documento deve auxiliar o especialista de segurança a preparar a estratégia de ataques em função de se evitar falsos positivos e atingir o escopo e o objetivo do *pentest*.
- Término do Levantamento Inicial
  Fase concluída com a geração dos respectivos artefatos.

4.2.5 Fase III: Planejamento do *Pentest*

De posse das informações acerca das vulnerabilidades dos serviços e aplicações, esta fase visa definir a estratégia de abordagem dos sistemas de informação a serem testados, sendo seu principal resultado a escolha dos alvos segundo o critério de acesso à informação, a definição das táticas a serem utilizadas e a elaboração dos planos executivos de ataque.

Prioritariamente, têm-se como alvos os bancos de dados , os serviços de informação e comunicação corporativos (como correio eletrônico e intranet), ainda que o teste de penetração seja externo. Também são alvos os sistemas de proteção e controle de acesso aos alvos prioritários.

A execução das atividades desta fase envolve a aplicação de dois conjuntos de critérios. O primeiro conjunto diz respeito à classificação das vulnerabilidades e leva em consideração aspectos intrínsecos à vulnerabilidade em si, como potencial de impacto e facilidade de aplicação, junto com aspectos específicos do ambiente em que se desenrolam os testes. O outro conjunto de critérios diz respeito à ordenação das vulnerabilidades para escolha de alvos, levando em consideração os objetivos definidos na Fase I. Este cruzamento de alvos e vulnerabilidades define as táticas e os planos executivos de teste.

Estes dois conjuntos de critérios são descritos a seguir:

- Classificação das vulnerabilidades As vulnerabilidades encontradas são classificadas segundo critérios qualitativos. São levados em conta três fatores, onde cada um destes parâmetros pode assumir os valores A (alta), M (média) ou B (baixa):

  - Relevância:
    Diz respeito à importância da funcionalidade do *host* em que a vulnerabilidade foi encontrada. A Tabela 4.3 ilustra as definições dos níveis de relevância.

Tabela 4.3: Classificação de Relevância

Relevância	Definição
Baixa	*Hosts* que não disponibilizam sistemas, serviços ou aplicações associados à atividade núcleo da organização. O comprometimento destes *hosts* teria efeitos circunscritos a elas mesmas com pequena propagação.
Média	*Hosts* que disponibilizam sistemas, serviços ou aplicações associados à atividade núcleo da organização, porém não críticos. O comprometimento destes *hosts* teria efeitos além delas mesmas com média propagação. Nesta classificação devem ser incluídos os equipamentos de infraestrutura e proteção de rede.
Alta	*Hosts* que disponibilizam sistemas, serviços ou aplicações críticos associados à atividade núcleo da organização. O comprometimento destes *hosts* teria efeitos além delas mesmas com ampla propagação. Nesta classificação devem ser incluídos os servidores corporativos, em particular bancos de dados.

  - Severidade:
    Diz respeito à gravidade do comprometimento possível dada a exploração da vulnerabilidade. A Tabela 4.4 ilustra as definições dos níveis de severidade.

Tabela 4.4: Classificação de Severidade

Severidade	Definição
Baixa	O comprometimento decorrente da exploração da vulnerabilidade teria efeitos limitados, propiciando acesso, ou controle de funcionalidades restritas e de pequeno impacto. Exemplo: acesso não autorizado, não privilegiado, a recursos não críticos do *host* comprometido.
Média	O comprometimento decorrente da exploração da vulnerabilidade teria efeitos limitados, propiciando acesso, ou controle de funcionalidades importantes e de razoável impacto. Exemplo: acesso não autorizado, não privilegiado, a recursos críticos do *host* comprometido.
Alta	O comprometimento decorrente da exploração da vulnerabilidade teria efeitos globais, propiciando acesso, ou controle de funcionalidades importantes e de grande impacto. Exemplo: acesso não autorizado, privilegiado, a recursos críticos do *host* comprometido.

- Facilidade:
  Diz respeito ao grau de dificuldade de se explorar a vulnerabilidade, incluindo o nível técnico necessário para êxito na exploração. A Tabela 4.5 ilustra as definições dos níveis de facilidade.

Tabela 4.5: Classificação de Facilidade

Facilidade	Definição
Baixa	A vulnerabilidade não dispõe de ferramentas que suportam sua exploração, demandando alto nível técnico do agente que operacionalizará a exploração.
Média	A vulnerabilidade dispõe de ferramentas que suportam parcialmente sua exploração, demandando nível técnico médio do agente que operacionalizará a exploração.
Alta	A vulnerabilidade dispõe de ferramentas que suportam totalmente sua exploração, demandando baixo nível técnico do agente que operacionalizará a exploração.

Assim, para cada vulnerabilidade encontrada designa-se uma trinca de valores, cada um correspondente a um dos parâmetros acima. Esta gradação das vulnerabilidades é realizada conforme descrito a seguir.

Para cada vulnerabilidade encontrada na Fase II devem-se atribuir os valores correspondentes aos parâmetros acima. Nesta atividade deve-se montar, para cada *host* encontrado, uma tabela as seguintes colunas: vulnerabilidade, relevância, severidade e facilidade. Um exemplo de como seria a tabela é ilustrado a seguir na Tabela 4.6.

Tabela 4.6: Vulnerabilidades do *host* H1

vulnerabilidade	relevância	severidade	facilidade
CVE-XXX	Alta	Média	Baixa
CVE-YYY	Média	Média	Média
CVE-WWW	Baixa	Alta	Alta
CVE-ZZZ	Alta	Média	Alta

- Classificação e ordenação de alvos

  As linhas das tabelas geradas no processo anterior precisam ser ordenadas segundo os parâmetros descritos acima. A ordem das vulnerabilidades segue o critério:

  Para uma vulnerabilidade V que ocorre em um *host* H, seu escore é a trinca de valores $E = (R, S, F)$, associados aos parâmetros de gradação, onde R, S e F podem assumir valores $A > M > B$. Para um *host* H, e duas de suas vulnerabilidades V_i e V_j, temos que V_i é mais grave que V_j se $E_i >> E_j$, onde:

  $$E_i >> E_j \text{ se } R_i > R_j;$$
  $$\text{caso } R_i = R_j, \text{ se } S_i > S_j; \text{ e}$$
  $$\text{caso } S_i = S_j, \text{ se } F_i = F_j.$$

  Se os três valores coincidem $E_i == E_j$, e diz-se que os escores têm mesma gravidade.

A consolidação da seleção de alvos consiste em ordenar as linhas das tabelas construídas no processo anterior. Os alvos selecionados serão os *hosts* que exibirem o maior número de vulnerabilidades de maior gravidade. Ao final desta atividade devem-se obter as tabelas ordenadas e a lista dos *hosts* segundo a ordenação.

Na sequência pode-se, então, definir o planejamento tático dos testes de penetração. De posse dos possíveis alvos, já priorizados, os preferenciais são definidos, sendo montados os planos de ataque. Nestes planos deve-se incluir a escolha específica das táticas bem como a forma de execução.

Nesta fase pode haver a necessidade de desenvolver ferramentas específicas para execução dos ataques, principalmente se não há exploits disponíveis. Por exemplo, na fase II identificou-se um serviço para o qual não há ferramenta de exploração específica, porém o serviço tem seu código fonte aberto sendo possível acessá-lo. Quando submetido a uma ferramenta de auditoria de código, são encontradas vulnerabilidades, sendo uma delas escolhida para exploração. Essa escolha deve ser guiada pela eficácia da exploração com relação aos objetivos do teste e pela facilidade de implementação do código de prova de conceito do exploit.

Como princípio, devem-se priorizar ataques a aplicações, em particular aplicações web, sobre o uso de exploits de serviços ou sistemas operacionais. Este princípio se justifica pelas seguintes razões:

a) normalmente as aplicações estão mais próximas dos bancos de dados, que nelas confiam; assim o esforço de obtenção da informação tende a ser menor;

b) exploits são úteis na escalação de privilégios, mas se mal utilizados podem levar a negação de serviço;

c) a princípio, para quem defende, exploits são mais fáceis de detectar e prevenir, possivelmente diminuindo a eficiência dos ataques.

Deve-se, entretanto, lembrar que os exploits são mais facilmente mecanizáveis que ataques a aplicações. Caso seja requisito que os ataques, ou pelo menos suas origens, não sejam identificados, as táticas devem incluir provisões (*e.g.* como uso de encadeamento de *open proxy* ou mesmo *hot spots*), para anonimizar a origem do ataque.

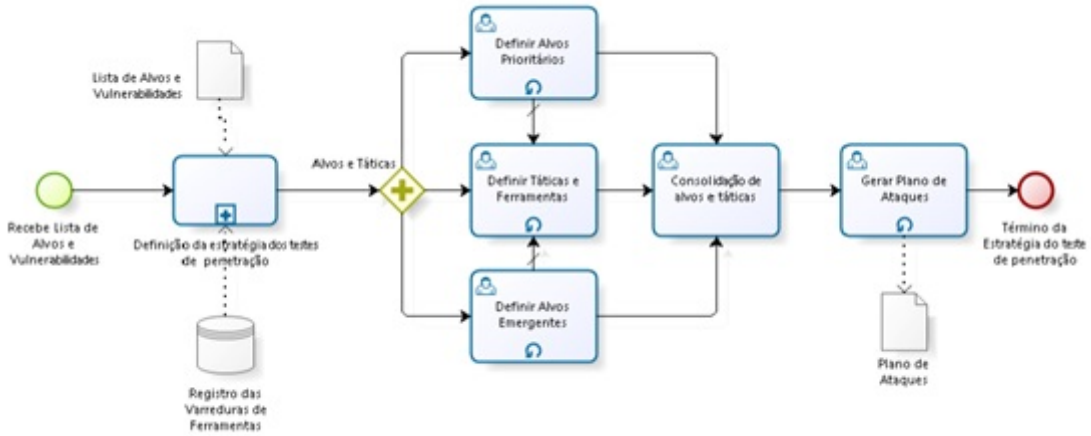

Figura 4.4: Fase 3 - Planejamento do *Pentest*.

4.2.6 Atividades da Fase III

A Figura 4.4 ilustra as atividades desta fase, que são detalhadas na sequência.

- Definição da Estratégia do Teste de Penetração

  Nesta atividade, os possíveis alvos e vulnerabilidades são classificados e ordenados. Além dos critérios definidos anteriormente para classificação e ordenação, deve-se avaliar a importância de cada alvo à luz das informações coletadas na Fase II e os objetivos e requisitos na especificação detalhada. Por exemplo, se foi estabelecido que o banco de dados de clientes é um ativo crítico e consta dos objetivos dos testes, o *host* que o hospeda deve ser alvo preferencial e assim uma estratégia para atingir o objetivo deve ser formulada.

  A seleção dos alvos preferenciais é essencialmente um processo de escolha e, possivelmente, desempate dos alvos ordenados na fase anterior. Enquanto na fase anterior a

classificação seguia padrões pré-estabelecidos, aqui o fator determinante será o expertise do agente executor dos testes. Este expertise reflete a competência, a experiência e o *feeling* do agente executor.

As ações aqui desempenhadas são:

- Análise detalhada dos alvos ordenados:
  Na fase anterior foi gerada uma lista ordenada dos alvos possíveis, com base nas informações providas pelas varreduras realizadas na Fase II e os critérios da Fase III. Aqui os critérios a serem utilizados são referentes à competência e experiência de quem irá executar os testes. Desta forma, tem-se a chance de projetar testes dentro da capacidade de execução disponível. Alguns pontos que devem ser levados em consideração na escolha dos alvos:
  - Disponibilidade de ferramentas mecanizadas para a realização dos testes:
    Se houver tais ferramentas disponíveis, estas devem ser adotadas pois a execução fica bem mais facilitada.
  - Alvos complexos:
    Pode-se ter uma situação em que, para um alvo que certamente seria preferencial, não há uma ferramenta disponível; tal situação pode decorrer da vulnerabilidade ser explorável por uma técnica complexa, ou pela situação particular do alvo. Deve-se considerar a possibilidade de se implementar as ferramentas que suportem o teste. A ferramenta de suporte não só facilita o teste, mas também possibilita que ele se torne repetível, o que leva a uma maior confiança nos resultados. Entretanto, deve-se avaliar a capacidade de desenvolvimento da equipe, e o tempo envolvido necessário para implementação, particularmente os testes. Deve-se lembrar que o desenvolvimento de ferramentas leva ao incremento da competência da equipe, sendo desejável.
  - Alvos emergentes:
    Não será raro acontecer de ser necessário testar um alvo de menor importância para se poder testar um alvo preferencial. Em tal situação, deve-se avaliar o esforço de testar o alvo emergente além do alvo preferencial.

Ao fim desta atividade será produzida a lista de alvos escolhidos. Os alvos serão listados juntamente com as vulnerabilidades que serão exploradas e uma descrição das táticas que serão utilizadas na exploração.

- Definir Alvos Prioritários

  Os alvos prioritários são aqueles cuja importância é justificada pelos objetivos e pelas informações levantadas devendo necessariamente ser avaliados pelo teste de penetração. Eles são determinados pelo cruzamento das informações da atividade anterior com os objetivos dos testes.

- Definir Alvos Emergentes

Os alvos emergentes são alvos de menor importância, com relação aos objetivos, ou barreiras de proteção, que uma vez explorados facilitam o acesso a um alvo prioritário. Estes alvos são identificados cruzando as informações de varreduras de redes e serviços com os objetivos dos testes.

- Definir Táticas e Ferramentas

  Definidos os alvos, é preciso para cada um deles escolher as técnicsa e as ferramentas, se existirem, que serão utilizadas para execução dos testes. Pode ser o caso de ser necessário implementar uma ferramenta específica para a dar suporte à execução de um teste. Deve-se avaliar o esforço de implementação face a importância do alvo, bem como os prazos envolvidos.

- Consolidação de Alvos e Táticas

  As definições específicas dos alvos, prioritários e emergentes, juntamente com as táticas e ferramentas identificadas, são cruzadas entre si já visando definir a sequência de condução dos testes e também detectar possíveis inconsistências e testes críticos. Tais situações deverão ser contornadas nos planos de ataque.

- Gerar Plano de Ataques

  Esta atividade é o estágio em que se realizam todos os preparativos anteriores à execução dos testes que explorarão as vulnerabilidades dos alvos escolhidos. A finalidade é antever os detalhes, opções, decisões, dificuldades e possíveis incidentes e contingências que venham a ocorrer durante os testes.

  Suas ações são:

  - Montagem dos planos:
    Para cada alvo escolhido, deve-se montar um plano de teste, que deve contemplar pelo menos os seguintes aspectos:
    - objetivo:
      o que se deseja alcançar com o teste;
    - alvo:
      os *hosts* e vulnerabilidades envolvidas;
    - técnicas a serem utilizadas;
    - ferramentas a serem usadas ou desenvolvidas, com configuração;
    - condução dos testes: descrição dos passos a serem executados; para cada passo deve-se ter o cenário, a ação a ser executada e o resultado esperado.
  - Preparação de ferramentas:
    Esta atividade abrange a simulação dos planos de teste, sempre que possível. Certamente, pode haver a necessidade de rever os planos elaborados na atividade anterior. No caso de uma ferramenta que precise ser desenvolvida, esta atividade

tem caráter crítico, já será nela que se dará o desenvolvimento e os testes da ferramenta.

Com a geração dos planos de ataque encerra-se esta Fase.

- Detalhamento das Atividades e Artefatos da Fase III

  - Recebe Lista de Alvos e Vulnerabilidades
    O processo é iniciado mediante o recebimento da lista de alvos e vulnerabilidades.
  - Lista de Alvos e Vulnerabilidades
    Documento que contém a lista de alvos e vulnerabilidades consolidado mediante o resultado das tarefas de varredura de redes, serviços e aplicações. Este documento deve auxiliar o especialista de segurança a preparar a estratégia de ataques em função de se evitar falsos positivos e atingir o escopo e o objetivo do *pentest*.
  - Registro das Varreduras de Ferramentas
    Repositório com os resultados das varreduras e seu detalhamento específico. Este repositório deve ter uma linha de tempo para avaliar possíveis mudanças de objetivos, contexto e do próprio escopo do projeto.
  - Definição da estratégia do *pentest*
    Este processo está relacionado as condições iniciais de preparar os planos que servirão de base para a execução do *pentest*. É importante se registrar o *rationale* e as estratégias que guiaram a formulação dos planos de ataques. Esta etapa diz respeito as linhas de ação que devem ser criadas mediante os objetivos, escopo e alvos em função da lista de alvos e vulnerabilidades.
  - Alvos e Táticas
    Para cada alvo, em função da estratégia, é necessário um conjunto de tarefas para filtrar a lista de alvos em função das necessidades estratégicas, objetivos e escopo.
  - Definir Alvos Prioritários
    Tarefa que deve ser realizada em função da estratégia definida e a lista de alvos levantada. Esta tarefa deve priorizar o alvo em função dos objetivos a serem alcançados, do escopo e critérios de oportunidade e valor agregado.
  - Definir Alvos Emergentes
    Identificar barreiras e sistemas a serem contornadas ou dominadas para se alcançar um alvo prioritário.
  - Definir Táticas e Ferramentas
    Tarefa responsável pela definição das táticas necessárias para se atacar determinado alvo e escolha da ferramenta ou conjunto de ferramentas que deve ser utilizado. Esta etapa também deve definir sequenciamento e temporização de ações

relacionadas à condução do ataque em função das ações que levam à consecução dos objetivos.

