MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO

MARTA RIGAUD FARIA

# DEFESA: UMA METODOLOGIA PARA ANÁLISE DE INCIDENTES DE SEGURANÇA DA INFORMAÇÃO

RIO DE JANEIRO
2020

**MARTA RIGAUD FARIA**

# DEFESA: UMA METODOLOGIA PARA ANÁLISE DE INCIDENTES DE SEGURANÇA DA INFORMAÇÃO

Dissertação apresentada ao Programa de Pós-graduação em Sistemas e Computação do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Mestre em Ciências em Sistemas e Computação.

Orientador: Maria Claudia Reis Cavalcanti - D.Sc.
Coorientador: Kelli de Faria Cordeiro - D.Sc.

Rio de Janeiro

2020

Faria, Marta Rigaud
DEFESA: uma Metodologia para Análise de Incidentes de Segurança da Informação / Marta Rigaud Faria. – Rio de Janeiro, 2020.
155 f.

Orientador: Maria Claudia Reis Cavalcanti.
Coorientador: Kelli de Faria Cordeiro.

Dissertação (mestrado) – Instituto Militar de Engenharia, Sistemas e Computação, 2020.

1. Ontologia. 2. UFO. 3. Incidente de Segurança da Informação. 4. Data Warehouse. I. Cavalcanti, Maria Claudia Reis, orient. II. Cordeiro, Kelli de Faria, coorient. III.Título

MARTA RIGAUD FARIA

# DEFESA: uma Metodologia para Análise de Incidentes de Segurança da Informação

Dissertação apresentada ao Programa de Pós-graduação em Sistemas e Computação do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Mestre em Ciências em Sistemas e Computação.

Orientador: Maria Claudia Reis Cavalcanti
Coorientador: Kelli de Faria Cordeiro

Aprovado em Rio de Janeiro, 19 de março de 2020, pela seguinte banca examinadora:

Prof. Julio Cesar Duarte - D.Sc do IME - Presidente

Profª. Maria Claudia Reis Cavalcanti - D.Sc. do IME

Profª. Kelli de Faria Cordeiro - D.Sc. do IME

Profª. Maria Luiza Machado Campos - Ph.D. da UFRJ

Prof. João Paulo Andrade Almeida - Ph.D. da UFES

Rio de Janeiro
2020

# AGRADECIMENTOS

À Marinha do Brasil, particularmente à Diretoria de Telecomunicação da Marinha e à Secretaria Geral da Marinha, por ter me proporcionado a oportunidade de realizar o curso. E, especificamente, ao CF(FN) Adriano Cunha por haver me incentivado a me inscrever no processo seletivo para o mestrado, ao CMG(IM) Weney responsável pela minha participação no processo seletivo para o mestrado e à CC(T) Kátia por haver me dado todo suporte necessário durante o curso.

Ao Instituto Militar do Exército que me recebeu e acolheu muito bem.

Ao Ten Cel Julio Cesar Duarte, coordenador do curso, pelo tratamento atencioso e por todo apoio principalmente nas questões burocráticas.

Aos membros da banca examinadora, pelas observações e sugestões que contribuíram e enriqueceram este trabalho.

Às minhas orientadoras Maria Claudia Cavalcanti e CF(T) Kelli Cordeiro que me acolheram. Durante os dois anos do mestrado, elas acenderam várias vezes uma lanterna quando estava escuro, me norteando durante o caminho.

À professora Maria Luiza Machado Campus da UFRJ pela oportunidade de participar da disciplina Fundamentos da Modelagem. Essa participação engrandeceu muito meu conhecimento e me ofereceu uma gama de possibilidades para o desenvolvimento do meu trabalho. Foram muitos e-mails, reuniões e consultorias valiosas.

À minha amiga e colega de mestrado CC(T) Glaucia Botelho que embarcou comigo nos estudos e no desenvolvimento do nosso artigo.

Ao Alessander Boni Benevides por todo suporte técnico com relação a aplicação da UFO no trabalho.

À minha amiga e colega de curso CC(T) Lucimar pelas conversas, pelo incentivo e por haver sido o ouvido amigo durante os momentos difíceis dessa trajetória.

À toda a minha família pelo apoio e paciência, especialmente à minha filha Julia Coutinho que esteve ao meu lado incansavelmente nessa caminhada me fortalecendo e encorajando a encarar todos os desafios do percurso.

# RESUMO

O aumento substancial da gravidade dos danos e prejuízos causados por Incidentes de Segurança da Informação levam à busca de novas estratégias de defesa cibernética. E, ao observar os relatórios estatísticos, percebe-se que a mesma técnica de ataque costuma ser replicada inúmeras vezes. Diante desses fatos, as organizações têm buscado reunir as informações de Incidentes de Segurança de Informação em um único ambiente para analisá-las e compará-las com o objetivo de oferecer um suporte à tomada de decisão que leve a uma redução significativa do número de ocorrências de incidentes. Porém, os conflitos conceituais inerentes do domínio e as diferentes formas de categorização tornam a tarefa de reunir informações bastante complexa. Apesar de haver várias ontologias de Incidente de Segurança da Informação, elas enfatizam a representação de parte do domínio, ou seja, não visam uma abordagem aberta e genérica para a interoperabilidade. Todas essas diferentes formas de representação e, até mesmo a falta de uma representação formal evidenciam a necessidade de haver um modelo amplamente aceito para ser utilizado como referência para comunicar, trocar informações e correlacionar as ontologias que representem parte do domínio. Em outros domínios os conflitos conceituais foram minimizados com o uso da chamada análise ontológica, que usa ontologias de fundamentação para nortear a concepção de um modelo de referência. Neste trabalho, escolheu-se utilizar a UFO, como ontologia de fundamentação, devido ao seu amplo uso em diversos domínios. Além disso, em domínios que requerem múltiplos níveis de classificação, a MLT tem sido aplicada em conjunto com a UFO. Outras abordagens, se beneficiam da expressividade semântica da ontologia para auxiliar no desenvolvimento de sistema de apoio à decisão. Visando obter todos esses benefícios no domínio de Incidente de Segurança da Informação, este trabalho propõe uma metodologia para **D**efinir, **E**specificar e **F**ormalizar os conceitos do domínio com **Ê**nfase em oferecer **S**uporte à tomada de decisão e **A**umentar a expressividade semântica (**DEFESA**). A metodologia DEFESA tem o propósito de ser abrangente com relação a descrição dos conceitos e específica em oferecer suporte à tomada de decisão formando uma arquitetura em camadas. Na camada fundacional fica a UFO-MLT para que os seus sistemas de categorias embasem a ontologia do domínio. Esta ontologia fica no centro da arquitetura, representando os conceitos, suas relações e seus tipos e alicerça o modelo dimensional com expressividade semântica para atender à demanda de construção do sistema de apoio à decisão. A aplicação da metodologia DEFESA no domínio de Incidente de Segurança da Informação resultou na modelagem de sCuDO, a ontologia de domínio de Incidente de Segurança da Informação, na elaboração de sCuD$^2$O, o modelo dimensional de Incidentes de Segurança da Informação e no desenvolvimento de um ambiente de análise com os dados de ocorrências de incidentes de um Grupo de Resposta de Incidentes de Segurança em Computadores.