- Consolidação de alvos e táticas
  Correlação de alvos e táticas definidos nas tarefas relacionadas ao processo.
- Gerar Plano de Ataques
  Tarefa que deve definir as operações ou ataques que devem ser executados. Bem como a sequência em que devem ser executados seguindo uma ordem temporal paralela ou sequencial e definir os critérios de avaliação para o fim de um ataque e para o início de outro.
- Plano de Ataques
  Documento executivo que deve conter:
  * Lista de alvos, táticas de exploração e ferramentas indicadas;
  * Critérios de início e fim de ataques;
  * Sequência de execução paralela ou serial em função do tipo de alvo ou do tipo de ataque;
  * Lista de alvos emergentes e técnicas de exploração em função dos objetivos;
- Término da Estratégia do *Pentest*
  Conclusão da fase e geração dos respectivos artefatos.

4.2.7 Fase IV: Execução de Testes

Esta é a principal fase da metodologia, em que os testes são de fato realizados. As fases anteriores são a preparação desta, e as seguintes a consolidação e apresentação de seus resultados. Em termos de tempo, expertise e esforço, esta fase junto com a seguinte são as que apresentam maiores demandas. A execução dos testes deve seguir o plano de execução elaborado quando da definição das táticas. A condução dos ataques é um processo de tentativa e erro, sendo necessário avaliar os resultados de cada investida e assim promover ajustes.

O cerne desta fase é o ciclo de testes. Adotou-se uma adaptação do ciclo OODA que é geral o suficiente para modelar o fluxo de decisões que são tomadas quando da condução de um teste ou ataque. Entretanto, mesmo com tal generalidade, o ciclo de testes ainda é capaz de acomodar as características dos testes, inclusive os voltados a avaliações de disponibilidade. Esta característica não está prevista em outras metodologias.

Além disso, o ciclo de teste desacopla totalmente a execução do teste do suporte ferramental que venha a ser utilizado. Dessa forma, a metodologia se torna independente de, ou agnóstica com relação a, ferramentas.

O processo de teste pode ser compreendido como uma série de estímulos e respostas. Os estímulos são determinados pelos objetivos dos testes e pelas condições do ambiente quando

da execução. As respostas aos estímulos, por sua vez, são analisadas e o teste como um todo avaliado para se decidir se pode-se parar ou se novos estímulos deverão ser aplicados ao alvo sob teste.

A Figura 4.5 ilustra o Ciclo de Testes nesta Fase.

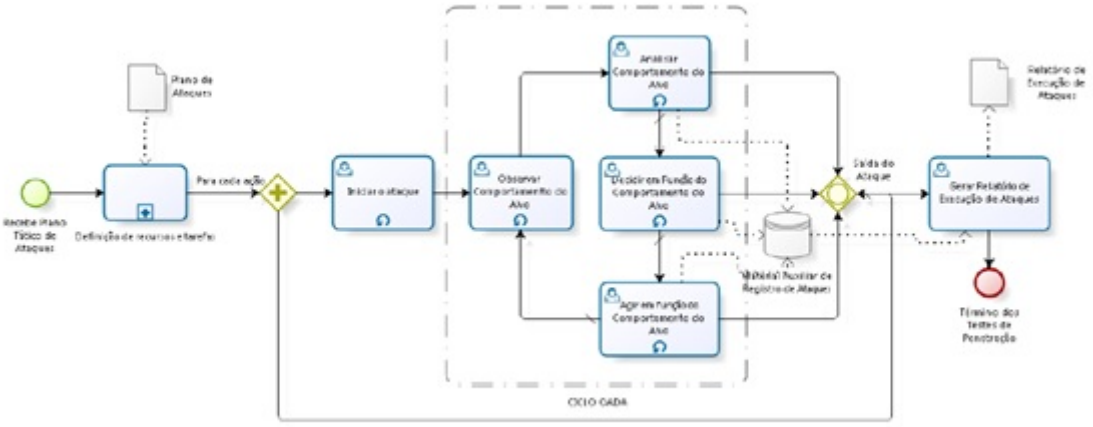

Figura 4.5: Fase 4 - Execução de Testes.

4.2.8 Atividades da Fase IV

As atividades dos ciclo são descritas abaixo.

- Definição de Recursos e Tarefas

  De posse do plano de ataques, os recursos e tarefas são alocados para a realização dos testes.

  O plano de ataques deve prever a condução dos testes levando em consideração a possibilidade de execução de testes em paralelo sempre que possível. Se os alvos são independentes e a equipe e os equipamentos permitirem, a opção por testes em paralelo deve ser exercitada.

- Iniciar o Ataque

  Esta ação marca o início de um teste ou ataque específico no conjunto dos testes de penetração, lembrando que os testes podem se realizados de forma serial ou em paralelo, segundo o plano de ataques.

- Ciclo de Teste - OADA

  No processo anterior foram gerados os planos de testes para exploração das vulnerabilidades selecionadas nos alvos priorizados. Neste processo, cada um daqueles planos será executado. A execução dos planos de teste normalmente é interativa. Visando

lidar convenientemente com esta característica, concebeu-se a execução das ações em ciclos em que se tem a chance de adaptar as ações planejadas aos resultados obtidos.

Estes ciclos são executados até que uma destas situações ocorra:

- o objetivo foi atingido, tendo assim o teste obtido sucesso;
- o teste se encerrou por término do tempo disponível, ou esgotamento de algum outro recurso, tendo o teste falhado total ou parcialmente;
- o teste foi interrompido por inadequação do planejamento, falhando totalmente ou parcialmente.

Ao final dos ciclos de teste, para cada plano executado, deve-se ter um relatório descrevendo as ações executadas e os resultados obtidos. O ciclo de teste é composto por quatro etapas:

- Observação do contexto:
  Consiste na verificação das precondições anteriores à execução das ações previstas no plano de testes. Como esta ação é a primeira do ciclo, ao se executar a última do ciclo, esta verificará as poscondições resultantes da execução das ações.
- Análise do contexto:
  Observado o contexto, deve-se analisá-lo para avaliar a eficácia das ações executadas e, se necessário, sugerir medidas corretivas que visem realinhar a condução das ações para a consecução do objetivo do teste.
- Decisão de ação:
  À luz da análise do contexto e do plano de teste, deve-se decidir a ação a implementar. Vale lembrar que as ações podem ser exploratórias, principalmente se os resultados das ações divergirem do esperado.
- Implementação de ação:
  Execução da ação decidida na atividade anterior. Uma vez completa, volta-se à observação.

- Gerar Relatório de Execução de Ataques

  Esta atividade compreende a geração de um documento com o relato dos ataques realizados e resultados obtidos, além de evidências destes. Este relatório será o principal insumo para a elaboração do relatório final.

- Detalhamento das Atividades e Artefatos da Fase IV

  - Recebe Plano Tático de Ataques
    Recebimento do plano tático de ataques para início das atividades do ciclo OADA.

– Plano de Ataques
  Documento executivo que deve conter:
– Lista de alvos, táticas de exploração e ferramentas indicadas;
– Critérios de início e fim de ataques;
– Sequencia de execução paralela ou serial em função do tipo de alvo ou do tipo de ataque;
– Lista de alvos emergentes e técnicas de exploração em função dos objetivos;
– Definição de recursos e tarefas
  Processo que deve utilizar o plano de ataques e definir em função do conjunto de ações organizadas do plano a metodologia de observar, analisar, decidir e agir (OADA) os recursos necessários para implementação de tarefas.
– Para cada ação
  Tomar decisão se deve se iniciar ou continuar o ataque ou finalizar o mesmo em termos de execução dos testes de penetração.
– Iniciar o ataque
  Executar as ações para o início de um ataque em função das vulnerabilidades.
– Observar Comportamento do Alvo
  Condição de perceber o comportamento do alvo em preparação ou sob execução do ataque.
– Analisar Comportamento do Alvo
  Condição de realizar inferências analíticas, formular hipóteses e propor estímulos ao alvo em função do comportamento do mesmo.
– Decidir em Função do Comportamento do Alvo
  Condição de escolha em função da análise de qual deve ser a próxima ação a ser executada em função do comportamento do alvo.
– Agir em função do Comportamento do Alvo
  Executar a decisão realizada na tarefa anterior.
– Material Auxiliar de Registro de Ataques
  Repositório dos registros de execução do plano de ataque para fins de documentação e possível comprovação de fatos.
– Saída do Ataque
  Condição para se reiniciar o ataque em função de um outro alvo ou gerar relatório de execução de ataques.
– Gerar Relatório de Execução de Ataques
  Tarefa que deve criar documento contendo a memória da execução dos ataques e o respectivo comportamento do alvo.
– Relatório de Execução de Ataques
  Documento que deve conter o relato dos ataques realizados e correspondente

comportamento do alvo, além das principais evidências que corroborem as afirmações e fatos descritos.

– Término dos Testes de Penetração
Conclusão dos testes de penetração e geração dos respectivos artefatos.

4.2.9 Fase V: Consolidação dos resultados

Nas fases anteriores são gerados documentos tais como: relatórios das varreduras, planos de ataque, resultados dos ataques, análises destes resultados e *log* das atividades. Eles servirão de base para a análise que irá decidir se os testes devem parar ou continuar. Se avaliado que são necessários mais testes, deve-se retornar à fase III e com os novos achados readequar alvos e planos para uma nova bateria de testes. Se avaliado que os objetivos foram alcançados, o teste pode ser encerrado. Por fim, se as oportunidades de ataque, o tempo ou algum outro recurso se esgotaram, os testes também devem ser interrompidos. Por outro lado, os documentos gerados nas fases anteriores devem ser consolidados para servir de base para a apresentação dos resultados, bem como dos procedimentos realizados (Figura 4.6).

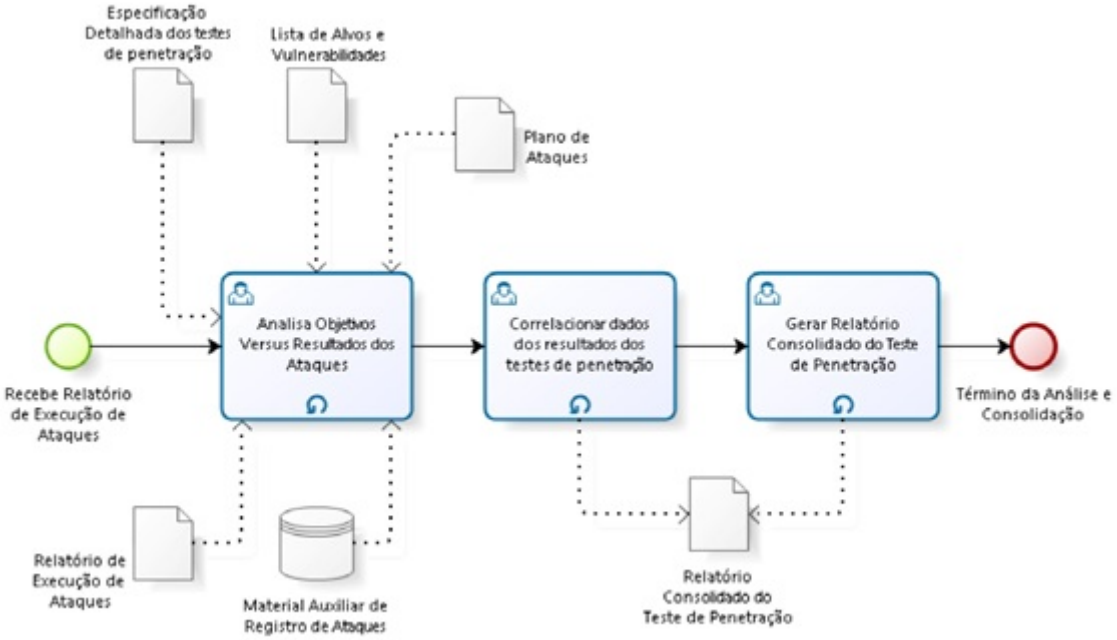

Figura 4.6: Fase 5 - Consolidação dos resultados.

4.2.10 Atividades da Fase V

As atividades da Fase V são detalhadas a seguir.

- Analisar Objetivos versus Resultados dos Ataques

Esta atividade consiste no batimento entre os resultados obtidos e os objetivos dos testes. Se os testes realizados atingirem resultados satisfatórios, não serão necessários novos testes. Caso sejam necessários novos testes, será necessário avaliar se ainda há tempo e recursos para que estes sejam viáveis. Assim, será necessário retornar à seleção de alvos e planejamento de ataque para preparar novos testes.

Outra possibilidade é a situação em que os resultados sugerem caminhos de exploração e aprofundamento que podem ir além do planejamento inicial. Novamente, será preciso avaliar se há tempo e recursos para a execução. Além disso, será preciso avaliar se os possíveis resultados adicionais são relevantes.

- Correlacionar Dados dos Resultados dos Testes de Penetração

  Esta atividade envolve a correlação dos dados e da documentação gerada nas fases anteriores, uma vez que é necessário conciliar possíveis resultados conflitantes ou preencher lacunas de resultados incompletos correlacionando informações na grande quantidade de documentos que foi gerada. Dessa forma, apesar destes serem necessários para a devida aplicação da metodologia, é bem possível que nem todos venham a constar dos resultados que serão apresentados. Deve-se lembrar que o conjunto dos documentos gerados deve ser devidamente arquivado como a memória técnica do projeto.

  Esta fase produz insumos para o relatório consolidado de testes de penetração.

- Gerar Relatório Consolidado do Teste de Penetração

  Esta atividade é a consolidação de todos os resultados obtidos, não só nos testes mas também nas varreduras da Fase II.

  A consolidação geral envolve:

  - Consolidação de resultados de testes:
    Os resultados dos testes devem ser consolidados e correlacionados. Deve-se também escolher o que será incluído no relatório final.
  - Consolidação de varreduras:
    Apesar das varreduras já terem sido consolidadas anteriormente, deve-se confrontá-las com os resultados dos testes para que se tenha a visão consolidada da topologia e serviços da rede.

  Por fim, o relatório consolidado é elaborado, sendo composto por um resumo executivo e um relatório técnico. O resumo é um documento que tem como público alvo as altas camadas gerenciais, servindo como base à tomada de decisão. Nele não devem constar informações técnicas. Estas são detalhadas no relatório técnico que deverá servir de base para a definição de ações de tratamento e implementação de controles para as vulnerabilidades encontradas e exploradas.

- Detalhamento das Atividades e Artefatos da Fase V

– Recebe Relatório de Execução de Ataques
  Início das atividades com o recebimento do relatório de execução de ataques.
– Especificação Detalhada do Projeto
  Documento Executivo que contém objetivos, escopo, recursos, prazos, condições de execução relacionados ao projeto segundo entendimento do demandante e do demandado.
– Lista de Alvos e Vulnerabilidades
  Documento que contém a lista de alvos e vulnerabilidades consolidado mediante o resultado das tarefas de varredura de redes, serviços e aplicações. Este documento deve auxiliar o especialista de segurança a preparar a estratégia de ataques em função de se evitar falsos positivos e atingir o escopo e o objetivo do *pentest*.
– Plano de Ataques
  Documento executivo que deve conter:
  * Lista de alvos, táticas de exploração e ferramentas indicadas;
  * Critérios de início e fim de ataques;
  * Sequência de execução paralela ou serial em função do tipo de alvo ou do tipo de ataque;
  * Lista de alvos emergentes e técnicas de exploração em função dos objetivos;
  * Alocação temporal das atividades;
  * Indicação de técnicas e procedimentos;
– Relatório de Execução de Ataques
  Documento que deve conter o relato dos ataques realizados e o comportamento do alvo, além das principais evidências que corroborem as afirmações e fatos descritos.
– Material Auxiliar de Registro de Ataques
  Repositório dos registros de execução do plano de ataque para fins de documentação e possível comprovação de fatos. É uma base auxiliar contendo registro de ataques e resultados dos mesmos.
– Analisar Objetivos Versus Resultados dos Ataques
  Tarefa onde devem ser analisados todos os artefatos produzidos no processo geral do teste de penetração e os resultados alcançados.
– Correlacionar dados dos resultados do *pentest*
  Tarefa em que devem ser relacionados os dados afins e principais ações realizadas nos ataques em função dos objetivos.
– Gerar Relatório Consolidado do Teste de Penetração
  Tarefa que deve gerar o documento executivo e consolidado do teste de penetração.
– Relatório Consolidado do Teste de Penetração
  Documento executivo que deve conter:

    - Sumário executivo;
    - Introdução;
    - Escopo;
    - Objetivo;
    - Lista de alvos;
    - Relatório de execução dos ataques e resultados obtidos para cada ataque realizado;
    - Principais conclusões;
- Término da Análise e Consolidação
  Conclusão da consolidação e análise e geração dos respectivos artefatos.

4.2.11 Fase VI: Apresentação dos resultados

Após todos estes documentos gerados ao longo do processo serem consolidados ocorre a apresentação dos resultados, sendo parte fundamental do processo de convencimento do demandante acerca de suas vulnerabilidades, para que ele se dê conta não só da importância das informações comprometidas mas também da forma como foram obtidas.

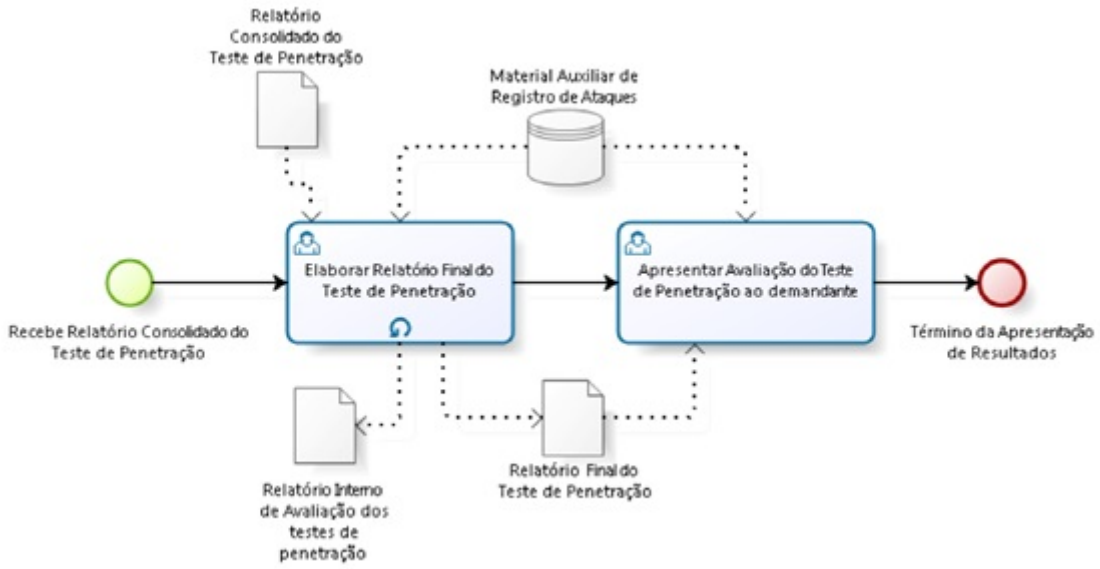

Figura 4.7: Fase 6 - Apresentação de resultados.

4.2.12 Atividades da Fase VI

Nesta fase (Figura 4.7) temos a elaboração do relatório final, bem como sua apresentação e entrega.

- Elaborar Relatório Final do Teste de Penetração

Nesta atividade será elaborado o relatório final em que são descritos os principais achados e resultados dos testes executados. Este relatório consolida todo processo. O relatório deve ter 3 (três) grandes blocos:

- o resumo executivo:
  breve descrição dos resultados, sem viés técnico, e uma avaliação geral da segurança da informação, destinada ao escalão mais alto da organização.
- o relatório técnico:
  com a descrição detalhada dos resultados encontrados e os métodos utilizados, destinado ao corpo técnico da organização; em certa medida, este relatório deve ser didático, principalmente na descrição dos métodos .
- apêndices:
  informações adicionais, como relatórios varreduras e ferramentas, listagens, etc, que venham a dar suporte ao corpo do relatório.

- Apresentar Avaliação do Teste de Penetração ao Demandante

  Com base no relatório, deve-se preparar uma apresentação com os resultados de maior impacto e uma avaliação do estado geral do ambiente e dos alvos examinados. Nesta apresentação certamente estará o pessoal técnico e possivelmente membros do alto escalão administrativo da organização. Além dos recursos usuais, um recurso adicional a ser usado na apresentação são filmes da realização dos testes, caso tenham sido gerados durante o processo. Estes filmes cumprem a função de convencimento da equipe técnica da organização.

  As bases do filme são o plano de teste e o relatório de sua execução. Quase sempre alguns testes precisam ser repetidos. A elaboração de um filme envolve:

  - definição do roteiro:
    com todos os passos a serem executados;
  - gravação do filme:
    produção do filme em si.

  O relatório final é o produto final de todo o processo dos testes de penetração. Sua entrega é o marco formal que encerra todo o processo dos testes de penetração.

- Detalhamento das Atividades e Artefatos da Fase VI

  - Recebe Relatório Consolidado do Teste de Penetração
    Insumo para início da fase de apresentação de resultados.
  - Relatório Consolidado do Teste de Penetração
    Documento executivo que deve conter:
    * Sumário executivo;

    - Introdução;
    - Escopo;
    - Objetivo;
    - Lista de alvos;
    - Relatório de execução dos ataques e resultados obtidos para cada ataque realizado;
    - Principais conclusões;

– Material Auxiliar de Registro de Ataques

  Repositório dos registros de execução do plano de ataque para fins de documentação e possível comprovação de fatos.

– Elaborar Relatório Final do Teste de Penetração

  Tarefa básica para gerar a versão final do teste de penetração e eventuais ajustes de documentação para realizar apresentação dos resultados ao cliente.

– Relatório Interno de Avaliação do Projeto
  Documento descritivo e analítico que deve conter informações como:
    - Avaliação do planejamento do projeto;
    - Avaliação de adequação, eficácia e eficiência das ferramentas, técnicas e procedimentos utilizados;
    - Lições aprendidas e principais recomendações;

– Relatório Final do Teste de Penetração
  Documento final baseado no relatório consolidado do teste de penetração e eventuais correções e adequações. Este documento deve conter:
    - Sumário executivo;
    - Introdução;
    - Escopo;
    - Objetivo;
    - Lista de alvos;
    - Relatório de execução dos ataques e resultados obtidos para cada ataque realizado;
    - Principais conclusões;

– Apresentar Avaliação do Teste de Penetração ao Demandante
  Tarefa de realizar apresentação executiva dos resultados ao demandante e eventual demonstração de fatos utilizando o repositório de material auxiliar de registro de ataques.

– Término da Apresentação de Resultados

Conclusão da apresentação e recebimento de análises decorrentes da apresentação.

4.3 Resumo do Capítulo 4

A metodologia de testes de penetração faz parte de um processo contínuo de melhoria. Todas as informações obtidas formam uma base de conhecimento essencial para a equipe de análise, atingindo um grau de maturidade cada vez maior a cada ciclo de processo executado.

Dentro da gestão de segurança da informação, dão subsídio para a análise gerência de risco, e deve ser sempre realizada, o que torna mais natural o próprio processo de execução, que leva em consideração a experiência e a base de conhecimento, que é particular para cada organização analisada.

Assim, o primeiro ciclo de aplicação da metodologia é normalmente mais longo do que os demais. A base de conhecimento que é formada pelos demais ciclos tende a crescer, passando a ser fundamental principalmente para a proteção de infraestrutura crítica da organização em que os testes foram realizados.

No Apêndice 6.1 é apresentado um resumo de fases, processos, atividades e artefatos que compõem a metodologia.

Capítulo 5

Aplicações e Resultados

A metodologia DOTA descrita no Capítulo 4 foi aplicada sob dois focos distintos. O primeiro foco, mais usual para aplicação de metodologias de testes de penetração, foi o de testes voltados a intrusão em sistemas visando acesso, controle e exfiltração de informações. O outro contexto, que não é coberto pelas metodologias abordadas no Capítulo 2.2, foi o de testes voltados à indisponibilidade especificamente no contexto de IoT.

A seguir são descritas as aplicações de DOTA sob estes dois focos.

5.1 Aplicações de DOTA Voltada à Intrusão

Tradicionalmente, as metodologias de testes de penetração são voltadas à avaliação da segurança de sistemas com relação à invasão. Aqui, entende-se por invasão as atividades referentes a contornar sistemas de controle de acesso e privilégio tendo como objetivo final a obtenção de controle total sobre os sistemas invadidos. Assim, a metodologia foi testada no contexto de invasão, que é aquele que foi originalmente projetada. Apesar de seu caráter prescritivo, a metodologia também engloba o registro e descrição de melhores práticas. A intenção era chegar a uma metodologia aplicável, próxima da cultura já existente, e gerando resultados úteis.

A abordagem adotada inicialmente no seu desenvolvimento foi a de testar as fases da metodologia isoladamente, visando ajustes e adequações. Estes testes iniciais foram realizados pelo autor em ambiente controlado. Houve também testes de bancada envolvendo uma passada por todas as fases, com resultados que permitiram consolidar uma versão da mesma.

Na sequência, foram realizados teste em ambientes reais, inclusive de produção. Foram realizadas três baterias de testes em cenários distintos. Estes testes, apesar de motivados pela oportunidade de realização, cobrem aspectos relativos aos objetivos a que se propunha a metodologia. Os testes voltados à intrusão, seus cenários de aplicação específicos e resultados obtidos são relatados a seguir.

5.1.1 Cenário 1: teste interno de infraestrutura e aplicações

O cenário do primeiro conjunto de testes consistiu em uma rodada completa da metodologia como parte de um processo de avaliação de riscos executado em um organização de médio porte. A metodologia foi executada em paralelo como uma das fases de um processo de análise de risco visando a identificação e avaliação dos riscos associados ao ambiente tecnológico.

- Aplicação da metodologia

  As condições, que refletem os requisitos para execução deste primeiro teste da metodologia foram (Fase 1):

  - não havia queixa específica relatada, sendo o objetivo prover uma visão panorâmica do estado da rede, sistemas, serviços e aplicações;
  - acesso interno à rede sob teste, inicialmente sem e depois com credenciais de usuário;
  - escopo abrangendo infraestrutura de conectividade e rede, serviços básicos (DNS, SMTP, HTTP, e afins) e também aplicações, tanto do lado dos servidores quanto dos clientes;
  - não seria testada a disponibilidade, recaindo o foco sobre confidencialidade e integridade;
  - a equipe de TI da organização tinha ciência dos testes, não sendo necessário executá-los de maneira furtiva;
  - equipe composta por duas pessoas;
  - prazo para realização de duas semanas.

  Estabelecidas as condições, foram realizadas as varreduras (Fase 2) na sequência redes, vulnerabilidades (sistemas e serviços) e aplicações. Foram identificadas vulnerabilidades e determinados alvos potenciais nas varreduras de rede e aplicação. Já na varredura de serviços, ocorreu uma situação de negação de serviço provocada pela baixa capacidade da rede. Mesmo reduzindo a taxa de injeção de *probes* (estímulos) a valores mínimos, os servidores não suportavam os testes. Esta varredura foi executada outras duas vezes com resultados idênticos. Repactuou-se então o escopo, sendo retirada a avaliação dos serviços.

  De posse de tais resultados, foram escolhidos alvos entre os equipamentos de rede e aplicações (Fase 3). Devido aos problemas com as varreduras de serviços, os servidores ficaram fora do teste. Foram priorizados os equipamentos segundo sua criticidade na rede. As aplicações foram priorizadas pela sua importância e alinhamento com a missão da organização. Foi assim possível definir a sequência dos testes (Fase 4).

Tabela 5.1: Resultados do Primeiro Teste

	Alvos	Vulnerabilidades		
	Explorados/Total	Alta	Média	Baixa
Infraestrutura	15/15	30	15	15
Aplicações	6/6	9	5	3

Executados os testes (Fase 5), os resultados obtidos foram: todos os equipamentos de rede (quinze), incluindo *core*, foram comprometidos com acesso de administrador; todas as aplicações web (seis) foram comprometidas com diversas vulnerabilidades (total de dezessete), como *SQL injection* e *XSS*, entre outras.

O relatório final foi elaborado (Fase 6) em aproximadamente um terço do tempo disponível, graças ao registro sistemático das fases anteriores, conforme preconizado pela metodologia.

- Resultados obtidos

  A execução do ciclo completo da metodologia em um ambiente de produção mostrou que era possível obter resultados válidos em tempo razoável. Posteriormente, a organização demandante mostrou-se satisfeita com o trabalho. Apesar da supressão dos testes dos servidores, ocasionada pela negação de serviço, esta foi considerada como um resultado válido que apontava a necessidade de atualização tecnológica da infraestrutura de conectividade da organização. A tabela 5.1 dá os números dos resultados obtidos.

5.1.2 Cenário de Teste 2: teste externo de aplicação

O cenário do segundo teste também consistiu em uma rodada completa da metodologia tendo por finalidade avaliar o serviço HTTP de uma organização de pequeno porte. A metodologia foi executada isoladamente visando a identificação e avaliação dos riscos associados ao sítio web da organização e suas aplicações. Como se tratava de um ambiente de produção e a organização demandante não tolerava interrupções de serviço durante os testes, o ambiente foi clonado e disponibilizado em um *host* espelho onde os testes foram efetivamente realizados.

- Aplicação da metodologia

  As condições para execução deste segundo teste da metodologia foram:

  - havia queixa específica com relação à segurança do sítio web da organização, sendo o objetivo avaliar as condições do servidor e aplicações;
  - acesso externo à rede sob teste, sem credenciais de usuário;
  - escopo abrangendo serviços HTTP e aplicações web, tanto do lado dos servidores quanto dos clientes;

  - não seria testada a disponibilidade, uma vez que não podia haver interrupção do serviço;
  - os testes tinham foco em integridade e confidencialidade;
  - a equipe de TI da organização tinha ciência dos testes, não sendo necessário executá-los de maneira furtiva;
  - equipe composta por duas pessoas;
  - prazo para realização de quatro semanas.

  Estabelecidas as condições (Fase 1), foram realizadas as varreduras na sequência serviços e aplicações (Fase 2). Uma vez que o alvo já estava determinado quando da definição das condições de execução, foram identificadas vulnerabilidades potenciais nas varreduras de serviço e aplicação.

  De posse de tais resultados, foram escolhidas as vulnerabilidades alvos dentre as encontradas (Fase 3). As vulnerabilidades foram priorizadas segundo sua criticidade no servidor. As vulnerabilidades das aplicações também foram priorizadas pela criticidade, ponderada com sua importância e alinhamento com a missão da organização (Fase 4).

  Executados os testes (Fase 5), os resultados obtidos foram: sistema operacional e servidor HTTP desatualizados com várias vulnerabilidades críticas; todas as aplicações web comprometidas com as vulnerabilidades da lista *Top 10* da OWASP, com mais de uma instância; todos componentes de software utilizados nas aplicações web (cinco) também apresentaram vulnerabilidades críticas.

  O relatório final foi elaborado em aproximadamente um quinto do tempo disponível, graças ao registro sistemático das fases anteriores, segundo a metodologia.

- Resultados obtidos

  A execução do ciclo completo da metodologia em um ambiente de produção replicado deu aos executores a liberdade de testar sem preocupações com a disponibilidade. Novamente, foi possível obter resultados válidos em tempo razoável. A organização demandante avaliou positivamente o trabalho. Como o diagnóstico obtido apontava na direção da desatualização generalizada, a organização posteriormente decidiu pela construção de um novo sítio, uma vez que constatou-se que as desatualizações se deviam à descontinuação do desenvolvimento de componentes críticos das aplicações web. A tabela 5.2 sumariza os resultados obtidos.

5.1.3 Cenário de Teste 3: comparativo

O cenário do terceiro teste foi o mais complexo e tinha por finalidade prover uma base de comparação, senão da metodologia em si, pelo menos da abordagem que ela adota. Outro objetivo do teste era consolidar a modelagem da execução de ataque com ciclos OADA. Este

Tabela 5.2: Resultados do Segundo Teste

Alvos	Vulnerabilidades/Instâncias		
	Alta	Média	Baixa
Sistema Operacional	1/1	3/3	2/2
Compenentes	12/18	5/8	4/7
Aplicação	10/13	4/4	5/8

teste também consistiu em uma execução completa da metodologia, entretanto esta ocorreu paralelamente a outras metodologias sendo executadas.

- Aplicação da metodologia

  O formato do teste foi o de uma competição. Foram formadas cinco equipes, cada uma com cinco membros. As equipes estavam niveladas pelo fato de ter pouca exposição às técnicas e ferramentas de *pentest*. Cada equipe podia utilizar a abordagem metodológica que desejasse, sendo que uma delas necessariamente usaria a metodologia aqui proposta.

  Os testes não foram executados em um ambiente real de produção, mas em uma rede simulada, com cada equipe realizando os testes em sua instância de simulação. O caráter do teste foi exploratório: as equipes receberam apenas um endereço inicial e a tarefa geral era identificar *hosts*, encontrar suas vulnerabilidades e explorá-las.

  As regras da competição foram as seguintes:

  - cada equipe teve acesso a uma instância de um ambiente simulado, sendo disponibilizada com a mesma topologia para todas as equipes;
  - as equipes não conheciam da antemão a topologia do ambiente, sendo-lhes passado apenas o endereço do *host* que dava acesso ao ambiente;
  - os *hosts* no ambiente executavam diferentes sistemas operacionais, apresentando diferentes vulnerabilidades;
  - em cada *host* identificado, cada equipe devia executar três tarefas diferentes de sua escolha envolvendo exploração de vulnerabilidades, (*e.g.* acesso indevido, escalação de privilégio, alteração de arquivo do sistema, etc);
  - cada equipe tinha acesso a sua instância do ambiente simulado em janelas de quatro horas;
  - cada equipe tinha direto a cinco janelas de acesso, segundo um cronograma, que impunha um intervalo de 48 horas entre cada janela;
  - após a última janela, cada equipe tinha 48 horas para entregar o relatório.
  - seria declarada vencedora a equipe que dentro das regras obtivesse a maior pontuação pela realização do maior número de tarefas ponderada com a pontuação do relatório.