**Palavras-chave**: Ontologia. UFO. Incidente de Segurança da Informação. Data Warehouse.

# ABSTRACT

The substantial increase in the gravity of damages and losses caused by Information Security Incidents guide us on the search for new cyber defence strategies. And, observing statistical reports, it is understood that the same attack technical is usually applied numerous times. In face of these facts, organizations have been trying to congregate the Information Security Incident informations in a single environment so it is possible to analyze and compare them aiming to offer a support to decision making that will lead to a significant reduction in the number of incidents occurrences. However, conceptual conflicts inherent of this domain and the different ways of categorization make the task of information congregating quite complex. Despite existing several Information Security Incident ontologies, they emphasize the representation of part of the domain, meaning they do not aim at an open and generic approach to facilitate the interoperability. All these different forms of representation and even the lack of formal representation show the need for a widely accepted model to be used as a reference to communicate, exchange information and correlate ontologies that represent part of the domain. In other domains, conceptual conflicts were minimized with the use of the so-called ontological analysis, which uses ontological foundations to guide the design of a reference model. In this work, it was chosen to use UFO, as a foundational ontology, due to its wide use in several domains. And, in domains that require multiple levels of classification, MLT has been applied along with UFO. Other approaches benefit themselves from the ontology semantic expressiveness to assist in the development of a decision support system. Aiming to obtain all of these benefits in the Information Security Incident domain, this work proposes a methodology to define, specify and formalize the concepts of the domain with emphasis in offering support to the decision making and enhance the semantic expressiveness (**DEFESA**). The DEFESA methodology has the purpose of being comprehensive regarding the description of the concepts and being specific in offering support to decision making, forming a layered architecture. In the foundational layer there is UFO-MLT so that its metacategories system underlies the domain ontology. This ontology stays in the center of the architecture, representing the concepts, its relations and its types and underlies the dimensional model with semantic expressiveness to meet the demands of a decision making system. The application of DEFESA methodology in Information Security Incident domain resulted in the modeling of sCuDO, an ontology of Information Security Incident domain, in the elaboration of sCuD$^2$O, the dimensional model of Information Security Incident and in the development of an environment of analysis with the data of incidents occurrences from a Group of Response to Security Incidents in Computers.

**Palavras-chave**: Ontology. UFO. Information Security Incident. Data Warehouse.

# LISTA DE ILUSTRAÇÕES

# LISTA DE QUADROS

# LISTA DE TABELAS

# LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| 1stOT | First-Order Type |
| 2ndOT | Second-Order Type |
| BI | Business Intelligence |
| CAPEC | Common Attack Pattern Enumeration and Classification |
| CERT.Bahia | Grupo de Resposta a Incidentes de Segurança da Bahia |
| CERT.br | Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil |
| CSIRT | Computer Security Incident Response Team |
| CTIR Gov | Centro de Tratamento e Resposta a Incidentes Cibernéticos do Governo |
| CVE | Common Vulnerabilities and Exposures |
| CWE | Common Weakness Enumeration |
| DDoS | Distributed Denial of Service |
| DoS | Denial of Service |
| DW | Data warehouse |
| HTTP | HyperText Transfer Protocol |
| END | Estratégia Nacional de Defesa |
| EUA | stados Unidos da América |
| IDS | Intrusion detection system |
| iof | Instance of |
| MD | Modelagem Dimensional |
| MLT | Multi-Level Theory |
| OLAP | Online Analytical Processing |
| RNP | Rede Nacional de Ensino e Pesquisa |
| SABiO | Systematic Approach to Build Ontologies |

| | |
|---|---|
| SEON | Software Engineering Ontology Network |
| sCuD$^2$O | Modelo dimensional de Incidentes de Segurança da Informação |
| sCuDO | Modelo multi-nível baseado em ontologias para representação de Incidente de Segurança da Informação |
| UFO | Unified Foundational Ontology |
| UML | Unified Modeling Language |

# LISTA DE SÍMBOLOS

| | |
|---|---|
| $\forall$ | Quantificação universal |
| $\rightarrow$ | Implicação material |
| $\neg$ | Negação |
| $\leftrightarrow$ | Equivalência material |
| $\exists$ | Quantificação existencial |
| $\exists!$ | Quantificação existencial única |
| $\wedge$ | Conjunção lógica |
| $\vee$ | Disjunção lógica |

# SUMÁRIO

# 1 INTRODUÇÃO

A popularização da Internet proporcionou a democratização do acesso à informação, bem como uma nova era de interações e comunicação. Paralelamente a esse fenômeno, há um aumento substancial da gravidade e prejuízos causados por Incidentes de Segurança da Informação. Tais incidentes podem causar danos graves e, portanto, as estratégias de defesa cibernética devem ser aprimoradas constantemente para evitá-los ou mitigá-los o mais rapidamente possível (7).

Muitos sistemas de detecção de intrusão (*Intrusion Detection System* - IDS) e ferramentas de segurança vem sendo utilizados na tentativa de proteger os ambientes computacionais de incidentes (18). Porém, a variedade de recursos utilizados para provocá-los dificulta a implementação de estratégias eficazes que inviabilize completamente a ação maliciosa de atacantes. Relatórios estatísticos sobre Incidentes de Segurança de Informação demonstram que o número de ocorrências ainda é muito elevado (19).

Apesar do grande número de ocorrência de incidentes, percebe-se que a mesma técnica de ataque é replicada inúmeras vezes. As organizações para se proteger têm buscado reunir as informações de Incidentes de Segurança de Informação em um único ambiente para analisá-las e compará-las com o objetivo de oferecer um suporte à tomada de decisão que leve a uma redução significativa do número de ocorrências de incidentes (20).

## 1.1 Motivação

Uma das estratégias que está sendo adotada por várias organizações para reunir as informações de Incidentes de Segurança de Informação é haver um Grupo de Resposta a Incidentes de Segurança em Computadores, geralmente conhecido como CSIRT, abreviação do termo em inglês *Computer Security Incident Response Team*. O CSIRT recebe, analisa e responde a notificações de Incidentes de Segurança da Informação. Geralmente, fornece serviços para uma comunidade bem definida, servindo como um ponto central para relatar problemas locais. Assim, todos os incidentes relatados devem ser coletados em um único local, onde as informações de toda a comunidade podem ser analisadas e correlacionadas para que possam ser descobertas tendências e padrões de atividades de atacantes e recomendar medidas de prevenção adequadas (20).

Apesar de haver vários CSIRTs em operação no Brasil, ainda não há uma conceituação clara de incidente que seja usada em consenso (21) e, muitas vezes, ocorrem inconsistências. Este fato foi evidenciado no relatório estatístico de 2018 do Centro de Tratamento e Resposta a Incidentes Cibernéticos do Governo (CTIR Gov), responsável

por apoiar os órgãos e entidades da Administração Pública Federal. Conforme extrato do relatório ilustrado na Figura 1, em 2018, 20.566 notificações foram reportadas e, após um cuidadoso processo de triagem, apenas 9.981 delas foram realmente considerados incidentes [1]. Mesmo o CTIR Gov não revelando as características analisadas e os critérios para avaliação dos incidentes, percebe-se que há uma divergência de conceituação na comunidade.

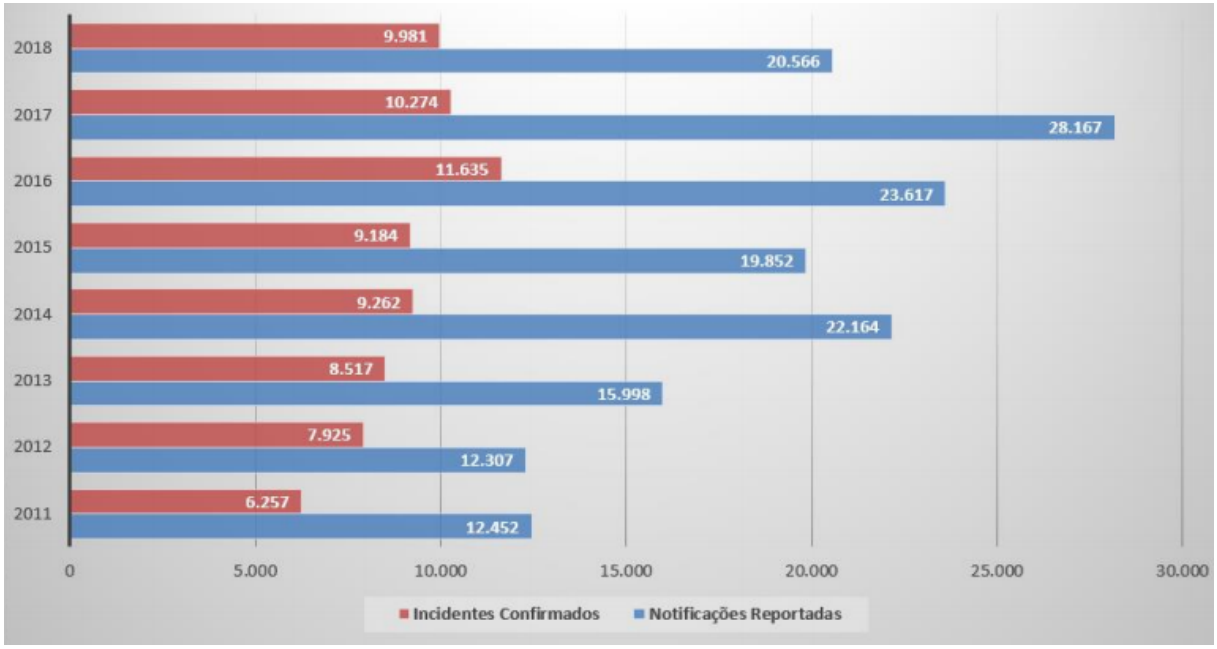



Figura 1 – Estatística de notificações reportadas e incidentes confirmados pelo CTIR Gov