Tabela 5.3: Resultados do Terceiro Teste

Equipe	Metodologia	Alvos/Tarefas	Relatório	Escore
A	DOTA	3 / 9	8	98
B	baseada em ferramentas	1 / 2	6	26
C	baseada em ferramentas	1 / 1	6	16
D	voltada a projeto	1 / 3	10	40
E	baseada em ferramentas	1 / 2	6	26

A formatação do teste como uma competição visava desencorajar a troca de informação entre os participantes de diferentes equipes. Já a imposição do intervalo de 48 horas entre as janelas de acesso foi devida à limitação de recursos disponíveis. Entretanto, esse intervalo, junto com a janela restrita no tempo, enfatizava a necessidade de planejamento de ações e a otimização das decisões. Deve-se ressaltar que se desejava avaliar se a metodologia aqui apresentada de fato favorecia estes aspectos.

As cinco equipes cumpriram o cronograma de acesso à simulação dentro das regras e entregaram o relatório no prazo. A tabela 5.3 sintetiza os resultados obtidos por cada equipe, e as figuras 5.1 e 5.2 o detalham graficamente.

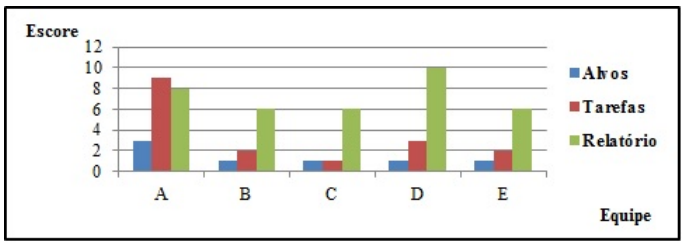

Figura 5.1: Comparativo entre equipes por alvos, tarefas e relato

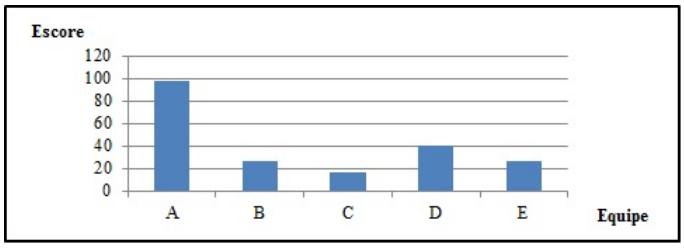

Figura 5.2: Escores finais por equipe

- Resultados obtidos

  Nota-se claramente que a equipe que adotou a metodologia DOTA obteve resultados superiores às outras equipes que usaram outras metodologias, tanto quantitativamente, pela quantidade de alvos e tarefas realizadas, quanto pela forma como todo o exercício foi conduzido. Em entrevista posterior aos testes, os membros da equipe A relataram que a primeira janela de acesso à simulação serviu apenas para reconhecimento, sendo

encontrado apenas um *host* sobre o qual foram executadas varreduras. Eles então aproveitaram o intervalo de 48 horas para avaliar resultados e decidir próximas ações.

Na segunda sessão, vencido o obstáculo do primeiro alvo, realizaram novas varreduras para reconhecer o novo ambiente, com análise de resultados e decisões sendo tomadas no intervalo seguinte. Na verdade, a equipe A identificou além da rede alvo, as redes de controle e simulação do ambiente. Seguindo a metodologia, a equipe manteve suas opções abertas realizando varreduras de amplo espectro e foram encontradas vulnerabilidades no software de gerenciamento da simulação. Com acesso ao código fonte deste software, desenvolveram uma ferramenta que deu-lhes a chance de controlar completamente a instância da simulação, incluindo parâmetros dos *hosts* simulados. Sucessivamente, foram realizando as tarefas e executando o ciclo OADA tanto na execução dos ataques em si quanto na avaliação das circunstâncias de execução.

5.1.4 Outros cenários

Cabe relatar que entre o segundo e o terceiro cenários de teste surgiu uma oportunidade de avaliar a flexibilidade de aplicação da metodologia. No caso, tratava-se de uma auditoria de segurança em um sistema de informação que devia ser avaliado antes de ser colocado em produção. A principal diferença em termos de aplicação da metodologia ocorreu na Fase II, onde se teve acesso a toda documentação de desenvolvimento do sistema em questão. O efeito de tal diferença foi que, dado o maior volume de informações relevantes acerca do sistema anterior às fases de planejamento e execução dos testes, houve uma amplificação dos resultados, *i.e.* obteve-se resultados mais abrangentes e aprofundados do que normalmente se obtinha nas condições normais de aplicação da metodologia.

5.1.5 Resumo dos Testes Orientados à Intrusão

Conforme relatado anteriormente, a metodologia aqui apresentada foi submetida a baterias de testes que demonstram que sua aplicação contribui na obtenção de resultados abrangentes, aprofundados e precisos.

No terceiro cenários de teste, durante a avaliação das equipes, os juízes precisaram repetir os testes. A única equipe que teve todos os seus resultados confirmados foi a vencedora. Assim, a aplicação da metodologia também contribui para a repetibilidade dos resultados. Além disso, a execução do ciclo OADA deixa evidente o ganho em se explicitar o processo de tomada de decisão, não só nas fases preparatórias, mas principalmente quando da execução dos testes.

Apesar de não ter sido um objetivo originalmente, mas certamente uma característica desejável, a flexibilidade de aplicação da metodologia em cenário diverso dos que lhe são tipicamente aplicáveis, no caso auditoria de segurança (Seção 5.1.4) também se apresenta

como um possível benefício de sua adoção.

A metodologia DOTA apresentou as seguintes características, que coincidem com seus requisitos:

- sua finalidade é especificamente teste de penetração;
- seu escopo é geral;
- seu processo está bem definido e detalhado;
- seu processo está organizado de forma coerente e consistente; e
- seu ciclo de execução de ataque é geral e capaz de acomodar novos ataques;

Desta forma, ainda que o volume de testes seja relativamente pequeno, os resultados não desabonam afirmar que a metodologia atinge os objetivos de repetibilidade e explicitação da tomada de decisão, além de também apresentar flexibilidade de aplicação.

5.2 Aplicação de DOTA Voltada à Indisponibilidade

Conforme descrito na Capítulo 4, e corroborado pelos resultados da Seção 5.1.5, a metodologia DOTA foi desenvolvida como uma metodologia geral para a avaliação ativa de vulnerabilidades, e sua aplicação em vários cenários diferentes levou à obtenção de resultados relevantes, demonstrando sua flexibilidade de uso. Apesar daqueles cenários se concentrarem nos requisitos de confidencialidade e integridade, eles incidentalmente incluíram questões de disponibilidade que surgiram como restrições na realização dos teste. Entretanto, a avaliação de aspectos de disponibilidade não estava envolvida nos objetivos e ficou claro que a aplicação do DOTA não seria simples e direta sendo necessárias algumas considerações adicionais. No entanto, também estava evidente que a DOTA poderia acomodar este novo objetivo.

No caso, DOTA foi adaptada e aplicada a um cenário de testes orientados à disponibilidade. Especificamente, avaliaram-se os possíveis impactos de ataques distribuídos de negação de serviço por reflexão amplificada no contexto de redes IoT.

5.2.1 Aplicando DOTA para avaliar AR-DDoS sobre IoT

Como visto na Seção 2.5.2, o consenso na comunidade de segurança é que os ataques DDoS representam uma séria ameaça à IoT, com seus dispositivos servindo como alvo direto, ou sendo abusados para atingir algum alvo. Essa percepção, juntamente com a flexibilidade de DOTA levaram à sua aplicação voltada à indisponibilidade no contexto de IoT.

A avaliação da aplicabilidade da metodologia a ataques de indisponibilidade teve como seu principal objetivo avaliar a ameaça à IoT representada por ataques AR-DDoS. O AR-DDoS foi escolhido entre outras formas de DDoS por sua relativamente baixa complexidade e esforço de ataque, grande disponibilidade de potenciais refletores e alta eficiência. De modo a se obter uma avaliação completa e adequada em relação aos ambientes de IoT, é necessário avaliar os impactos sobre refletores e atacantes, e os papéis que dispositivos IoT poderiam desempenhar no caso de um ataque AR-DDoS.

A principal hipótese de interesse a ser verificada está relacionada ao comportamento do refletor, em termos de saturação. Este ponto é considerado crucial na avaliação do potencial abuso de IoT, uma vez que seus dispositivos seriam principalmente refletores em um ataque desse tipo. Especificamente, o comportamento de saturação dos dispositivos IoT quando usado como refletores deve ser determinado. Observe-se também que a saturação pode ocorrer como exaustão de largura de banda ou carga de processamento excessiva.

Metodologicamente, a primeira opção considerada para caracterizar esse tipo de comportamento seria via simulação de rede. No entanto, os resultados seriam restritos à saturação devido à exaustão da largura de banda. Assim, a simulação não foi escolhida.

Uma segunda opção foi avaliar o comportamento de saturação usando técnicas de teste de estresse, mas essa abordagem também foi considerada inadequada. A principal razão é que esses testes, embora eficientes em determinar capacidade, não incluem a perspectiva do atacante em seu modelo; ou seja, a influência de uma entidade que decide o curso das ações durante um ataque para deliberadamente causar interrupção do serviço.

Implementar e executar uma instância de um ataque (neste caso, AR-DDoS) em um ambiente controlado era uma opção a ser considerada. Com essa abordagem, não só o refletor, mas o comportamento dos vários atores participantes do ataque poderia ser estudado e ainda motivar questões de pesquisa interessantes. Por exemplo, se a saturação ocorre, como ela influencia a eficiência de ataque; ou, dado o comportamento geral de ataque, como estas informações podem ser usadas para melhorar as estratégias de mitigação e prevenção. Como primeira questão, havia a necessidade de escolher uma instância particular do ataque. Essa instância deveria representar uma ameaça realista aos dispositivos IoT. Houve também o requisito de dominar completamente o ciclo de ataque.

A questão da escolha do ataque e do desenvolvimento das capacidades operacionais motiva a aplicação da metodologia proposta: não apenas para orientar essa escolha, mas todo o processo de planejamento, implementação, e execução do ataque. Na verdade, a metodologia proposta foi adaptada para se concentrar na indisponibilidade.

Assim, a metodologia orientou a escolha do protocolo utilizado em AR-DDoS, planejamento de ataque, implementação e execução, que posteriormente apoiaram análise e apresentação de resultados.

Neste cenário, o objetivo é avaliar a viabilidade de ataques AR-DDoS na infraestrutura e nos dispositivos de uma rede IoT. Esperava-se que esta avaliação cobriria uma base quanti-

tativa e qualitativa para que fosse possível corroborar ou refutar afirmações sobre possíveis consequências e impacto, caso um ataque AR-DDoS viesse a abusar da infraestrutura de uma rede IoT. Particularmente, um ponto de interesse era a saturação, especificamente no refletor, uma vez que a saturação na vítima é o efeito esperado do ataque. Como restrição, os testes deveriam ser executados em um ambiente controlado, de forma a não tentar causar interrupção nos dispositivos em produção. (DOTA Fase 1).

Assumindo o ponto de vista de um atacante, foi necessário escolher o ataque AR-DDoS específico a ser executado. Isso implica que se defina o protocolo a ser utilizado nos refletores. Entre os protocolos utilizados nos dispositivos e infraestrutura de redes IoT, três se mostraram como candidatos para exploração em um ataque AR-DDoS. Eles são: *Constrained Application Protocol* (CoAP) [IETF], *Simple Service Discovery Protocol* (SSDP) [Forum], e *Simple Network Management Protocol* (SNMP) [Case et al. 1990] (DOTA Fase 2).

Os três são transportados pelo *User Datagram Protocol* (UDP), sendo que os dois primeiros (CoAP e SSDP) são baseados em mensagens do tipo HTTP e podem fornecer amplificação, enquanto o SNMP usa variáveis da MIB (*Management Information Base*). No caso de ataques AR-DDoS, há relatos de incidentes envolvendo SSDP e SNMP. Por outro lado, o SNMP é tipicamente usado na gerência de redes, e fornece a maior taxa de amplificação. Para as três possibilidades, se fazia necessário desenvolver uma ferramenta específica para a detecção de refletores e execução do ataque, diferindo apenas na mensagem de teste (*probe*) a ser montada e enviada. A Tabela 6 resume esses recursos.

Tabela 5.4: Comparação entre Protocolos Candidatos

Critério	**Protocolo**		
	CoAP	**SSDP**	**SNMP**
Usa UDP	✓	✓	✓
Mensagens tipo HTTP	✓	✓	
Tabelas e variáveis da MIB			✓
Abuso por AR-DDoS já relatado		✓	✓
Necessidade de ferramenta	✓	✓	✓
Amplificação	✓	✓	✓
Escopo IoT	✓	✓	✓
Escopo Geral		✓	✓

Adicionalmente, em termos de utilização, CoAP está associado com sensores e dispositivos IoT mais simples, enquanto SSDP e SNMP cobrem não só IoT mas o âmbito geral da Internet. Portanto, dada a alta taxa de amplificação, o grande potencial de refletores na Internet comparado com outras duas opções e a facilidade de implementar um gerenciador de dispositivos IoT realista e poderoso computacionalmente, o protocolo escolhido foi SNMP (DOTA Fase 3).

O próximo passo foi definir os requisitos de teste que determinaram os parâmetros de ataque específicos a serem implementados na ferramenta de detecção e ataque (DOTA Fase 4).

Para detecção, foi definido que não se devia apenas identificar o refletor, mas também estabelecer sua taxa de amplificação. Para o ataque, a ferramenta deveria agregar funcionalidades para permitir um estresse incremental, a saturação pudesse ser caracterizada.

O plano básico para os testes era executar o ataque contra um refletor que amplifica e envia o tráfego para o alvo. O ataque deve começar a uma taxa baixa e escalar gradualmente para volumes de tráfego mais altos, com todos os fluxos de tráfego devendo ser registados para análise e apresentação de relatórios. No que diz respeito ao ambiente, os requisitos estipulam que infraestruturas IoT de produção não deveriam ser afetadas. Assim, optou-se pelo uso de uma configuração isolada para executar os testes de forma controlada e sem causar transtornos.

5.2.2 Testes

Os testes centraram-se em ataques DDoS por reflexão amplificada que abusam do SNMP. O SNMP é usado no gerenciamento de redes e é um protocolo também comumente usado em IoT e redes de sensores.

Ataques AR-DDoS ficaram em evidência a partir de 2013, tornando-se não só mais freqüentes, mas também gradualmente envolvendo maiores volumes de tráfego [Prolexic 2015]. O que torna o AR-DDoS atraente para os atacantes é que requer muito menos esforço de preparação se comparados a outras formas de DDoS, onde a preparação para o ataque necessita do uso de *malware* no comprometimento de sistemas para formar *botnets*. No caso do AR-DDoS, os atacantes só precisam identificar nós que são vulneráveis à reflexão, geralmente devido a má configuração. A identificação de possíveis refletores é facilmente automatizada em scripts e programas, que por sua vez são modificados trivialmente para executar o ataque em si (DOTA Fase 5).

- Amplificação Usando o Simple Network Management Protocol (SNMP)

  SNMP é o protocolo de gerenciamento padrão no contexto do *Transmission Control Protocol/Internet Protocol*(TCP/IP), incluindo IoT e redes de sensores. Ele fornece mecanismos para a troca de informações entre dispositivos geridos e gestores. Em sua primeira versão, o SNMP usou as *community strings* para o controle de acesso. Cada mensagem SNMP tinha um *community name* que devia corresponder ao nome da comunidade do dispositivo acessado. Este mecanismo simples era insuficiente, uma vez que os *community names* eram transmitidos em texto claro, de modo que podiam ser conhecidos por simples captura de tráfego. Além disso, havia nomes de comunidade padrão como *public* e *private*, tornando ainda mais fácil o acesso e a modificação de informações de gerenciamento de dispositivos.

  No SNMP versão 2, a operação GetBulkRequest foi introduzida para reduzir o tráfego de pedidos entre gerente e dispositivos gerenciados. Assim, uma única solicitação gera respostas longas (*i.e.* amplificação). Esta operação também está presente na

versão 3. No entanto, a segurança da versão 3 fornece controle acesso muito mais forte controle, incluindo autenticação e criptografia [Blumenthal e Wijnen 2002]. A versão SNMPv2c combina operação GetBulkRequest com controle de acesso baseado em *community names* [Case et al. 1996], como na versão 1, e ainda é amplamente utilizada. Esta combinação de uma operação que gera amplificação de tráfego com controle de acesso fraco torna o SNMPv2c facilmente explorável em ataques AR-DDoS. [Open SNMP Scanning Project 2016] mostra números de potenciais refletores SNMPv2.

- Preparação e Operacionalização dos Testes

  Para executar testes de forma repetível possibilitando entender e controlar completamente o ciclo de ataque, optou-se por desenvolver uma ferramenta para implementar a descoberta de refletores e o próprio ataque em si. A ferramenta desenvolvida usa a operação *GetNextRequest*, que consulta um único parâmetro para descoberta e a operação *GetBulkRequest*, com valores que levam a atingir atingem a taxa de amplificação máxima para o dispositivo. A ferramenta Foi desenvolvida em Java, devido à sua portabilidade, e em C, dada a sua rápida execução e controle total sobre a construção de datagramas.