Outro fator relevante está relacionado aos relatórios estatísticos produzidos pelos CSIRTs, cada um deles utiliza uma forma diferente de categorização os incidentes, dificultando a correlação de informações entre CSIRTs. Por exemplo, CTIR Gov usa oito categorias para classificar incidentes, o Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil (CERT.br) usa seis categorias [2] e o Grupo de Resposta a Incidentes de Segurança da Bahia (CERT.Bahia) utiliza dez categorias [3], conforme ilustrado, respectivamente, nas Figuras 2, 3 e 4.

Os incidentes são classificados de acordo com o tipo de ataque que os ocasionou. Porém, os ataques foram evoluindo ao longo do tempo e existem muitos tipos de ataque que são especializados em outros tipos. Por exemplo, um incidente que tenha levado a indisponibilidade de um *site* pode ser classificado de diversas maneiras. De forma mais genérica, pode-se dizer que o *site* sofreu ataques que abusaram de alguma de suas funcionalidades tornando-o indisponível (*Abuse existing funcionality*). Posteriormente, o pessoal responsável pelo monitoramento do *site* ao perceber que foram enviadas simultaneamente

[1] https://emnumeros.ctir.gov.br/
[2] https://www.cert.br/stats/incidentes/2018-jan-dec /types-tape.html
[3] https://certbahia.pop-ba.rnp.br/pages/stats/

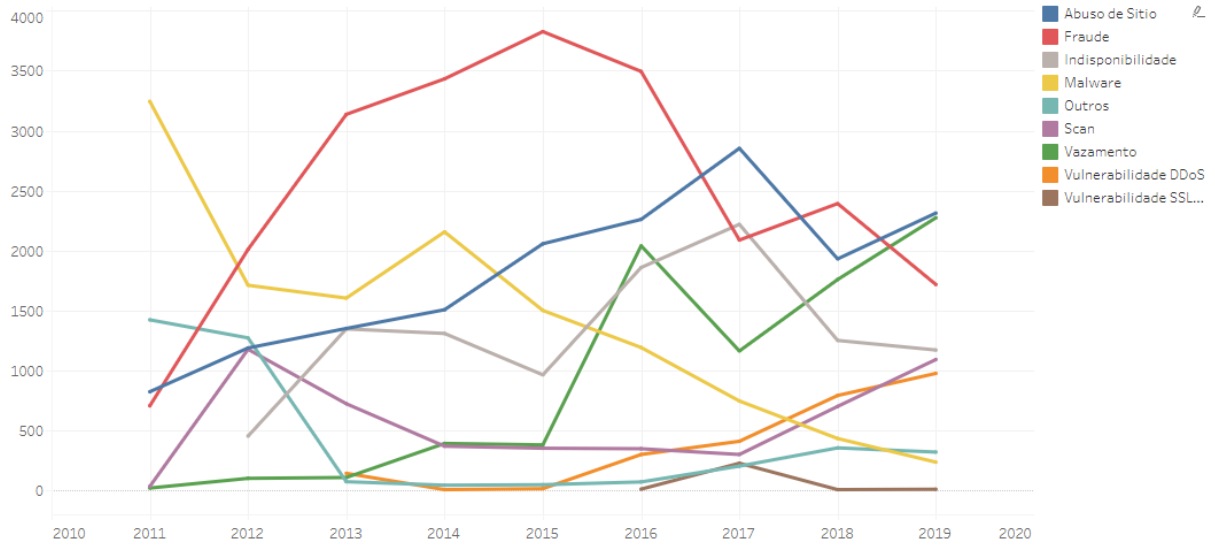



Figura 2 – Categorias de incidentes do CTIR Gov

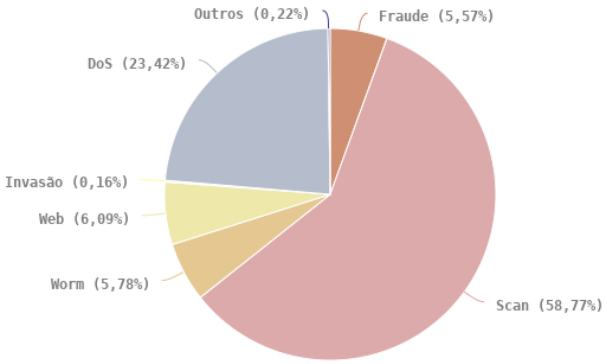



Figura 3 – Categorias de incidentes do CERT.br

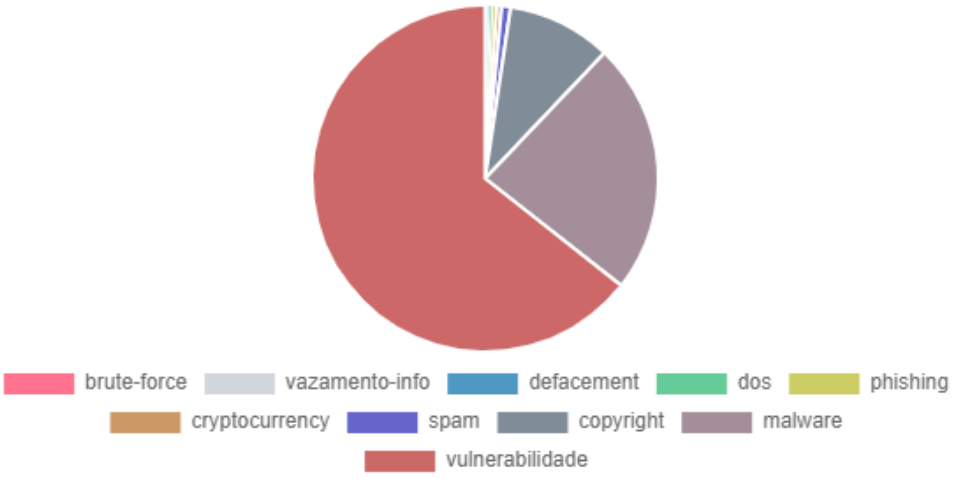



Figura 4 – Categorias de incidentes do CERT.Bahia

mais requisições para o *site* do que ele suporta, o incidente poderia ser classificado considerando a inundação (*Flooding*) de requisições. E, mais especificamente, o *site* recebeu

requisições HTTP que o inundaram (*HTTP flood*). Esse exemplo ilustra três possibilidades de classificação de uma mesma ocorrência de incidente: o abuso de funcionalidade existente (*Abuse existing funcionality*), inundação (*Flooding*) e inundação utilizando protocolo HTTP (*HTTP flood*). Porém, essa estrutura hierárquica de tipos e subtipos normalmente não é claramente definida levando a inconsistências e ambiguidades, tais como, possuir diversos registros da mesma ocorrência usando diferentes critérios de classificação e cada registro erroneamente ser considerado uma ocorrência distinta.

Todas essas diversidades tornam a tarefa de reunir informações bastante complexa. Essa complexidade foi evidenciada durante as Olimpíadas de 2016 quando houve uma grande preocupação com a defesa cibernética. Neste período, o Brasil contou com cerca de 200 especialistas de várias empresas e órgãos governamentais para atuar na proteção cibernética durante os jogos. Encerrada as Olimpíadas, foram divulgados dados estatísticos sobre a defesa cibernética que revelam como a falta de uma conceituação comum dificulta a comunicação. A empresa Cisco, responsável pelos equipamentos de rede e serviços corporativos das Olimpíadas, divulgou que ocorreram 4,2 milhões de eventos de segurança e 731.607 tentativas de ataques de negação de serviço foram bloqueadas (22). A empresa de segurança Symantec, responsável pelos sistemas e softwares empregados no esquema de segurança digital, informou que ocorreram 2.686 incidentes e, contando os ataques presentes em cada incidente, os programas da Symantec barraram 50 mil ataques (23). Já a Força Aérea Brasileira informou haver identificado uma média de 40 incidentes por dia (24).

Esses números discrepantes retratam que cada órgão utilizou sua própria conceituação para registrar um evento como um incidente, dificultando a análise e comparação desses dados. Apesar do sucesso da operação, ficou evidente a necessidade de aperfeiçoamento das táticas de defesa cibernética de modo a estruturar medidas sistemáticas (25).

Tal necessidade vem ganhando destaque do Governo Brasileiro, desde 2008, quando foi estabelecido o setor cibernético como um dos três setores de importância estratégica para a Defesa Nacional (26). Dentre as prioridades desse setor está o fomento a pesquisa científica voltada para o setor cibernético.

Nesse contexto, o Instituto Militar de Engenharia (IME) atua na esfera de ciência e tecnologia e na formação de recursos humanos (27). E, a Marinha do Brasil no investimento na formação de seus militares na área de defesa. Em virtude disso, este trabalho foi motivado pela necessidade de usar a formação do IME para capacitar militares da Marinha do Brasil para que sejam capazes de estudar formas de colaborar com a defesa cibernética.

## 1.2 Caracterização do problema

A necessidade de correlacionar informações sobre Incidentes de Segurança da Informação estimulou a construção de algumas ontologias, como por exemplo, (10), (11) e outras citadas na Seção 3.1. Contudo, essas ontologias enfatizam a representação de alguns elementos ou classes para um ambiente específico, ou seja, nenhuma delas visa uma abordagem aberta e genérica para a interoperabilidade.

Todas essas diferentes formas de representação e, até mesmo a falta de uma representação formal evidenciam a necessidade de haver um modelo amplamente aceito para ser utilizado como referência para comunicar, trocar informações e correlacionar as ontologias que representem parte do domínio.

Neste domínio, para reduzir a heterogeneidade semântica, alguns elementos presentes no evento incidente devem ser representados. Para isso, devem ser especificados o contexto no qual o incidente ocorre, os participantes envolvidos e seus respectivos papéis, bem como as consequências do incidente. Além disso, os critérios de classificação dos tipos de ataque mais gerais que são especializados em tipos mais específicos também devem ser elucidados.

## 1.3 Objetivo

O objetivo geral deste trabalho é prover um apoio metodológico conceitual para a troca de informações sobre Incidentes de Segurança da Informação visando oferecer suporte à tomada de decisões na área de defesa cibernética. Para tal, os conceitos do domínio devem possuir maior expressividade semântica para serem melhor compreendidos e comunicados. Isto poderá facilitar o entendimento de informações sobre Incidentes de Segurança da Informação que foram produzidas isoladamente e que possuem diferentes conceituações. Esse objetivo geral pode ser decomposto nos seguintes objetivos específicos:

(i) Representar os conceitos do domínio de forma mais explícita para facilitar na distinção dos diferentes conceitos, prover um melhor entendimento deles, bem como uma melhor comunicação entre os envolvidos, ajudando no estabelecimento de um consenso.

(ii) Representar as relações existentes entre os conceitos, visando apoiar discussões entre as partes envolvidas e um aumento da consciência situacional.

(iii) Evidenciar os critérios de classificação dos tipos de ataques e as características usadas para os particionar em subtipos. Muitas vezes, as ocorrências de incidentes são agrupadas de acordo com os tipos de ataque que as ocasionaram. Porém, cada comunidade utiliza seu próprio critério de classificação dificultando a correlação de

informações. Para conseguir correlacionar tais informações, primeiramente deve ser comunicado o critério de classificação desses tipos. Além disso, os tipos podem ser especializados em diversos subtipos. A hierarquia de tipos nos quais os tipos mais específicos geralmente formam uma partição de um tipo mais geral distinguindo ocorrências de incidentes de acordo com um critério de classificação específico também deve ser conhecida.

(iv) Projetar um ambiente favorável a integração e análise das notificações de incidentes de forma a oferecer suporte à tomada de decisão.

## 1.4 Contribuições esperadas

Baseando-se nos objetivos definidos e nos problemas de pesquisa apresentados as contribuições esperadas para este trabalho são:

(i) Especificação de uma metodologia para definir, especificar e formalizar os conceitos do domínio com ênfase em oferecer Suporte à tomada de decisão e aumentar a expressividade semântica. Esta metodologia embasará as demais contribuições;

(ii) Especificação formal e explícita dos conceitos do domínio de Incidente de Segurança da Informação para apoiar o compartilhamento de informações;

(iii) Especificação de um projeto de um ambiente de apoio à decisão de Incidentes de Segurança da Informação para que as ocorrências de incidentes possam ser integradas e analisadas.

## 1.5 Metodologia de pesquisa

Este trabalho, a fim de colaborar com a defesa cibernética, iniciou com o levantamento dos problemas do domínio para que fossem definidos os objetivos da pesquisa. Uma vez delineado o escopo, foram buscados trabalhos relacionados para que experiências de sucesso pudessem servir de exemplo para atender a demanda deste domínio. Com base nos exemplos apresentados nos trabalhos relacionados, foi realizado um levantamento do referencial teórico para identificar quais teorias seriam mais adequadas para atingir o objetivo proposto.

Como resultado foi verificado que seria necessário o uso de uma combinação de teorias. Sendo assim, a estratégia adotada foi desenvolver uma metodologia para combinar tais teorias para prover uma análise de Incidentes de Segurança da Informação. Esta metodologia foi aplicada no domínio de Incidente de Segurança da Informação e testada através de cenários de uso e no desenvolvimento de um sistema de apoio à decisão.

## 1.6 Organização do trabalho

Este trabalho está dividido em 9 capítulos. Além da presente introdução, os fundamentos teóricos que embasam a abordagem proposta estão reunidos no Capítulo 2. O Capítulo 3 apresenta algumas ontologias para representação de Incidente de Segurança da Informação, uma abordagem usada para modelar tipo e subtipos, os benefícios do uso do sistema de apoio à decisão de Incidentes de Segurança da Informação e as vantagens da modelagem dimensional baseada em ontologia. No capítulo 4, desenvolve-se a metodologia DEFESA para análise de Incidentes de Segurança da Informação composta por dois macroprocessos. No Capítulo 5, o primeiro macroprocesso foi usado para construir sCuDO, a ontologia de domínio de Incidente de Segurança da Informação, e no Capítulo 6 sCuDo foi usada para representar ocorrências de incidentes. O segundo macroprocesso da metodologia DEFESA deu origem a dois capítulos: o Capítulo 7 dedicado a elaboração de sCuD$^2$O, o modelo dimensional de Incidentes de Segurança da Informação; e o Capítulo 8 que desenvolve um ambiente analítico de dados. Por fim, o Capítulo 9 apresenta a conclusão deste trabalho, as contribuições, as dificuldades e os trabalhos futuros.

# 2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo descreve os fundamentos teóricos que envolvem a abordagem proposta. No sentido de buscar uma representação mais explícita do domínio de Incidente de Segurança da Informação, teorias e técnicas para a análise conceitual foram buscadas na literatura (Seção 2.1) e metodologias usadas para criar esse tipo de representação (2.2). A Seção 2.3 mostra como projetar um ambiente de apoio à decisão para que as ocorrências Incidentes de Segurança da Informação possam ser integradas e analisadas e métodos usados para criá-lo com base em um modelo conceitual bem fundamentado (2.4).