  Para a descoberta do refletor em um dado intervalo de endereços, a ferramenta envia mensagens *GetNextRequest*. Se uma resposta é recebida, um possível refletor é encontrado. Após esgotar o intervalo, para cada possível refletor encontrado, a taxa de amplificação máxima é determinada. O fator de amplificação γ pode ser expresso tanto em amplificação de pacotes (γ_{pkt}) ou de bits (γ_{bit}). γ_{pkt} é a relação entre o número de pacotes de entrada e de saída, pkt_{in} e pkt_{out}, respectivamente, recebidos ou enviados durante o ataque (Equação 5.1):

$$\gamma_{pkt} = \frac{pkt_{out}}{pkt_{in}} \tag{5.1}$$

  Enquanto γ_{bit} é a relação entre os fluxos de bits de entrada e de saída, bit_{in} e bit_{out}, respectivamente, recebidos ou enviados durante o ataque, como na Equação (5.2):

$$\gamma_{bit} = \frac{bit_{out}}{bit_{in}} \tag{5.2}$$

  As mensagens *GetBulkRequest* são enviadas com uma lista de variáveis e parâmetros *NonRepeaters* e *MaxRepetitions*. O dispositivo gerenciado toma os valores correspondentes às primeiras n variáveis, Onde $n = NonRepeaters$. Para as variáveis restantes, tomam-se os vales correspondentes às m variáveis seguintes, onde $n = NonRepeaters$. Para a taxa de amplificação máxima, um único valor de variável é solicitado $NonRepeaters = 0$. A escolha para *MaxRepetitions* deve ser tal que o tamanho total do datagrama resultante da consulta não exceda 65.535 bytes. Se tal limite for excedido (o tamanho máximo para um datagrama IP), não haverá res-

posta. A abordagem adotada define o valor para gerar datagramas com pouco menos de 65 mil bytes. Esta estratégia conservadora é mais segura, já que diferentes dispositivos armazenam diferentes quantidades de informações. Assim, o mesmo valor para *MaxRepetitions* enviados para diferentes dispositivos pode gerar respostas com tamanhos diferentes. Um valor máximo típico é 2250 (ver Tabela 5.5).

Tabela 5.5: Parâmetros usados na operação *GetBulkRequest*.

Parâmetro	**Valor**
NonRepeaters	0
MaxRepetitions	2250

Assim, o valor de *MaxRepetitions* para amplificação máxima é definido por aproximação progressiva: um valor inicial é definido e a mensagem é enviada. Após recebimento da resposta, o valor é incrementado e o processo, transparente para o usuário, é repetido até que o tamanho da resposta se aproxime do tamanho máximo do datagrama (65535). Esta abordagem gera mais tráfego, porém consegue uma amplificação quase ótima, que é um diferencial em comparação com outras ferramentas [Prolexic 2015].

Para o ataque, os refletores que serão efetivamente usados são marcados na lista de candidatos e o alvo definido, antes de iniciar a execução. A ferramenta oferece oito níveis de intensidade. Em cada nível, O número de *probes* geradas é especificado. Começa-se gerando um pacote por segundo no nível 1, e em cada nível subsequente, a taxa é multiplicada por dez. Com oito níveis, o comportamento de saturação é facilmente observado. Isto também satisfaz a exigência de que a ferramenta permaneça atualizada, mesmo ainda que o hardware ou infraestrutura venha a evoluir.

- Requisitos do Teste

  Os testes foram focados na caracterização quantitativa de ataques AR-DDoS. Os objetivos específicos foram: estimar o esforço de ataque e sua eficácia; e identificar limites de saturação no atacante, refletor e vítima. Deve-se ressaltar que não se encontrou trabalho ou ferramenta similar para comparar com os resultados obtidos.

  Com relação à configuração, o refletor representa o dispositivo numa rede IoT ou de sensores. Sua configuração foi deliberadamente escolhida para ser muito mais robusta do que a de dispositivos típicos em uma dessas redes. As razões dessa escolha foram: os testes tinham por finalidade avaliar a saturação, e se isso ocorreu com pequenos volumes de tráfego tráfego, o comportamento do ataque de interesse não seria registrado; a configuração do refletor é consistente com a de uma estação de gerência de rede, inclusive a de um em uma rede IoT ou de sensores, sendo assim uma configuração realista; por fim, resultados e conclusões para dispositivos com limitações de poder computacional, largura de banda, ou consumo energético são válidas, com a ressalva de que a saturação irá ocorrer para taxas de tráfego muito mais baixas.

Os ensaios foram realizados em dois cenários: Teste 1 e Teste 2. Dois *switches* foram usados, um com baixa largura de banda (*Switch* 1, usado no Teste 1) e o outro com largura de banda alta (*Switch* 2, utilizado no Teste 2). A motivação para estes dois cenários foi identificar uma possível influência dos equipamentos de comutação no comportamento de saturação. Cada cenário era um ambiente controlado consistindo de três nós: Um atacante, um refletor e uma vítima. A Figura 5.3 ilustra a topologia de teste, onde o atacante corresponde a um *host* na camada intermediária, como na Figura 2.11. Os nós foram conectados a um *switch* por enlaces de 100 Mbps, em um único segmento de rede exclusivo. As configurações dos nós são mostradas na Tabela 5.6. Para cada cenário, O tráfego foi capturado em todos os três nós com um analisador de protocolo (Wireshark [Wireshark Foundation 2015]). Para efeitos destes teste, apenas o tráfego SNMP foi capturado e analisado.

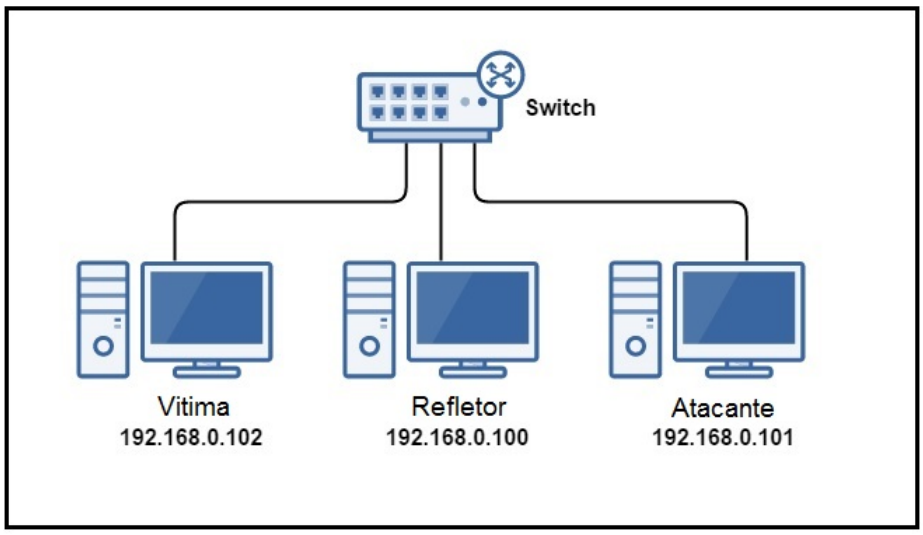

Figura 5.3: Topologia de Teste.

Tabela 5.6: Configuração do Ambiente.

Dispositivo	**Configuração**	**Endereço IP**
Atacante	MAC OSX, 2.3 GHz Intel Core i7, 16 GB DDR3	192.168.0.101
Refletor	Windows 8.1 64 bits, 1.6 GHz Intel Core i5, 4 GB DDR2	192.168.0.100
Vítima	Windows 7 64 bits, 3.4 GHz Intel Core i7, 8 GB DDR2	192.168.0.102
Switch 1	Multilaser E24 150 Mbps	N/A
Switch 2	Enterasys C-Series C5G124-48P2 1 Gbps	N/A

Os testes foram executados em rodadas, uma para cada nível de intensidade de ataque. Em cada rodada, o ataque foi executado durante 30s com o tráfego capturado em todos os três nós. Embora observado, o tráfego relacionado a outros protocolos (como ICMP) foi descartado e não analisado.

5.2.3 Resultados dos Testes

Os resultados dos testes são apresentados para cada cenário, com as Tabelas 5.7 a 5.9 correspondente ao Teste 1 e Tabelas 5.10 a 5.12 ao Teste 2. Os resultados para o fator de amplificação em ambos os cenários são apresentados na Tabela 5.13.

Tabela 5.7: Teste 1: Tráfego (packet/s).

Level	Atacante	Refletor		Vítima
	Egresso	Ingresso	Egresso	Ingresso
1	1	1	33	33
2	10	10	332	332
3	103	103	864	864
4	1021	1020	872	872
5	10317	8835	870	873
6	103331	32331	785	785
7	293869	61129	700	699
8	314122	60873	547	546

Tabela 5.8: Teste 1: Tráfego (Byte/s).

Level	Atacante	Refletor		Vítima
	Egresso	Ingresso	Egresso	Ingresso
1	82	82	50013	50013
2	816	819	502124	502124
3	8140	8140	1307921	1307921
4	80659	80548	1282255	1282255
5	815064	697994	1279351	1284195
6	8163136	2554131	1187994	1187994
7	23215614	4829191	1058325	1057670
8	24815635	4808938	805020	804460

Tabela 5.9: Teste 1: Tráfego (bit/s).

Level	Atacante	Refletor		Vítima
	Egresso	Ingresso	Egresso	Ingresso
1	653	653	400103	400103
2	6531	6552	4016993	4016993
3	65117	65117	10463369	10463369
4	645272	644387	10258037	10258037
5	6520513	5583952	10234809	10273561
6	65305087	20433045	9503950	9503950
7	185724913	38633528	8466598	8461363
8	198525083	38471504	6440158	6435678

Para o Teste 1, as Tabelas 5.7 a 5.9 mostram que o tráfego máximo (10,46 Mbps, linha 3, Tabela 5.9 e 1,3 MBps, linha 3, Tabela 5.8) atinge a vítima no nível 3, correspondente à saída máxima do refletor. A amplificação do refletor parece começar a saturar do nível 3 para 4 (de 864 pkt / s para 872 pkt / s, linhas 3 e 4, Tabela 5.7), enquanto o atacante satura do nível

6 para 7 (de 103 Kpkt / s para 293 Kpkt / s, Tabela 5.7), em termos de geração de *probes*. A capacidade do atacante (em termos de tráfego injetado) satura entre os níveis 4 e 5 (de 644 Kbps para 5,58 Mbps, linhas 4 e 5, Tabela 5.9 e de 80 KBps para 697 KBps, linhas 4 e 5, Tabela 5.8).

Tabela 5.10: Teste 2: Tráfego (packet/s).

Level	Atacante	Refletor		Vítima
	Egresso	Ingresso	Egresso	Ingresso
1	1	1	33	33
2	10	10	331	331
3	103	103	875	875
4	1031	1029	910	910
5	10331	8889	842	839
6	103333	44227	717	717
7	300173	70229	682	682
8	305976	70229	685	685

Tabela 5.11: Teste 2: Tráfego (Byte/s).

Level	Atacante	Refletor		Vítima
	Egresso	Ingresso	Egresso	Ingresso
1	79	79	49802	49802
2	790	790	499421	499421
3	8139	8139	1309128	1309128
4	81483	81304	1377979	1377979
5	816180	702296	1259074	1254143
6	8163333	3493969	1075626	1075626
7	23813690	5548151	1026244	1026244
8	24172125	5548164	1033519	1033519

Tabela 5.12: Teste 2: Tráfego (bit/s).

Level	Atacante	Refletor		Vítima
	Egresso	Ingresso	Egresso	Ingresso
1	632	632	398419	398419
2	6320	6320	3995371	3995371
3	65117	65117	10473024	10473024
4	651865	650433	11023839	11023839
5	6529444	5618374	10072593	10033144
6	65306666	27951758	8605009	8605009
7	190509525	44385212	8209957	8209957
8	193377000	44385317	8268154	8268154

Para o Teste 2, as Tabelas 5.10 a 5.12 mostram que o tráfego máximo (11,02 Mbps linha 3, Tabela 5.12 e 1,38 MBps, linha 3, Tabela 5.11) atinge a vítima no nível 4, correspondente à saída máxima do refletor. A amplificação do refletor parece começar a saturar do nível 3 para 4 (de 875 pkt / s para 910 pkt / s, linhas 3 E 4, Tabela 5.10), enquanto o atacante satura

desde o nível 6 até 7 (desde 300 Kpkt / s até 305 Kpkt / s, Tabela 5.10) Em termos de geração de sondas. A capacidade do atacante, em termos de tráfego injetado, satura entre os níveis 4 E 5 (de 650 Kbps para 5,61 Mbps, linhas 4 e 5, Tabela 5.12 e de 81 KBps para 702 KBps, linhas 4 E 5, Tabela 5.11).

A Tabela 5.13 mostra que para ambos os cenários, a amplificação máxima ocorre no nível 2, mas não se sustenta a partir desse nível. Os fatores de amplificação máximos observados para o Teste 1 são 613,12 vezes Bits e 32 vezes em pacotes; Enquanto que para o Teste 2, foi 632,18 vezes em bits e 33,11 vezes em pacotes. Essas taxas correspondem à eficiência máxima de ataque (isto é, o esforço de execução de ataque é mínimo quando comparado ao efeito sobre a vítima).

Tabela 5.13: Taxas de Amplificação.

Level	Teste 1		Teste 2	
	Bits	Packets	Bits	Packets
1	612.65	32.00	630.41	33.00
2	613.12	32.00	632.18	33.11
3	160.69	8.39	160.83	8.50
4	15.92	0.86	16.95	0.88
5	1.83	0.10	1.79	0.09
6	0.47	0.02	0.31	0.02
7	0.22	0.01	0.18	0.01
8	0.17	0.01	0.19	0.01

5.2.4 Análise dos Testes

Os resultados para o Teste 1 estão ilustrados nas Figuras 5.4 e 5.5, enquanto que as Figuras 5.6 e 5.7 mostram resultados para Teste 2. Para todos os gráficos, o eixo horizontal representa o nível de intensidade de ataque, enquanto os valores do eixo vertical representam cada taxa específica em escala logarítmica.

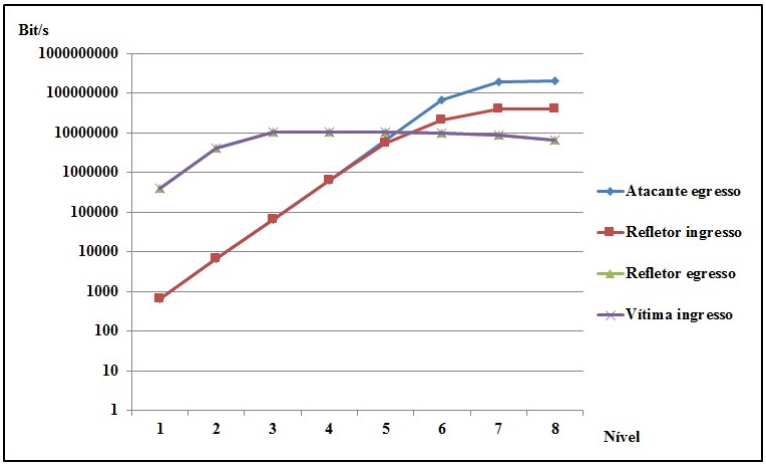

Figura 5.4: Teste 1: Taxas de Tráfego por Nível de Ataque (Bit/s).

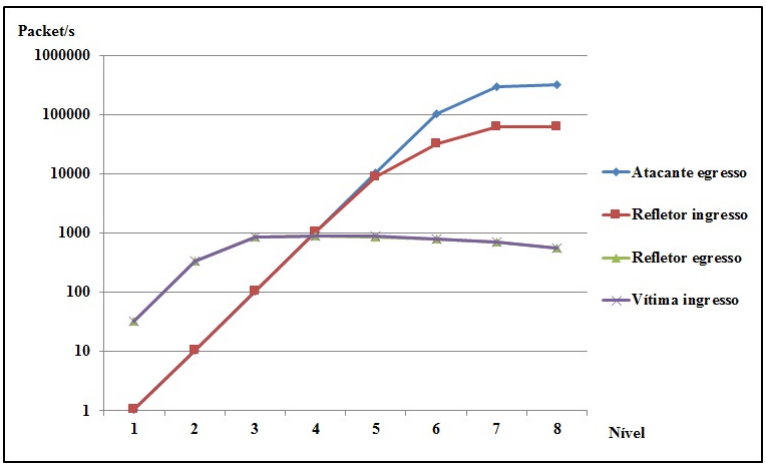

Figura 5.5: Teste 1: Taxas de Tráfego (Packet/s).

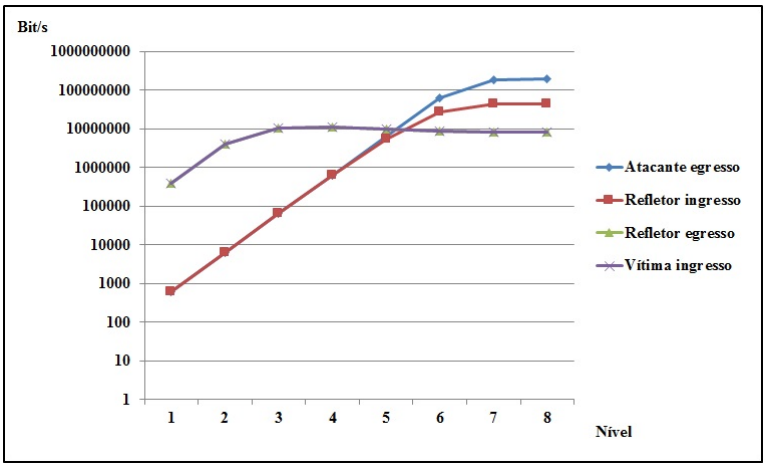

Figura 5.6: Teste 2: Taxas de Tráfego por Nível de Ataque (Bit/s).