## 2.1 Teorias de modelagem conceitual

### 2.1.1 Ontologia

Uma ontologia é uma especificação formal e explícita de uma conceituação compartilhada, ou seja, um modelo abstrato que representa um fenômeno no mundo real (28). A ontologia vem sendo utilizada para remover ambiguidades conceituais e facilitar o compartilhamento de informações e interoperabilidade de sistemas (29). Conceitos, relações e axiomas, previamente definidos e fundamentados, são utilizados para descrever um modelo de domínio uniforme e não ambíguo de entidades e suas relações. Fornecendo assim, uma conceituação sobre o domínio modelado (30).

Ontologias possuem diversas classificações. Uma das mais conhecidas foi proposta por GUARINO(1), que define quatro tipos de ontologias com base no grau de generalidade: Ontologias de Fundamentação, Ontologias de Domínio, Ontologias de Tarefa e Ontologias de Aplicação. Ontologias de Fundamentação descrevem conceitos muito gerais como espaço, tempo, objeto, evento etc. Elas servem como base para as ontologias de domínio e de tarefa. Ontologias de Domínio, por sua vez, descrevem o vocabulário relacionado a um domínio específico. Ontologias de Tarefa descrevem tarefas ou atividades genéricas. E, por fim, as Ontologias de Aplicação descrevem conceitos que dependem de domínios e tarefas específicas, que são comumente especializados das duas ontologias relacionadas. Tal classificação é a adotada nesse trabalho e demonstrada na Figura 5.

No entanto, alguns domínios são grandes e complexos que se fossem representados em uma única ontologia do domínio ficaria extensa e difícil de manipular, usar e manter (31). Por outro lado, representar cada subdomínio separadamente ficaria muito caro, fragmentado e difícil de manusear. Sabendo disso, Ruy et al.(2) propuseram um arcabouço ontológico integrado para representar o domínio de engenharia de *sofware*, de forma que as ontologias sejam construídas de forma incremental e integrada, compartilhando conceitos

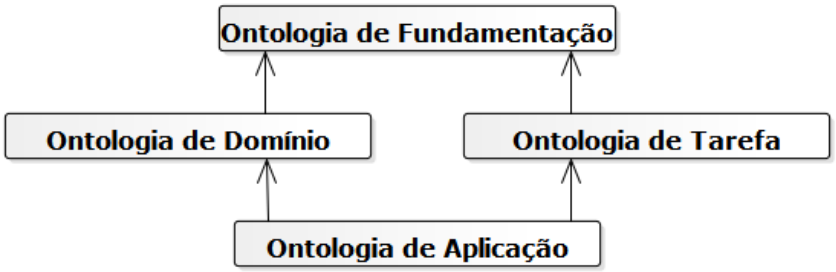



Figura 5 – Tipos de ontologias. Adaptado de:(1)

e relações com outras ontologias e formando uma rede de ontologias de engenharia de *software* (*The Software Engineering Ontology Network - SEON*).

SEON é organizada em camadas, conforme ilustrado na Figura 6. Na base da rede fica a camada fundacional (Fundacional Layer) composta por uma ontologia de fundamentação para fornecer o conhecimento geral para classificar conceitos e relações de todas as ontologias da rede. Na camada central (Core Layer), as ontologias de núcleo (Core ontologies) devem ser usadas para representar o conhecimento geral do domínio, sendo a base para as ontologias de subdomínio. Na camada de conhecimentos específicos do domínio (Domain-specific Layer) ficam as ontologias do domínio (Domains Ontologies) que descrevem conhecimentos mais específicos. E, ontologias mais específicas são desenvolvidas com base em outras ontologias de domínio mais gerais.

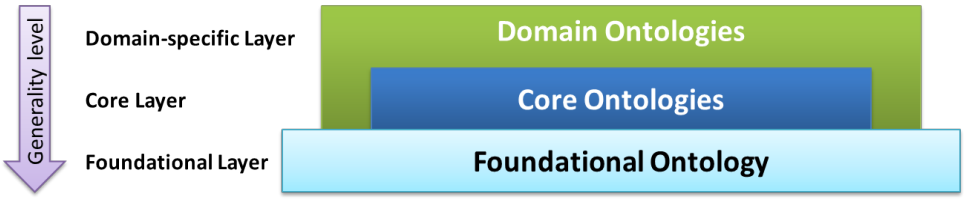



Figura 6 – Arquitetura SEON. Fonte:(2)

### 2.1.2 Ontologia de Fundamentação

As ontologias de fundamentação são sistemas de categorias formais independentes de domínio e filosoficamente bem fundamentadas, que podem ser usadas para expressar claramente conceituações do mundo real (28). Existem diversas ontologias de fundamentação, dentre elas, a Ontologia Fundacional Unificada (*Unified Foundational Ontology* - UFO) desenvolvida baseada em várias teorias das áreas de lógica filosófica, ontologias formais, linguística e psicologia cognitiva (32).

Ao longo dos anos, a UFO tem sido utilizada com sucesso em diferentes áreas, como no desenvolvimento de ontologias de núcleo e domínio de engenharia de software e na modelagem de exploração de petróleo (2) (32) (5). A capacidade da UFO de expressar cla-

ramente conceitos do mundo real, reduzindo ambiguidades conceituais, leva a comunidade científica, bem como os profissionais de modelagem conceitual, a considerar o UFO como um recurso importante para modelar as ontologias de domínio. Por isso, o seu sistema de categorias da UFO foi escolhido para embasar a modelagem apresentada neste trabalho.

No escopo desse trabalho somente são utilizadas alguns termos da UFO, conforme detalhados abaixo. Como todos os termos do sistema de categorias da UFO foram definidos em inglês, para manter a compatibilidade com a UFO, os diagramas ao longo do trabalho estão em inglês e no texto, sempre que possível, os elementos da UFO foram traduzidos para português com o seu nome original em inglês entre parênteses.

A UFO, inicialmente proposta em (33), reúne teorias axiomáticas que versam sobre as principais categorias de conceitos usados em modelagem conceitual. Uma distinção fundamental nesta ontologia está entre as categorias dos chamados urelementos (*Urelements*) e conjuntos (*sets*). Urelementos são entidades que não são conjuntos, mas podem ser elementos de um conjunto. Um urelemento deve ser um indivíduo ou um universal, mas não ambos (3). A Figura 7 mostra a distinção entre *Urelement* e *Set*.

Universais (*Universals*) e indivíduos (*Individuals*) são conceitos distintos que embasam a UFO. Os universais representam os aspectos gerais e comuns de diferentes indivíduos, ou seja, os universais tipificam os indivíduos. Assim, os indivíduos instanciam um ou mais universais. Os indivíduos representam as entidades que existem no mundo real e carregam uma identidade própria.

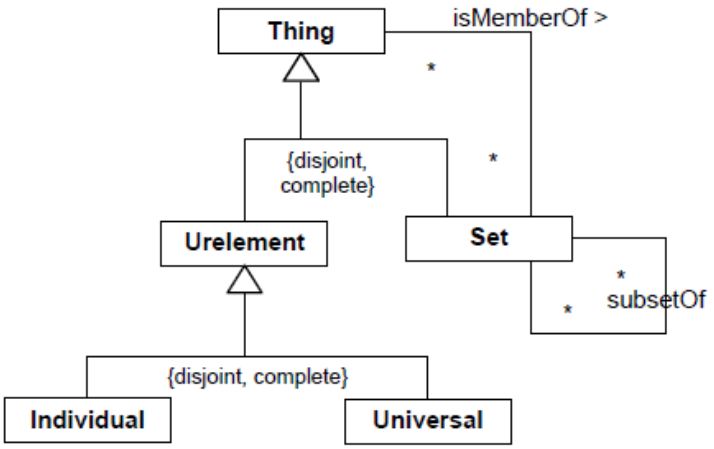



Figura 7 – Fundamental distinção entre *Urelement* e *Set*. Fonte: (3)

Os indivíduos se distinguem entre entidades endurantes (*Endurants*) e perdurantes (*Perdurants*). A distinção entre indivíduos endurantes e perdurantes pode ser feita em termos de seu comportamento em relação ao tempo. Endurantes estão totalmente presentes em qualquer instante do tempo que estiverem presentes, ou seja, eles são no tempo, no sentido de que, se em uma circunstância *c1*, um endurante (*Endurant*) *e* tem uma propriedade *p1* e na circunstância *c2* *e* tem uma propriedade a propriedade *p2*, é o mesmo

endurante (*Endurant*) *e* ao qual nos referimos em cada uma dessas situações. Então, endurantes (*Endurants*) tem diferentes partes que existem em diferentes tempos. Em contrapartida, os perdurantes (*Perdurants*) são indivíduos compostos de partes temporais, acontecem no tempo no sentido de que se estendem no tempo acumulando partes temporais. Sempre que um perdurante está presente, alguma de suas partes temporais está presente. Como consequência, os perdurantes não sofrem mudança no tempo em um sentido genuíno, pois nenhuma de suas partes temporais mantêm sua identidade ao longo do tempo (3). Enquanto endurantes (*Endurants*) existem no tempo, os perdurantes (*Perdurants*) acontecem no tempo (34).