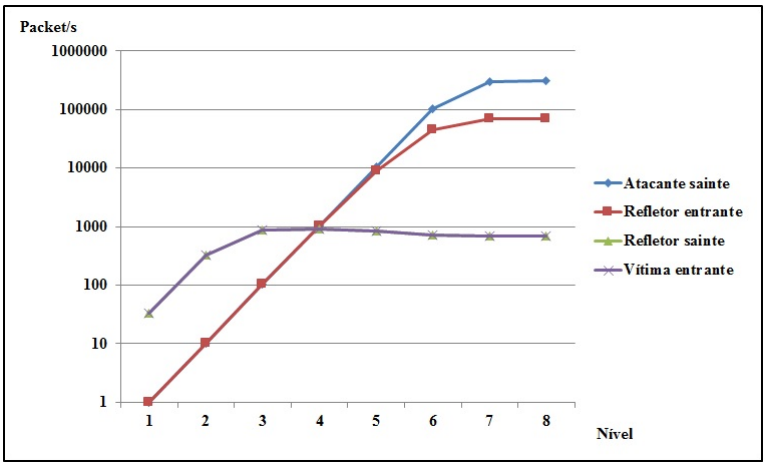

Figura 5.7: Test 2: Taxas de Tráfego (Packet/s).

- **Análise do atacante**:

Como mostrado, em ambos os cenários, o tráfego gerado pelo atacante satura em quase 200 Mbps (198 Mbps, linha 8, Tabela 5.9 e 193 Mbps, linha 8, Tabela 5.12). No entanto, a velocidade da linha é limitada a 100 Mbps. Esta aparente inconsistência deve-se ao fato do analisador de protocolos usado capturar os pacotes de saída antes da efetiva transmissão. Assim, este número não representa os bits efetivamente enviados, mas a capacidade de geração de tráfego, em bits, da ferramenta.

- **Análise do refletor**:
  De acordo com os gráficos (Figuras 5.4–5.7), em ambos os cenários, a saturação do desempenho do refletor aparece sob duas formas. Primeiro, na sua capacidade de gerar tráfego, a partir do nível 3. Isso também está evidente na sua capacidade de lidar com o tráfego de entrada, em torno do nível 5. Note que os gráficos mostram que o tráfego egresso do atacante e ingresso no refletor divergem no nível 5, mas a capacidade de geração de *probes* do atacante satura no nível 8. Assim, o ganho não é sustentável. De fato, a partir do nível 6, o comportamento de amplificação não está mais presente, uma vez que o refletor envia menos tráfego do que recebe. Isso ocorre porque o processamento *GetBulkRequest* não é instantâneo, pois requer a coleta de informações e montagem de datagrama. Conforme já mencionado, a configuração do refletor usada nestes testes é muito mais robusta do que a de um típico dispositivo de rede, IoT ou sensor. Mesmo supondo que esses dispositivos têm seus agentes otimizados para SNMP, ainda é razoável esperar que eles saturarem consideravelmente para níveis de tráfego bem mais baixos.

- **Análise da vítima**:
  O gráfico também demonstra que, em ambos os cenários, a vítima recebe tráfego do reflector na taxa em que é enviado. Para o sucesso do ataque DoS, outros refletores seriam necessários para incapacitar a vítima ou inundar seu enlace.

- **Análise do comportamento do** *switch*:
  Em ambos os cenários, o tráfego de entrada no refletor é menor que o tráfego de saída do atacante. Isso pode sugerir que o switch está saturado. De fato, isso motivou o uso de switches com capacidades distintas em cada rodada de teste. No entanto, como mostram os gráficos, o tráfego egresso do refletor e o ingresso na vítima coincidem em seus volumes, indicando que os *switches* não saturaram durante os testes, lembrando que as suas capacidades excedem com folga os volumes de tráfego observados. Resumindo, os resultados em ambos os testes são muito próximos, apesar das diferentes capacidades dos equipamentos.

- **Análise de ganho**:
  A Figura 5.8 mostra o fator de amplificação em termos de pacotes, γ_{pkt}, e bits γ_{bit}. Ambos mostram fatores de amplificação que claramente não são sustentáveis a partir do nível 2. A partir do nível 4, A amplificação de pacotes cessa, acontecendo o mesmo com a amplificação de bits a partir do nível 6.

Embora a negação de serviço completa não tenha sido alcançada na vítima, os testes mostram conclusivamente que a saturação do refletor ocorre a baixas taxas de injeção. Para que os atacantes alcancem a amplificação máxima sustentável, a geração de *probes* requer um certo cuidado sendo necessário conhecimento específico e detalhado não só sobre a vítima, mas também sobre os refletores.

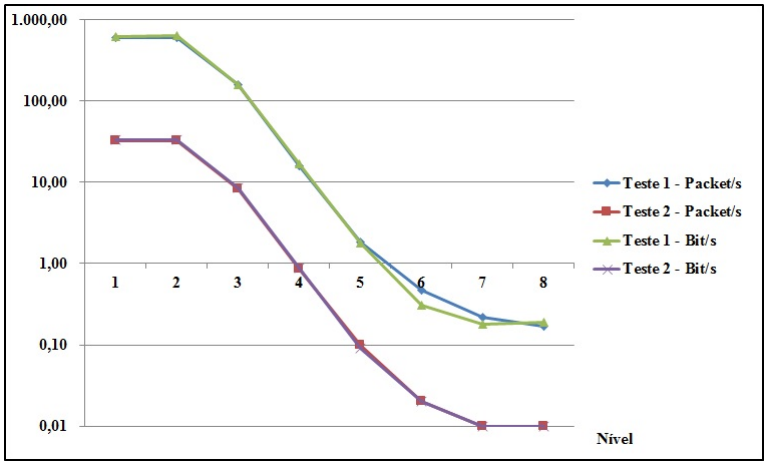

Figura 5.8: Amplificação no Refletor: fator em Bits, γ_{bit}, e Packets, γ_{pkt}, por Nível de Intensidade do Ataque.

5.2.5 Resumo da Aplicação de DOTA Voltada à Indisponibilidade

Os testes alcançaram seus objetivos e mostraram o comportamento de saturação no refletor e no atacante. Consequentemente, apesar de obter taxas de amplificação elevadas, estas não são sustentáveis mesmo para refletores com razoável poder de computação. Note-se que no refletor, a saturação ocorre a taxas de entrada relativamente baixas. O ataque é exequível e eficaz, mas precisa de uma execução precisa para se obter a máxima eficiência.

Este resultado mostra que a aplicação da metodologia DOTA voltada à indisponibilidade é adequada e eficaz. Os resultados obtidos nos testes revelam um quadro mais completo e aprofundado do ataque realizado. Do ponto de vista do alvo, apesar de não ter sofrido a negação de serviço de forma completa, pode-se estabelecer o grau de vulnerabilidade a que estaria exposto.

Por parte do refletor, que seria o papel de dispositivos IoT para um ataque deste tipo, a ocorrência clara de saturação leva a concluir que no caso de um ataque desse tipo a infraestrutura abusada corre o risco de sofrer negação de serviço antes mesmo do alvo. Assim, ainda que o ataque não atinja o alvo, teria impacto sobre os dispositivos IoT.

5.3 Resumo do Capítulo 5

Neste capítulo a metodologia foi aplicada em cenários voltados à intrusão, levando à obtenção de resultados que demonstram sua eficácia. A eventual aplicação da metodologia em um cenário distinto daqueles originalmente a motivaram, no caso auditoria, motivou a aplicação da mesma em uma aplicação voltada à indisponibilidade.

O cenário escolhido para a indisponibilidade foi a IoT, onde a metodologia foi adaptada e guiou as decisões quanto à definição do ataque, sua implementação, operacionalização e avaliação de resultados.

Capítulo 6

Conclusão

Conforme se viu no Capítulo 2 e nas Seções 3.1 e 3.2, a revisão bibliográfica demonstrou que existem metodologias de teste de penetração bem estabelecidas, mas que foram construídas principalmente em torno de procedimentos operacionais e conjuntos de ferramentas. Conforme mencionado anteriormente na seção 3.6, as metodologias mais abrangentes dependentes de conjuntos de ferramentas tinham restrições ao ciclo de vida que poderiam invalidar resultados a médio e longo prazo. Tomou-se então o caminho de desenvolver uma metodologia *pentest* focalizando as decisões para orientar o processo. A metodologia foi desenvolvida testando seus blocos de construção e, em seguida, rodadas completas em diferentes cenários foram executadas para avaliar sua eficácia na geração de resultados. Na sequência, essa metodologia foi adaptada para se concentrar na disponibilidade e, posteriormente nas principais questões de interesse neste trabalho.

Tradicionalmente, as metodologias de testes de penetração são voltadas à avaliação da segurança de sistemas com relação à invasão. Aqui, entende-se por invasão as atividades referentes a contornar sistemas de controle de acesso e privilégio tendo como objetivo final a obtenção de controle total sobre os sistemas invadidos. Assim, a metodologia foi testada no contexto de invasão, que é aquele que foi originalmente projetada.

DOTA, a metodologia aqui desenvolvida, apesar de seu caráter prescritivo, também engloba o registro e descrição das práticas utilizadas, chegando a uma metodologia aplicável, próxima da cultura já existente, e gerando resultados úteis. A abordagem de sua construção foi a de testar as fases da metodologia isoladamente, visando ajustes e adequações.

O principal objetivo deste trabalho, desenvolver uma metodologia de *pentest* que fosse mais perene, tendo por base a modelagem de critérios e decisões, foi atingido. A modelagem da execução dos testes e ataques com o ciclo OADA possibilita modelá-los sem que seja necessário recorrer à descrição de procedimentos operacionais ou instrução de uso de ferramentas. A metodologia aqui apresentada foi submetida a baterias de testes que demonstram que sua aplicação contribui na obtenção de resultados mais abrangentes, mais aprofundados e mais precisos.

A execução do ciclo OADA deixa evidente o ganho em se explicitar o processo de tomada de decisão, não só nas fases preparatórias, mas principalmente quando da execução dos testes. Assim, o ciclo é a peça fundamental para desacoplar a metodologia de procedimentos operacionais e conjuntos de ferramentas e manter o foco nos critérios e processos de decisão.

Além dos testes voltados a intrusão, houve a oportunidade de avaliar a flexibilidade de aplicação da metodologia. No caso, tratava-se de uma auditoria de segurança em um sistema de informação que devia ser avaliado antes de ser colocado em produção. A principal diferença em termos de aplicação da metodologia ocorreu na Fase 2, onde se teve acesso a toda documentação de desenvolvimento do sistema em questão. O efeito de tal diferença foi que, dado o maior volume de informações relevantes acerca do sistema anterior às fases de planejamento e execução dos testes, houve uma amplificação dos resultados, *i.e.* obteve-se resultados mais abrangentes e aprofundados do que normalmente se obtinha nas condições normais de aplicação da metodologia.

Desta forma, ainda que o volume de testes seja relativamente pequeno, os resultados não desabonam afirmar que a metodologia atinge os objetivos de repetibilidade e explicitação da tomada de decisão, além de também apresentar flexibilidade de aplicação. Esta é a principal limitação dos resultados de aplicação da metodologia: o fato de se ter um pequeno número de relatos completos de sua aplicação em condições controladas. Além das aplicações voltadas à intrusão nos cenários descritos anteriormente, há relatos informais de sua aplicação com sucesso. Entretanto, não houve o acompanhamento, controle e registro necessários para inclusão dos resultados.

Deve-se ressaltar que a aplicação de uma rodada completa da metodologia em um ambiente de produção requer, além de um período relativamente longo para execução dos testes, toda uma série de arranjos e acertos para obter o consentimento para iniciar os testes no ambiente alvo. Outro ponto relevante diz respeito à equipe que realiza os testes, cujo recrutamento e treinamento na metodologia também impõem limitadores na realização de experimentos.

Com respeito à aplicação da metodologia ao contexto de DDoS, especificamente AR-DDoS, em ambiente de IoT, foram demonstrados aspectos da metodologia que não estavam previstos no seu *design* inicial. A abordagem de execução dos testes sob a ótica do atacante possibilitou a obtenção de resultados mais abrangentes para a caracterização dos atores participantes em um ataque desse tipo.

Os testes voltados à indisponibilidade produziram resultados que indicam que a saturação foi atingida nos níveis de refletor e atacante. Para o refletor, a velocidade da linha não é alcançada porém o processamento dos *probes* para amplificação requer mais esforço que a geração dos *probes*. Assim, a saturação do refletor é devido ao esforço de processamento adicional.

Os resultados também demonstraram que AR-DDoS sobre IoT é viável, e pode atingir altas taxas de amplificação. No entanto, estas altas taxas de amplificação não são sustentáveis, mesmo para os refletores com potência computacional relevante, uma vez que a saturação

do refletor ocorre a taxas de entrada relativamente baixas. Assim, se os ataques não forem cuidadosamente conduzidos, a infraestrutura do refletor abusada satura e a indisponibilidade na vítima não é totalmente alcançada. Como consequência, o ataque atingiria o refletor de forma mais impactante que a vítima. Mesmo assim, uma vítima ainda poderia ser incapacitada se vários refletores coordenarem o seu tráfego. Do ponto de vista do atacante, isso pode ser compensados por taxas de injeção menores, demandando mais refletores, o que requer mais infraestrutura de refletores e melhor gerenciamento de execução de ataques.

Do ponto de vista das redes IoT e de sensores, ficou demonstrado que elas oferecem uma superfície de ataque muito expandida, especificamente para o tipo de ataque descrito. A partir dos resultados dos testes, fica claro que, embora a alegação seja realista, os resultados de uma tentativa de abuso contra IoT e dispositivos de rede de sensores em um AR-DDoS têm grande chance de ser confinado à sua própria infraestrutura. Em todo caso, fica claro que a infraestrutura e os dispositivos de IoT serão severamente atingidos, pelo menos tanto quanto as vítimas.

No contexto dos refletores serem sensores, dispositivos IoT ou elementos de sua infraestrutura de rede, devido à modesta capacidade computacional e de largura de banda, pode-se prever que esses refletores certamente devem saturar, sem necessariamente incapacitar os alvos finais. O número de refletores envolvidos para ataques bem sucedidos será determinado pela contribuição residual que cada dispositivo fornece em termos de tráfego dirigido às vítimas finais. Conforme observado, para aproveitar a máxima eficiência, um atacante deve injetar *probes* a taxas baixas. Por si só, isso significa menos esforço do atacante, mas exige um melhor gerenciamento e condução do ataque, além de melhor ajuste fino de taxas de injeção e número de refletores, juntamente com mais conhecimento específico sobre a infraestrutura dos refletores a serem abusados.

Embora os testes tenham sido executados sobre uma implementação específica de ataques AR-DDoS, o SNMP foi escolhido uma vez que é utilizado na gerência de redes IoT e tem uma *footprint* relativamente pequena. Os componentes que levaram à saturação do refletor estão presentes na maioria dos protocolos candidatos a abuso. Eespecificamente: custos computacionais para processamento de pedidos não desprezíveis e baixo poder computacional. Se o segundo não for suficiente para satisfazer o primeiro, certamente ocorrerá saturação. Portanto, não é exagerado dizer que as conclusões aqui apresentadas são generalizáveis para AR-DDoS baseado em outros protocolos.

Como base para comparação, Pacheco et al. [Pacheco et al. 2016] avaliam o efeito de um ataque DDoS sobre uma Wireless Sensor Network (WSN) montada com Zigbee e 6lowPAN usando um ambiente de simulação de rede. Seus resultados também indicam saturação que ocorre em taxas de entrada de *probes* muito baixas. No entanto, a saturação está presente apenas em termos de capacidade de rede, mas não há caracterização de possível saturação devido à carga de processamento. Sgouras et ai. [**?**] também avaliam o impacto dos ataques DoS / DDoS sobre *smart grids*, onde a saturação também está presente. No entanto, os dispositivos não fazem parte do ataque como vítimas, o que está fora do escopo deste

trabalho.

Os resultados aqui apresentados foram obtidos pela aplicação de uma metodologia suficientemente flexível para acomodar testes de avaliação de ameaça de disponibilidade. A abordagem é inovadora ao contemplar a visão de um atacante, trazendo não apenas outra perspectiva, mas um quadro mais completo com respeito ao ataque escolhido. Assim, um ataque DDoS simples, fácil de implementar e executar como AR-DDoS é dissecado em uma série de experimentos de modo a caracterizar sua dinâmica de tráfego.

Conforme apresentado na Seção 2.4.1, os incidentes envolvendo dispositivos IoT e infraestrutura já são realidade, com ataques que atingem capacidades cada vez maiores, mas amplificação não é comumente usada. Atualmente, os incidentes exploram vulnerabilidades simples no software do dispositivo. Por isso é mais atraente para os atacantes refatorar *malware* de *botnets* já conhecidos para atingir um grande número de novas vítimas. Assim, é válido argumentar que os esforços para melhorar a segurança em IoT deve ser centrada na correção de falhas no software dos dispositivos. Contudo, isso não será suficiente para evitar ataques por reflexão. Note que se as *botnets* reduzirem, ataques de amplificação sobre IoT certamente se tornarão mais frequentes.

Quanto à prevenção de AR-DDoS, as melhores práticas atuais ainda são válidas. Especificamente no caso de IoT, segregação de rede, se possível e aplicável, e colocação atrás de *Network Address Translation* (NAT) combinado com filtragem *anti-spoofing* são medidas simples, mas eficientes. No entanto, eles exigem uma melhor gerência da rede e controle de configuração.

Os resultados apresentados mostram que o AR-DDoS é uma técnica poderosa que aumenta a capacidade de ataque. Apesar de seus efeitos potencialmente esmagadores, para um atacante aproveitar ao máximo, a execução cuidadosamente planejada e executada é um requisito essencial, contrastando com a força bruta pura como em outras formas de DDoS.

A infraestrutura de IoT oferece uma superfície de ataque enorme em termos de AR-DDoS que ainda não foi amplamente explorada em ataques. Pela saturação identificada nos testes, espera-se que os dispositivos IoT quando usados como refletores sejam atingidos pelo menos tão severamente quanto as vítimas. Felizmente, as melhores práticas atuais de prevenção estão disponíveis e podem ser usadas para mitigar alguns dos ataques. Em suma, a ameaça é real, mas há maneiras de tratá-la. Entretanto, requer esforços no gerenciamento e aprimoramento do software para IoT.

A principal limitação dos resultados da aplicação de DOTA a ataques de indisponibilidade está no fato de que a abordagem utilizada é inovadora não só na sua aplicação à IoT, mas para a indisponibilidade em outros cenários em geral. A metodologia, embora nova já está madura o suficiente para gerar resultados práticos úteis, mas ainda precisa ser aplicada em diferentes contextos. De qualquer forma, a metodologia quando utilizada corretamente produz resultados verificáveis e aplicáveis em ambientes reais.

Como uma síntese final, os quadros comparativos de classificação das metodologias es-

tudadas são representados incluindo agora DOTA nas Tabelas a seguir.

Tabela 6.1: Quadro Comparativo das Metodologias com DOTA - Finalidade

Metodologia	**Finalidade**
NIST	gerenciamento e análise de risco
OSSTMM	auditoria de segurança
OWASP	desenvolvimento seguro de aplicações web
PTES	testes de penetração
PCI-DSS	auditoria de risco
SANS	testes de penetração
DOTA	testes de penetração

DOTA tem por finalidade a sistematização dos testes de penetração, segundo a Tabela 6.1 e escopo geral, como na Tabela 6.2.

Tabela 6.2: Quadro Comparativo das Metodologias com DOTA- Escopo

Metodologia	**Escopo**
NIST	geral
OSSTMM	geral
OWASP	aplicações web
PTES	geral
PCI-DSS	aplicações PIN
SANS	geral
DOTA	geral

Quanto ao processo em si, DOTA tem seu processo definido e detalhado, porém sem excessiva complexidade, como na Tabela 6.3.

Tabela 6.3: Quadro Comparativo das Metodologias com DOTA - Processo

Metodologia	Processo		
	Definido	Detalhado	Complexo
NIST	✓	✓	✓
OSSTMM	✓	✓	✓
OWASP	N/A	N/A	N/A
PTES	✓		
PCI-DSS	✓		
SANS	✓		
DOTA	✓	✓	

Quanto à descrição do processo, Tabela 6.4 DOTA descreve procedimentos, ainda que em alto nível. Assim DOTA adere a uma descrição operacional, porém sem adotar um conjunto de ferramentas. A porção crítica referente ao ciclo de execução de ataque em DOTA é modelada por um ciclo geral de tomada de decisão, enquanto as outras metodologias estudas ou não apresentam uma modelagem ou modelam segundo o ciclo de ataque de *malware*, que é o caso do NIST.

Tabela 6.4: Quadro Comparativo das Metodologias com DOTA - Descrição do Processo

Metodologia	Descrição do Processo		
	Operacional	Ferramentas	Ataque
NIST	✓	✓	*malware*
OSSTMM	✓	✓	
OWASP	N/A	N/A	N/A
PTES	✓	✓	
PCI-DSS	✓	✓	
SANS	✓	✓	
DOTA	✓		geral

Em DOTA o processo está organizado de forma consistente, *i.e.* o encadeamento de fases e atividades não apresenta incoerências. Os conceitos, em especial critérios e procedimentos estão claramente definidos e separados; e o foco do desenvolvimento de DOTA é o fluxo de decisão e os critérios que devem guiá-la, conforme a Tabela 6.5.

Tabela 6.5: Quadro Comparativo das Metodologias com DOTA - Organização do Processo

Metodologia	Organização do Processo		
	Consistente	Separação	Decisão
NIST	✓	✓	
OSSTMM	✓	✓	
OWASP	N/A	N/A	N/A
PTES	✓		
PCI-DSS			
SANS	✓		
DOTA	✓	✓	✓

Por fim, enquanto as metodologias estudadas têm sua aplicação voltada à intrusão, DOTA é aplicável tanto à intrusão como à indisponibilidade (Tabela 6.6)

Tabela 6.6: Quadro Comparativo das Metodologias com DOTA - Aplicabilidade

Metodologia	Aplicabilidade	
	Intrusão	Indisponibilidade
NIST	✓	
OSSTMM	✓	
OWASP	N/A	N/A
PTES	✓	
PCI-DSS	✓	
SANS	✓	
DOTA	✓	✓

6.1 Trabalhos Futuros

As linhas de trabalhos futuros para a metodologia vão em duas vertentes. A primeira é o aprimoramento da própria metodologia, com a aplicação em novos cenários e situações

visando gerar uma base de resultados que possibilite identificar e quantificar o esforço e o tempo necessários para obtenção de resultados. Assim, espera-se poder se dimensionar equipes e prazos de forma mais precisa.

Conforme indicado, a metodologia utilizada ainda está evoluindo. Experimentos, não apenas no âmbito da avaliação da ameaça à disponibilidade, mas avaliação ativa de vulnerabilidades em geral, para refinar suas fases e atividades e comparar com outras abordagens.

A outra linha está relacionada a explorar os aspectos de flexibilidade da metodologia, explorando situações que não estavam nos objetivos iniciais de sua formulação. Assim, é uma direção imediata a combinação da aplicação da metodologia com AR-DDoS em IoT, na forma de estudos comparativos de ameaças de indisponibilidade representadas por outros protocolos, como SSDP e CoAP.

Referências Bibliográficas

[ABNT 2005]ABNT, N. Iec 27002: 2005. *Código de Prática para a Gestão da Segurança da Informação. Associação Brasileira de Normas Técnicas*, 2005.

[Ali 2007]ALI, F. *IP Spoofing*. 2007. `http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_10-4/104_ip-spoofing.html`.

[Alisherov e Sattarova 2009]ALISHEROV, F.; SATTAROVA, F. Methodology for penetration testing. *International Journal of Grid and Distributed Computing*, p. 43–50, 2009.

[Anonymous 2011]ANONYMOUS. *Penetration testing*. Clifton Park, NY, USA: Course Technology, Cengage Learning, 2011. ISBN 1-4354-8366-9 (vol. 1 : paperback), 1-4354-8367-7 (vol. 2 : paperback), 1-4354-8368-5 (vol. 3 : paperback), 1-4354-8369-3 (vol. 4 : paperback), 1-4354-8370-7 (vol. 5 : paperback).

[Arış, Oktuğ e Yalçın 2015]ARış, A.; OKTUğ, S. F.; YALçıN, S. B. . Internet-of-things security: Denial of service attacks. In: *2015 23nd Signal Processing and Communications Applications Conference (SIU)*. New York, NY, USA: IEEE, 2015. p. 903–906. ISSN 2165-0608.

[Arukonda e Sinha 2015]ARUKONDA, S.; SINHA, S. The innocent perpetrators: reflectors and reflection attacks. *Advances in Computer Science: an International Journal*, v. 4, n. 1, p. 94–98, 2015.

[Ashton 2009]ASHTON, K. That 'internet of things' thing. *RFiD Journal*, v. 22, n. 7, p. 97–114, 2009.

[Attanasio, Markstein e Phillips 1976]ATTANASIO, C. R.; MARKSTEIN, P. W.; PHILLIPS, R. J. Penetrating an operating system: a study of vm/370 integrity. *IBM Systems Journal*, v. 15, n. 1, p. 102–116, 1976. ISSN 0018-8670.

[Atzori, Iera e Morabito 2010]ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. *Computer networks*, Elsevier, v. 54, n. 15, p. 2787–2805, 2010.

[Baloch 2014]BALOCH, R. *Ethical hacking and penetration testing guide*. 2000 N.W. Corporate Blvd., Boca Raton, FL 33431-9868, USA: CRC Press, 2014. ISBN 1-4822-3161-1 (paperback).

[Basta 2013]BASTA, A. *Computer security and penetration testing*. Second. Boston, MA: Course Technology, Cengage Learning, 2013. ISBN 0-8400-2093-7.

[Bergman et al. 2013]BERGMAN, N. et al. *Hacking exposed: mobile security secrets and solutions*. New York, NY, USA: McGraw-Hill, 2013. xxvii + 289 p. ISBN 0-07-181701-8 (paperback).

[Bishop 1986]BISHOP, M. Analyzing the security of an existing computer system. In: IEEE COMPUTER SOCIETY PRESS. *Proceedings of 1986 ACM Fall joint computer conference*. New York, NY, USA, 1986. p. 1115–1119.

[Blumenthal e Wijnen 2002]BLUMENTHAL, U.; WIJNEN, B. *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*. IETF, dez. 2002. RFC 3414. (Request for Comments, 3414). Disponível em: <http://www.ietf.org/rfc/rfc3414.txt>.

[Borgia 2014]BORGIA, E. The internet of things vision: Key features, applications and open issues. *Computer Communications*, Elsevier, v. 54, p. 1–31, 2014.

[Botella et al. 2014]BOTELLA, J. et al. Risk-based vulnerability testing using security test patterns. In: SPRINGER. *International Symposium On Leveraging Applications of Formal Methods, Verification and Validation*. "Berlin, Heidelberg", 2014. p. 337–352.

[Boyd 1976]BOYD, J. R. *Destruction and Creation*. Fort Leavenworth, KS, USA, 1976. "Notes from talk presented at U.S. Army Command and General Staff College em 3 Set 1976".

[Boyd 1995]BOYD, J. R. *The Essence of Winning and Losing*. Fort Leavenworth, KS, USA, 1995. "Slide set presented in 28 Jun 1995".

[Bright 2013]BRIGHT, P. *Spamhaus DDoS grows to Internet-threatening size*. USA, 2013. In `http://arstechnica.com/security/2013/03/spamhaus-ddos-grows-to-internetthreatening-size/`. Accessed in May 2016.

[Broad e Bindner 2014]BROAD, J.; BINDNER, A. *Hacking with Kali: practical penetration testing techniques*. First. Amsterdam, The Netherlands: Syngress, an imprint of Elsevier, 2014. ix + 227 p. ISBN 0-12-407749-8 (alkaline paper).

[Case et al. 1990]CASE, J. et al. *Simple Network Management Protocol (SNMP)*. IETF, maio 1990. RFC 1157 (Historic). (Request for Comments, 1157). Disponível em: <http://www.ietf.org/rfc/rfc1157.txt>.

[Case et al. 1996]CASE, J. et al. *Introduction to Community-based SNMPv2*. IETF, jan. 1996. RFC 1901 (Historic). (Request for Comments, 1901). Disponível em: <http://www.ietf.org/rfc/rfc1901.txt>.

[Cox 2013]COX, R. *5 Notorious DDoS Attacks in 2013 : Big Problem for The Internet of Things*. Palo Alto, CA, USA, 2013. In http://searchsecurity.techtarget.com/news/4500246858/2015-DDoS-attacks-on-the-rise-attackers-shift-tactics. Accessed in May 2016.

[Cvitić, Vujić e Husnjak 2016]CVITIĆ, I.; VUJIĆ, M.; HUSNJAK, S. Classification of security risks in the iot environment. In: *26TH DAAAM INTERNATIONAL SYMPOSIUM ON INTELLIGENT MANUFACTURING AND AUTOMATION*. Wolkersdorf, Austria: DAAAM, 2016.

[Damon et al. 2012]DAMON, E. et al. Hands-on denial of service lab exercises using slowloris and rudy. In: ACM. *proceedings of the 2012 information security curriculum development conference*. New York, NY, USA, 2012. p. 21–29.

[DeFino 2010]DEFINO, S. *Official certified ethical hacker review guide*. Boston, MA, USA: Course Technology, Cengage Learning, 2010. xxii + 361 p. ISBN 1-4354-8853-9 (paperback).

[Deming 1986]DEMING, W. E. Out of crisis, centre for advanced engineering study. *Massachusetts Institute of Technology*, Cambridge, MA, USA, 1986.

[DoD 1985]DOD. *5200.28-STD Trusted Computer System Evaluation Criteria*. Washington, DC, USA, December 1985.

[Elkhodr, Shahrestani e Cheung 2016]ELKHODR, M.; SHAHRESTANI, S.; CHEUNG, H. The internet of things: New interoperability, management and security challenges. *arXiv preprint arXiv:1604.04824*, 2016.

[Engebretson 2013]ENGEBRETSON, P. P. H. (Ed.). *The basics of hacking and penetration testing: ethical hacking and penetration testing made easy*. Second. Amsterdam, The Netherlands: Syngress, an imprint of Elsevier, 2013. xviii + 204 p. ISBN 0-12-411644-2 (paperback).

[Engebretson e Broad, CISSP. 2011]ENGEBRETSON, P. P. H.; Broad, CISSP., J. (Ed.). *The basics of hacking and penetration testing: ethical hacking and penetration testing made easy*. Rockland, MA, USA: Syngress Publishing, Inc., 2011. xvii + 159 p. (Syngress the basics). ISBN 1-59749-655-3.

[ENISA 2011]ENISA. *Cyber Exercises*. ENISA, Crete, Greece, 2011. In: https://www.enisa.europa.eu/topics/cyber-exercises/trainings/cyber_exercises. Accessed Oct, 2016.

[FIPS 2004]FIPS. Pub 199. *Standards for Security Categorization of Federal Information and Information Systems*, v. 2, 2004.

[FIPS 2006]FIPS. Pub 200. *Minimum Security Requirements for Federal Information and Information Systems*, v. 2, 2006.

[Forum]FORUM, U. UPnP Device Architecture version 1.0. ISSN 2070-1721. Disponível em: <http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0.pdf>.

[Goodin 2016]GOODIN, D. *Record-breaking DDoS reportedly delivered by >145k hacked cameras*. USA, 2016. In `http://arstechnica.com/security/2016/09/botnet-of-145k-cameras-reportedly-deliver-internets-biggest-ddos-ever/`. Accessed in Oct. 2016.

[Großmann et al. 2014]GROSSMANN, J. et al. Combining risk analysis and security testing. In: ____. *Leveraging Applications of Formal Methods, Verification and Validation. Specialized Techniques and Applications: 6th International Symposium, ISoLA 2014, Imperial, Corfu, Greece, October 8-11, 2014, Proceedings, Part II*. Berlin, Heidelberg: Springer, 2014. p. 322–336. ISBN 978-3-662-45231-8. Disponível em: <"http://dx.doi.org/10.1007/978-3-662-45231-8_23">.

[Hamlet 1977]HAMLET, R. G. Testing programs with the aid of a compiler. *IEEE Transactions on Software engineering*, IEEE, New York, NY, USA, n. 4, p. 279–290, 1977.

[Herzog 2015]HERZOG, P. *Open Source Security Testing Methodology Manual (OSSTMM) - v4*. Cardedeu, Spain, 2015. In: `https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf`. Accessed Aug, 2016.

[Hu 2016]HU, F. *Security and Privacy in Internet of Things (IoTs): Models, Algorithms, and Implementations*. Boca Raton, FL, USA: CRC Press, 2016.

[IETF]IETF. The Constrained Application Protocol (CoAP). ISSN 2070-1721. Disponível em: <https://tools.ietf.org/html/rfc7252>.

[Infosec Institute 2014]Infosec Institute. *Conducting a Penetration Test on an Organization*. Madison, WI, USA, 2014. In: `http://www.sans.org/reading-room/whitepapers/auditing/conducting-penetration-test-organization-67.`. Accessed Aug, 2016.

[Jackson 2010]JACKSON, C. *Network security auditing*. Indianapolis, IN, USA: Cisco Press, 2010. xxiv + 488 p. (Cisco Press networking technology series). ISBN 1-58705-352-7 (paperback).

[Jackson 2013]JACKSON, W. *How hackers can turn the Internet of Things into a weapon*. McLean, VA, USA, 2013. In `https://gcn.com/blogs/cybereye/2013/05/how-hackers-turn-internet-of-things-into-weapon.aspx`. Accessed in May 2016.

[Jason 2014]JASON, K. *Cyber Exercise Playbook*. The MITRE Corporation, Bedford, MA, USA, 2014.

[Kenney 1996]KENNEY, M. *Ping of Death*. Palo Alto, CA, USA, 1996. In = "http://insecure.org/sploits/ping-o-death.html". Accessed in Oct 2014.

[Kim et al. 2013]KIM, G. et al. The design of vulnerability management system. *International Journal of Computer Science and Network Security (IJCSNS)*, International Journal of Computer Science and Network Security, v. 13, n. 4, p. 19, 2013.

[Kumar, Vealey e Srivastava 2016]KUMAR, S. A.; VEALEY, T.; SRIVASTAVA, H. Security in internet of things: Challenges, solutions and future directions. In: IEEE. *System Sciences (HICSS), 2016 49th Hawaii International Conference on*. New York, NY, USA, 2016. p. 5772–5781.