Essa distinção entre endurantes e perdurantes deu origem a duas divisões da UFO: UFO-A e UFO-B. A UFO-A, que é o núcleo da ontologia, lida com os endurantes, com foco em aspectos estruturais da modelagem conceitual. O fragmento UFO-B enfoca nos perdurantes, lidando com eventos, processos, isto é, aspectos temporais e as possíveis conexões entre os endurantes e os perdurantes (32).

A UFO-A sistematiza conceitos como tipos e estruturas taxonômicas (35), relações todo-parte (36), propriedades intrínsecas e espaços de valores de atributos (37), propriedades relacionais (33), entre outros. Esse fragmento constitui uma teoria estável, formalmente caracterizada com o aparato de uma lógica modal de alta expressividade e possuindo forte suporte empírico promovido por experimentos em psicologia cognitiva (3).

Esta parte da ontologia trata dos endurantes (*Endurants*) que são classificados em substanciais (*Substantials*) e momentos (*Moments*). O momento é uma propriedade, evento ou processo individualizado que não faz parte da essência de um indivíduo (*Individual*) (3). Este conceito, na literatura é referenciado usando diversos termos, tais quais: *moment, trope, abstract particular, particular quality, individual accident* e *property instance*. Neste trabalho o momento será referenciado como momento (*Moment*) ou tropo (*Trope*).

Uma característica importante de todos os momentos (*Moments*) é que eles só podem existir em outros indivíduos (*Individuals*). Os momentos são existencialmente dependentes (*existentially dependent*) de outros indivíduos, nomeados seus portadores (*bearers*). A dependência existencial é uma condição necessária, mas não suficiente, para que algo seja um momento. Os momentos são como os indivíduos são (38) (39) e não podem ser concebidos independentemente dos indivíduos (*Individuals*) em que são inseridos (3).

Essa dependência pode ser de um único indivíduo, isto é, representam um momento intrínseco (*Intrisic Moment*) de um indivíduo. Esse momento intrínseco pode ser uma qualidade (*Quality*) como cor, peso, altura, carga elétrica, forma circular; modo (*Mode*) como um pensamento, uma habilidade, uma crença, uma intenção, uma dor de cabeça; bem como uma disposição (*Disposition*) como a disposição de um material magnético para atrair um objeto metálico (3).

Também há possibilidade do momento ser dependente existencialmente de uma pluralidade de indivíduos, neste caso são denominados de momentos relacionais (*Relacional Moments* ou *Relators*) (3). Esses indivíduos são aglutinados por relações que podem ser formais ou materiais (40). As relações materiais se caracterizam pelo relacionamento entre entidades mediados (*mediates*) por indivíduo chamado modo relacional (*Relator*). Os modos relacionais são indivíduos com o poder de conectar outros indivíduos (Regra 1) (4). As relações materiais têm estrutura material por si mesmas e incluem exemplos trabalhar em, estar matriculado em ou estar conectado a. Por exemplo, a matrícula é um modo relacional (*Relator*) que conecta um aluno a uma instituição educacional (40).

As relações formais, por outro lado, se mantêm diretamente entre duas ou mais entidades, sem qualquer outra entidade para mediar (40). Elas podem ser relações de comparação entre propriedades dos indivíduos relacionados. Assim, valem sempre que os indivíduos e as referidas propriedades existirem. Por exemplo, a relação "mais pesado que" entre João e José é verdade enquanto os dois indivíduos existirem e o peso de João for maior que o de José. A Figura 8 mostra a estrutura de endurante (*Endurant*).

$\underline{Regra\ 1} : \forall x\ \ Relator(x) \rightarrow \exists y, z\ \ Object(mediates(x, y) \wedge mediates(x, z) \wedge y \neq z)$

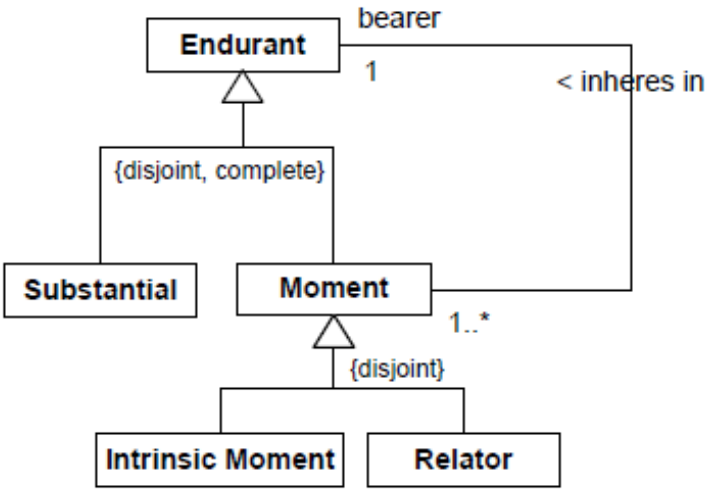



Figura 8 – *Endurant* e suas especializações. Fonte: (3)

O substancial (*Substantial*) é um endurante (*Endurant*) que não é inerente a outro endurante (*Endurant*), isto é, que não é um momento (*Moment*). Os substanciais, ao contrário dos momentos, são indivíduos existencialmente independentes. Substanciais são indivíduos que possuem momentos espaço-temporais e, qualidades que necessariamente são inerentes a um substancial, essas qualidades são chamadas de momentos do substancial (*Substantial Moments*).

Os substanciais podem ser classificados em *Amounts of Matter* ou objeto (*Object*). *Amounts of Matter* são substanciais cuja identidade é determinada pela soma de suas partes e, portanto, uma mudança em uma das partes altera sua identidade. Exemplos

de *Amounts of Matter* são indivíduos referenciados linguisticamente por substantivos em massa como açúcar, areia e ouro. Os objetos, ao contrário de *Amounts of Matter*, nem todas as partes dele serão essenciais. Exemplos de objetos são substantivos contáveis, tais como, um cachorro, uma casa, um martelo, um carro. A Figura 9 mostra as especializações de substancial (*Substantial*).