[Linde 1975]LINDE, R. Operating system penetration. *Proc. AFIPS 1975 Natl. Computer Conf.*, v. 44, p. 361–368, 1975.

[Liu et al. 2012]LIU, B. et al. Software vulnerability discovery techniques: A survey. In: IEEE. *Multimedia Information Networking and Security (MINES), 2012 Fourth International Conference on*. New York, NY, USA, 2012. p. 152–156.

[Mainka, Somorovsky e Schwenk 2012]MAINKA, C.; SOMOROVSKY, J.; SCHWENK, J. Penetration testing tool for web services security. In: IEEE. *Services (SERVICES), 2012 IEEE Eighth World Congress on*. New York, NY, USA, 2012. p. 163–170.

[Marback et al. 2013]MARBACK, A. et al. A threat model-based approach to security testing. *Software: Practice and Experience*, v. 43, n. 2, p. 241–258, 2013. ISSN 1097-024X. Disponível em: <http://dx.doi.org/10.1002/spe.2111>.

[McDowell 2009]MCDOWELL, M. *Understanding Denial-of-Service Attacks*. Washington, DC, USA, 2009.

[MI5 2016]MI5. *Cyber*. London, UK, 2016. InfoSec Reading Room. In: `https://www.sans.org/reading-room/whitepapers/analyst/penetration-testing-assessing-security-attackers-34635`. Accessed Mai, 2016.

[Miura-Ko e Bambos 2007]MIURA-KO, R. A.; BAMBOS, N. Securerank: A risk-based vulnerability management scheme for computing infrastructures. In: IEEE. *Communications, 2007. ICC'07. IEEE International Conference on*. New York, NY, USA, 2007. p. 1455–1460.

[Musa 1975]MUSA, J. D. A theory of software reliability and its application. *IEEE transactions on software engineering*, IEEE, New York, NY, USA, n. 3, p. 312–327, 1975.

[Nagpal et al. 2015]NAGPAL, B. et al. DDoS tools: Classification, analysis and comparison. In: *Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on*. New York, NY, USA: IEEE, 2015. p. 342–346.

[Neely, Hamerstone e Sanyk 2013]NEELY, M.; HAMERSTONE, A.; SANYK, C. *Wireless reconnaissance in penetration testing*. Amsterdam, The Netherlands: Elsevier/Syngress, 2013. xvi + 166 p. ISBN 1-59749-731-2.

[Neugent et al. 1985]NEUGENT, W. et al. Technology assessment: Methods for measuring the level of computer security. *NIST Special Publication SP-500-133*, Gaithersburg, MD, USA, 1985.

[Neumann 1977]NEUMANN, P. Computer system security evaluation. *Proc. AFIPS 1977 Natl. Computer Conf.*, v. 46, p. 1087–1095, 1977.

[NIST 1980]NIST. *FIPS PUB 73, Guidelines for security of computer applications*. Gaithersburg, MD, USA, 1980.

[NIST 2011]NIST. Managing information security risk–organization, mission, and information system view. national institute of standards and technology. *Special Publication 800-39*, 2011.

[NIST 2014]NIST. Guide for applying the risk management framework to federal information systems: A security life cycle approach. *Special Publication 800-37, Revision 1*, 2014.

[NIST 2014]NIST. Guide for assessing the security controls in federal information systems and organizations, building effective security assessment plans. *Special Publication 800-53A*, 2014.

[NIST 2015]NIST. Recommended security controls for federal information systems. *Special Publication 800-53 Rev. 4*, 2015.

[Nyanchama 2005]NYANCHAMA, M. Enterprise vulnerability management and its role in information security management. *Information Systems Security*, v. 14, n. 3, p. 29–56, 2005. Disponível em: <http://dx.doi.org/10.1201/1086.1065898X/45390.14.3.20050701/89149.6>.

[Open SNMP Scanning Project 2016]OPEN SNMP Scanning Project. USA, 2016. In "`https://snmpscan.shadowserver.org`. Accessed in Jan,2016.

[O'Reilly 2009]O'REILLY, P. National vulnerability database (nvd). 2009.

[OWASP 2014]OWASP. *OWASP Testing Guide*. Bel Air, MD, USA, 2014. In: `https://www.owasp.org/index.php/OWASP_Testing_Guide_v4_Table_of_Contents`. Accessed Aug, 2015.

[Pa et al. 2016]PA, Y. M. P. et al. Iotpot: A novel honeypot for revealing current iot threats. *Journal of Information Processing*, v. 24, n. 3, p. 522–533, 2016.

[Pacheco et al. 2016]PACHECO, L. A. B. et al. Evaluation of distributed denial of service threat in the internet of things. In: IEEE. *Network Computing and Applications (NCA), 2016 IEEE 15th International Symposium on*. New York, NY, USA, 2016. p. 89–92.

[Paxson 2001]PAXSON, V. *An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks*. 2001. AT&T Center for Internet Research at ICSI.

[PCI 2015]PCI. *PCI Data Security Standard (PCI DSS), Information Supplement: Penetration Testing Guidance, Version: 1.0*. Wakefield, MA, USA, 2015. In: `https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf`. Accessed Aug, 2016.

[Pfleeger, Pfleeger e Theofanos 1989]PFLEEGER, C. P.; PFLEEGER, S. L.; THEOFANOS, M. F. A methodology for penetration testing. *Computers & Security*, Elsevier, v. 8, n. 7, p. 613–620, 1989.

[Polstra e Ramachandran 2015]POLSTRA, P.; RAMACHANDRAN, V. (Ed.). *Hacking and penetration testing with low power devices*. Amsterdam, The Netherlands: Elsevier, 2015. xv + 243 p. ISBN 0-12-800751-6, 0-12-800824-5 (e-book), 1-322-09763-1 (e-book). Disponível em: <http://www.sciencedirect.com/science/book/9780128007518>.

[Pras et al. 2016]PRAS, A. et al. DDoS 3.0-How terrorists bring down the Internet. In: *Measurement, Modelling and Evaluation of Dependable Computer and Communication Systems*. Berlin, Heidelberg: Springer, 2016. p. 1–4.

[Prince 2013]PRINCE, M. *The DDoS That Knocked Spamhaus Offline (And How We Mitigated It)*. USA, 2013. In `https://blog.cloudflare.com/the-ddos-that-knocked-spamhaus-offline-and-ho/`. Accessed in May 2016.

[Prolexic 2015]PROLEXIC. *Threat Advisory: SNMP Reflection DDoS Attacks*. Hollywood, FL, USA, 2015. In `https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/snmp-reflector-attacks-threat-advisory.pdf`. Accessed in May 2015.

[PTES 2009]PTES. *PTES: Penetration Testing Execution Standard*. USA, 2009. In: `http://www.pentest-standard.org`. Accessed Aug, 2014.

[Rossow 2014]ROSSOW, C. Amplification Hell: Revisiting network protocols for DDoS abuse. In: *21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, February 23-26, 2014*. The Internet Society, 2014. Disponível

em: <http://www.internetsociety.org/doc/amplification-hell-revisiting-network-protocols-ddos-abuse>.

[Salas e Martins 2014]SALAS, M.; MARTINS, E. Security testing methodology for vulnerabilities detection of xss in web services and ws-security. *Electronic Notes in Theoretical Computer Science*, v. 302, p. 133 – 154, 2014. ISSN 1571-0661. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1571066114000255>.

[SANS Inst. 2002]SANS Inst. *Conducting a Penetration Test on an Organization*. Bethesda, MD, USA, 2002. Security Reading Room. In: http://resources.infosecinstitute.com/penetration-testing-methodology-web-applications/. Accessed Aug, 2014.

[SANS Inst. 2003]SANS Inst. *Penetration Testing: Assessing Your Overall Security Before Attackers Do*. Bethesda, MD, USA, 2003. InfoSec Reading Room. In: https://www.sans.org/reading-room/whitepapers/analyst/penetration-testing-assessing-security-attackers-34635. Accessed Aug, 2014.

[Scarfone et al. 2008]SCARFONE, K. A. et al. *NIST SP 800-115. Technical Guide to Information Security Testing and Assessment*. NIST. Gaithersburg, MD, United States, 2008.

[Sgouras, Birda e Labridis 2014]SGOURAS, K. I.; BIRDA, A. D.; LABRIDIS, D. P. Cyber attack impact on critical smart grid infrastructures. In: IEEE. *Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES*. New York, NY, USA, 2014. p. 1–5.

[Sharon 2015]SHARON, S. *2015 DDoS attacks on the rise, attackers shift tactics*. USA, 2015. In http://searchsecurity.techtarget.com/news/4500246858/2015-DDoS-attacks-on-the-rise-attackers-shift-tactics. Accessed in May 2016.

[Shewhart e Deming 1939]SHEWHART, W. A.; DEMING, W. E. *Statistical method from the viewpoint of quality control*. North Chelmsford, MA, USA: Courier Corporation, 1939.

[Singh e Panda 2015]SINGH, B.; PANDA, S. Defending against DDoS flooding attacks - A data streaming approach. *International Journal of Computer & IT*, 2015.

[Sonar e Upadhyay 2016]SONAR, K.; UPADHYAY, H. Proceedings of International Conference on ICT for Sustainable Development: ICT4SD 2015 Volume 2. In: ____. Singapore: Springer Singapore, 2016. cap. An Approach to Secure Internet of Things Against DDoS, p. 367–376. ISBN 978-981-10-0135-2. Disponível em: <http://dx.doi.org/10.1007/978-981-10-0135-2_36>.

[Spafford 1989]SPAFFORD, E. H. The internet worm program: An analysis. *SIGCOMM Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 19, n. 1, p. 17–57, Jan 1989. ISSN 0146-4833. Disponível em: <http://doi.acm.org/10.1145/66093.66095>.

[Stoll 1989]STOLL, C. *The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage*. New York, NY, USA: Doubleday, 1989. ISBN 0-385-24946-2.

[Tang 2014]TANG, A. A guide to penetration testing. *Network Security*, v. 2014, n. 8, p. 8 – 11, 2014. ISSN 1353-4858. Disponível em: <"http://www.sciencedirect.com/science/article/pii/S1353485814700790">.

[Ten, Liu e Manimaran 2008]TEN, C. W.; LIU, C. C.; MANIMARAN, G. Vulnerability assessment of cybersecurity for scada systems. *IEEE Transactions on Power Systems*, v. 23, n. 4, p. 1836–1846, Nov 2008. ISSN 0885-8950.

[Tiller 2012]TILLER, J. S. *CISO's guide to penetration testing: a framework to plan, manage, and maximize benefits*. 2000 N.W. Corporate Blvd., Boca Raton, FL 33431-9868, USA: CRC Press, 2012. xiv + 374 p. ISBN 1-4398-8027-1 (hardcover).

[Toms 2016]TOMS, L. *Closed for Business - The Impact of Denial of Service Attacks in the IoT*. Portsmouth, NH, USA, 2016. In `https://www.globalsign.com/en/blog/denial-of-service-in-the-iot/`. Accessed in May 2016.

[US-CERT 2014]US-CERT. *Alert (TA14-017A UDP-Based Amplification Attacks)*. Washington, DC, USA, 2014. In `https://www.us-cert.gov/ncas/alerts/TA14-017A`. Accessed in May 2016.

[Wang e Guo 2009]WANG, J. A.; GUO, M. Ovm: An ontology for vulnerability management. In: *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*. New York, NY, USA: ACM, 2009. (CSIIRW '09), p. 34:1–34:4. ISBN 978-1-60558-518-5. Disponível em: <http://doi.acm.org/10.1145/1558607.1558646>.

[Watson 2014]WATSON, G. *Social engineering penetration testing: executing social engineering pen tests, assessments and defense*. Rockland, MA, USA: Syngress Publishing, Inc., 2014. ISBN 0-12-420124-5.

[Weidman 2014]WEIDMAN, G. *Penetration testing: a hands-on introduction to hacking*. San Francisco, CA, USA: No Starch Press, 2014. ISBN 1-59327-564-1 (paperback).

[Wilhelm 2010]WILHELM, T. (Ed.). *Professional penetration testing: creating and operating a formal hacking lab*. Rockland, MA, USA: Syngress Publishing, Inc., 2010. xix + 504 p. ISBN 1-59749-425-9 (paperback), 1-59749-466-6 (DVD).

[Wilhelm 2013]WILHELM, T. (Ed.). *Professional penetration testing*. Second. Rockland, MA, USA: Syngress Publishing, Inc., 2013. ISBN 1-59749-993-5.

[Wilhelm e Andress 2011]WILHELM, T.; ANDRESS, J. *Ninja hacking: unconventional penetration testing tactics and techniques*. Rockland, MA, USA: Syngress Publishing, Inc., 2011. xii + 310 p. ISBN 1-59749-588-3.

[Wireshark Foundation 2015]WIRESHARK FOUNDATION. USA, 2015. In `https://www.wireshark.org`. Accessed in Oct, 2015.

[Xylogiannopoulos, Karampelas e Alhajj 2016]XYLOGIANNOPOULOS, K.; KARAMPELAS, P.; ALHAJJ, R. Real time early warning DDoS attack detection. In: ACADEMIC CONFERENCES AND PUBLISHING LIMITED. *11th International Conference on Cyber Warfare and Security: ICCWS2016*. Sonning Common, England, UK, 2016. p. 344.

[Yeo 2013]YEO, J. Using penetration testing to enhance your company's security. *Computer Fraud & Security*, v. 2013, n. 4, p. 17 – 20, 2013. ISSN 1361-3723. Disponível em: <"http://www.sciencedirect.com/science/article/pii/S1361372313700393">.

[Yu et al. 2015]YU, T. et al. Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the internet-of-things. In: *Proceedings of the 14th ACM Workshop on Hot Topics in Networks*. New York, NY, USA: ACM, 2015. (HotNets-XIV), p. 5:1–5:7. ISBN 978-1-4503-4047-2. Disponível em: <http://doi.acm.org/10.1145/2834050.2834095>.

[Zhang e Green 2015]ZHANG, C.; GREEN, R. Communication security in internet of things: Preventive measure and avoid DDoS attack over IoT network. In: *Proceedings of the 18th Symposium on Communications & Networking*. San Diego, CA, USA: Society for Computer Simulation International, 2015. (CNS '15), p. 8–15. ISBN 978-1-5108-0100-4. Disponível em: <http://dl.acm.org/citation.cfm?id=2872550.2872552>.

[Zhuang et al. 2015]ZHUANG, R. et al. A theory of cyber attacks: a step towards analyzing mtd systems. In: ACM. *Proceedings of the Second ACM Workshop on Moving Target Defense*. New York, NY, USA, 2015. p. 11–20.

Apêndice: Resumo das Fases, Atividades e Artefatos

- Fase I – Especificação do Projeto
  - Atividades:
    - Apresentação da Demanda
    - Definir Escopo da Demanda
    - Validar Escopo da Demanda
    - Definir os Requisitos
    - Validar os Requisitos da Demanda
    - Detalhamento do Projeto
    - Validar o Detalhamento do Projeto
  - Artefatos:
    - Cheklist de Objetivos
    - Checklist de Requisitos
    - Especificação Detalhada do Projeto
- Fase II - Levantamento Inicial
  - Atividades:
    - Definição do Trabalho e Alocação de Tarefas
      - Levantamento de Informações
      - Realizar Varreduras de Redes
      - Realizar Varreduras de Serviços de Redes
      - Realizar Varreduras de Aplicações
      - Gerar Relatório de Varreduras
      - Gerar Lista de Alvos e Vulnerabilidades
    - Artefatos:

·Registro de Varreduras de Ferramentas

·Relatório de Varreduras

·Lista de Alvos e Vulnerabilidades

–Fase III – Planejamento do Pentest

∗Atividades:

·Definição da Estratégia do Pentest

·Definir Alvos Prioritários

·Definir Alvos Emergentes

·Definir Táticas e Ferramentas

·Consolidação de Alvos e Táticas

∗Artefatos:

·Lista de Alvos e Vulnerabilidades

·Registro das Varreduras das Ferramentas

·Plano de Ataques

–Fase IV - Execução dos Testes de Penetração

∗Atividades:

·Definir Recursos e Tarefas

·Iniciar o Ataque

·Observar o Comportamento do Alvo

·Analisar o Comportamento do Alvo

·Decidir em Função do Comportamento do Alvo

·Analisar em Função do Comportamento do Alvo

∗Artefatos:

·Plano de Ataques

·Material Auxiliar de Registro de Ataques

·Relatório de Execução de Ataques

–Fase V - Consolidação e Análise

∗Atividades:

·Analisa Objetivos versus Resultados

·Correlacionar Dados dos Resultados do Pentest

·Gerar Relatório Consolidado do Teste de Penetração

∗Artefatos:

·Especificação Detalhada do Projeto

·Lista de Alvos e Vulnerabilidades

·Plano de Ataques

·Material Auxiliar de Registro de Ataques

·Relatório de Execução de Ataques

·Relatório Consolidado do Teste de Penetração

–Fase VI - Apresentação dos Resultados

∗Atividades:

·Elaborar Relatório Final do Teste de Penetração

·Apresentar Relatório Final do Teste de Penetração

∗Artefatos:

·Relatório Consolidado do Teste de Penetração

·Material Auxiliar de Registro de Ataques

·Relatório Final do Teste de Penetração

·Relatório Interno de Avaliação do Projeto