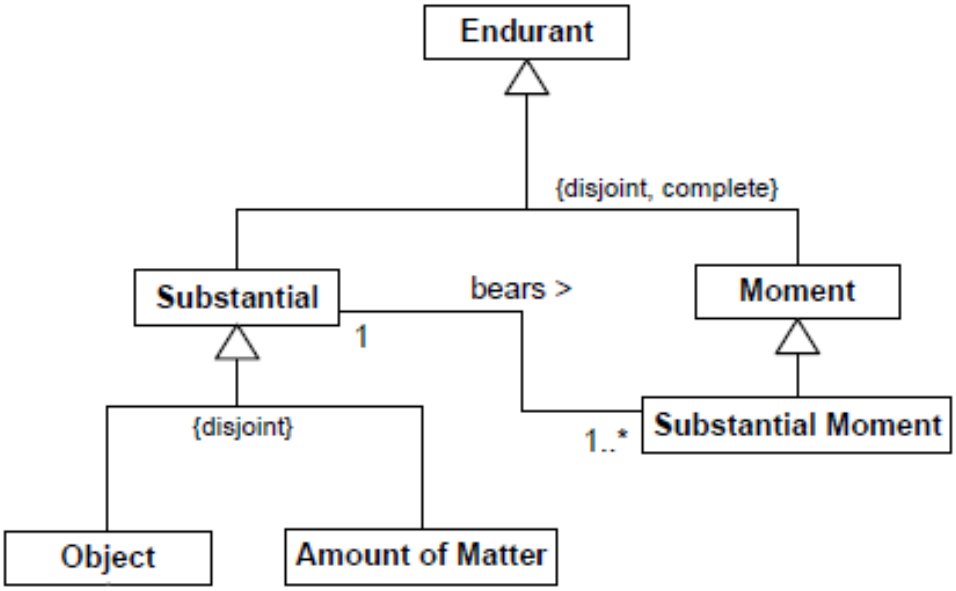



Figura 9 – *Substantial* e suas especializações. Fonte: (3)

O princípio de identidade de objetos é fornecido pelos tipos que eles instanciam. Esses tipos podem ser essenciais, obrigatórios ou contingentes para o objeto (3), definindo as propriedades essenciais e acidentais de um indivíduo (41) e possibilitando mudanças qualitativas em certos aspectos sem haver perda de identidade. Dentre os tipos, há o tipo *sortal* que fornece ou carrega um princípio uniforme de identidade para suas instâncias (4). O tipo rígido (*rigid*) classifica necessariamente suas instâncias, ou seja, as instâncias desse tipo não podem deixar de existir sem deixar de existir o indivíduo (4). O *Kind* é um tipo *sortal* rígido (*Rigid Sortal*) que fornece identidade para suas instâncias, por exemplo, Pessoa. Os subtipos (*Subkind*) são tipos *sortal* rígido (*Rigid Sortal*) que carregam o princípio de identidade fornecido por um único *Kind*, por exemplo, os subtipos (*Subkinds*) Homem e Mulher que carregam o princípio de identidade fornecido pela Pessoa (3).

A anti-rigidez (*AntiRigid*), por outro lado, caracteriza um tipo cujas instâncias podem se mover para dentro e para fora de sua extensão sem alterar sua identidade (3). O tipo fase (*Phase*) é um tipo *sortal* anti-rígido (*AntiRigid Sortal*) instanciado apenas ocasionalmente devido a alteração de uma propriedade intrínseca (3). Por exemplo, Criança é uma fase da Pessoa, instanciada por instâncias de pessoas que têm a propriedade intrínseca de possuir menos de 12 anos de idade. Os papéis (*Roles*), por outro lado, são do tipo *sortal* anti-rígido e dependentes relacionais que capturam propriedades relacionais compartilha-

das por instâncias de um determinado tipo. As entidades desempenham papéis quando relacionadas com outras entidades por meio das chamadas relações materiais, denominado papel relacional (*Relational Role*), ou quando participam de eventos, denominado papel processual (*Processual Role*). Por exemplo, João e Maria são casados, o casamento é um relacionamento em que João desempenha o papel relacional (*Relational Role*) de marido e Maria desempenha o papel relacional (*Relational Role*) de esposa. E, no evento casamento, o João participa desempenhando o papel processual (*Processual Role*) de marido e a Maria o papel processual (*Processual Role*) de esposa.

Os tipos não *sortal* (*NonSortal*) agregam propriedades comuns a diferentes indivíduos do tipo *sortal*, isto é, classificam diferentes entidades. Eles não fornecem um princípio uniforme de identidade para suas instâncias; em vez disso, apenas classificam coisas que compartilham propriedades comuns, mas têm diferentes princípios de identidade. As propriedades de rigidez e anti-rigidez também podem ser aplicadas para distinguir diferentes indivíduos não *sortal*. Uma categoria (*Category*) representa um tipo não *sortal* rígido (*Rigid NonSortal*) e relacionalmente independente, isto é, um tipo dispersivo que agrega propriedades essenciais comuns a diferentes indivíduos do tipo *sortal* rígido (*Rigid Sortal*) (3). Por exemplo, a categoria objeto físico agrega propriedades essenciais de tipos de mesas, carros, óculos. Uma mistura de papéis (*Role Mixin*) representa um tipo não *sortal*, anti-rígido (*AntiRigid Sortal*) e relacionalmente dependente, ou seja, um tipo dispersivo que agrega propriedades comuns de diferentes papéis (*Roles*). Por exemplo, o *Role Mixin* cliente agrega propriedades de clientes individuais e clientes corporativos (3) (4). A Figura 10 mostra os tipos de endurante (*Endurant*).

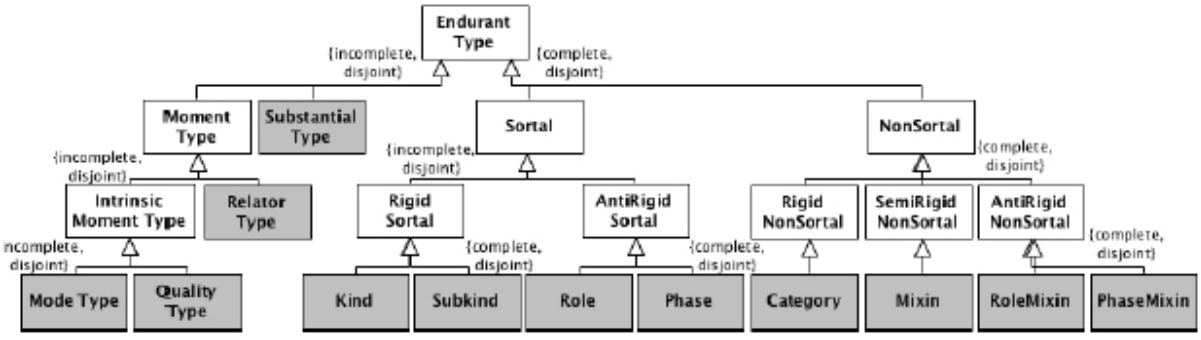



Figura 10 – Tipos de *Endurant*. Fonte: (4)

A Figura 10 representa as propriedades dos endurantes (*Endurants*) rígidas (*Rigid*) e as propriedades que tem a possibilidade de mudar (*AntiRigid*). Visando descrever como essas mudanças ocorrem, foi desenvolvida a ontologia para trata esses eventos denominada UFO-B. Os eventos (*Events*) são definidos como relações entre estados de coisas, ou seja, cada evento realiza o mapeamento de uma situação (*Situation*) no mundo para uma outra, na qual os endurantes (*Endurants*) são caracterizados por apresentar certas

propriedades. Entre essas mudanças, os eventos podem ser ocasionados por situações ou ocasionar situações nas quais os objetos (*Objects*) são criados, deixam de existir, mudam suas propriedades ou simplesmente participam desempenhando certo papel processual (*Processual Role*).

A UFO-B é responsável por eventos como manifestações de disposições de objetos e explica como os objetos, situações e eventos se relacionam (5). O evento pode ser visto como uma entidade extensional definida pela soma de suas partes (34), isto é, como uma sucessão de mudanças no mundo (41) que ocorrem em um intervalo temporal. Por ser uma entidade que acontece no tempo, no sentido de que se estendem no tempo acumulando partes temporais, o evento (*Event*) também é conhecido como perdurante (*Perdurant*) e pode ser composto de outros eventos. Um evento é atômico (*Atomic Event*) se não tiver partes (*hasPart*) (Regra 2) e, caso contrário, complexo (*Complex Event*) (Regra 3) (5).

$\underline{Regra\ 2} : \forall e : Event(AtomicEvent(e) \leftrightarrow \neg\exists$ e':Event(hasPart(e, e')))

$\underline{Regra\ 3} : \forall e : Event(ComplexEvent(e) \leftrightarrow \neg AtomicEvent(e))$

O evento complexo (*Complex Event*) tem outros eventos como parte mas, ele não pode possuir ele mesmo como sua parte (*hasPart*) (Regra 4). Um evento complexo (*Complex Event*) *e* pode possuir outro evento complexo (*Complex Event*) *e'* como parte (*hasPart*) desde que o evento complexo (*Complex Event*) *e* não seja parte de evento complexo (*Complex Event*) *e'* (Regra 5). E, se o evento complexo (*Complex Event*) *e* tiver um evento complexo (*Complex Event*) *e'* como sua parte (*hasPart*) e, esse evento complexo (*Complex Event*) *e'* tiver um evento (*Event*) *e"* como parte (*hasPart*), então evento (*Event*) *e"* também é parte (*hasPart*) do evento complexo (*Complex Event*) *e* (Regra 6) (5).

$\underline{Regra\ 4} : \forall e : ComplexEvent(\neg hasPart(e, e))$

$\underline{Regra\ 5} : \forall e,$ e': ComplexEvent(hasPart(e, e') $\rightarrow \neg$hasPart(e', e))

$\underline{Regra\ 6} : \forall e,$ e': ComplexEvent; e": Event((hasPart(e, e') $\wedge$ hasPart(e', e")) $\rightarrow$ hasPart(e, e"))

Os eventos são manifestações de propriedades dos objetos, em particular, de disposições. As disposições incluem propensões, capacidades, responsabilidades expressas em situações particulares. Por exemplo, o evento "Dengue de João" tem o objeto João que está na situação doente e manifesta diversas disposições, tais como, febre, dor no corpo etc (41). O evento atômico (*Atomic Events*) é a manifestação de (*manifestedBy*) uma disposição (*Disposition*) de um único objeto (*Object*) (Regra 7). A disposição (*Disposition*) é uma propriedade que está presente no (*inheresIn*) objeto (*Object*) (Regra 8) e, se manifesta, em situações que levam a ocorrência de eventos ou situações ocasionadas por eventos (34). Logo, se uma disposição (*Disposition*) está presente em (*inheresIn*) um objeto (*Object*) e é manifestada em (*manifestedby*) um evento atômico (*Atomic Event*), então esse evento

atômico (*Atomic Event*) depende (*dependsOn*) desse objeto (*Object*) (Regra 9). Sendo um evento atômico (*Atomic Event*), ele depende de (*dependsOn*) um único objeto (*Object*) (Regra 10). Finalmente, pode-se concluir que todo evento atômico depende exclusivamente (*exclusivelyDependsOn*) do objeto se e somente se depende do (*dependsOn*) objeto (*Object*) (Regra 11) (5).

$\underline{Regra\ 7} : \forall e : AtomicEvent(\exists! d : Disposition(manifestedBy(d, e)))$

$\underline{Regra\ 8} : \forall d : Disposition(\exists! o : Object(inheresIn(d, o)))$

$\underline{Regra\ 9} : \forall d : Disposition; e : AtomicEvent; o : Object((manifestedBy(d, e)$

$\wedge inheresIn(d, o)) \rightarrow dependsOn(e, o))$

$\underline{Regra\ 10} : \forall e : AtomicEvent\ \exists! o : Object(dependsOn(e, o))$

$\underline{Regra\ 11} : \forall e : AtomicEvent; o : Object(exclusivelyDependsOn(e, o) \leftrightarrow$

$dependsOn(e, o))$

Os eventos complexos como são compostos por partes, então, manifestam várias disposições. E, de forma análoga ao evento atômico, os eventos complexos (*Complex Event*) dependem exclusivamente (*exclusivelyDependsOn*) dos objetos (*Objects*) dos quais suas partes dependem exclusivamente (Regra 12). Um evento complexo pode manifestar disposições de diferentes objetos, ou seja, ele pode ter a participação de vários objetos. A participação captura a manifestação de um único objeto em um evento, sendo assim, a participação (*Participation*) é um evento (*Event*) que depende exclusivamente (*exclusivelyDependsOn*) de um único objeto (*Object*) (Regra 13). E, esse objeto (*Object*) participa desse (*participationOf*) participação (*Participation*) (Regra 14) (5).

$\underline{Regra\ 12} : \forall e : ComplexEvent, o : Object(exclusivelyDependsOn(e, o) \leftrightarrow$

$\forall$ e': Event(hasPart(e, e') $\rightarrow$ exclusivelyDependsOn(e', o)))

$\underline{Regra\ 13} : \forall e : Event(Participation(e) \leftrightarrow \exists! o : Object(exclusivelyDependsOn(e, o)))$

$\underline{Regra\ 14} : \forall o : Object,\ p : Participation(participationOf(p, o) \leftrightarrow$

exclusivelyDependsOn(p, o))

O objeto (*Object*), ao participar do evento, desempenha um papel nesse processo. O papel processual (*Processual Role*) é uma especialização de papel (*Role*) usada para representar o papel desempenhando pelo objeto ao participar do evento. Como todo indivíduo (*Individual*), evento (*Event*) é instância de evento universal (*Event Universal*) e o evento universal (*Event Universal*) tem como especialização própria a participação universal (*Participation Universal*). As participações universais (*ParticipationUniversals*) são universais (*Universals*) que classificam os eventos (*Events*). Sendo assim, um evento atômico (*Atomic Event*) depende (*dependsOn*) do objeto (*Object*) que desempenha (*plays*) um Processual Role (*Processual Role*) induzo (*inducedBy*) por uma participação universal

(*Participation Universal*). A instância dessa participação universal pode ser um evento atômico (*Atomic Event*) ou uma participação (*Participation*) que tenha como parte (*hasPart*) o evento atômico (*Atomic Event*) (Regra 15) (5).

$\underline{Regra\ 15} : \forall o : Object,\ \ r : ProcessualRole(plays(o, r) \rightarrow (\exists p : AtomicEvent, u : ParticipationUniversal(dependsOn(p, o) \wedge inducedBy(r, u) \wedge (instantiates(p, u) \vee (\exists c : Participation(hasPart(c, p) \wedge instantiates(c, u)))))))$

Conforme dito anteriormente nesta seção, o evento (*Event*) é a manifestação da (*manifestedBy*) disposição (*Disposition*) do objeto. Essa disposição (*Disposition*) está presente (*presentIn*) na situação (*Situation*) que aciona (*triggers*) o evento (*Event*) e na situação (*Situation*) que é desencadeada (*bringsAbout*) pelo evento (*Event*) (Regra 16) (5).

$\underline{Regra\ 16} : \forall e : Event, d : Disposition(manifestedBy(d, e) \rightarrow$

(∀ s, s':Situation((triggers(s, e) → presentIn(d, s)) ∧ (bringsAbout(e, s') →

presentIn(d, s')))))

Todos os eventos (*Events*) são acionados (*triggers*) por uma situação (*Situation*) (Regra 17) e cada evento (*Event*) desencadeia (*bringsAbout*) uma situação (*Situation*) (Regra 18). Quando o evento (*Event*) ocasiona (*brings-about*) uma situação (*Situation*), a situação (*Situation*) ocorre (*obtainsIn*) no mesmo instante de tempo do final do evento (*end-point*) (Regra 19) e, a situação (*Situation*) desencadeadora (*triggers*) do evento (*Event*) representa o estado do mundo necessário para a manifestação do evento, isto é, a situação que o mundo se encontra quando o evento se inicia (*begin-point*) (Regra 20). A situação (*Situation*) obtida em (*obtainIn*) um ponto específico do tempo (*TimePoint*), denomina-se fato (fact) (Regra 21) (5). O evento e os seus elementos usados para descrevê-lo estão ilustrados nas Figuras 11.

$\underline{Regra\ 17} : \forall e : Event(\exists! s : Situation(triggers(s, e)))$

$\underline{Regra\ 18} : \forall e : Event(\exists! s : Situation(bringsAbout(e, s)))$

$\underline{Regra\ 19} : \forall s : Situation, e : Event(brignsAbout(s, e) \rightarrow$

$\exists t : TimePoint(obtainsIn(s, t) \wedge endPoint(e, t)))$

$\underline{Regra\ 20} : \forall s : Situation, e : Event(triggers(s, e) \rightarrow obtainsIn(s, t) \wedge beginPoint(e, t)))$

$\underline{Regra\ 21} : \forall s : Situation(Fact(s) \leftrightarrow \exists! t : TimePoint(obtainsIn(s, t)))$

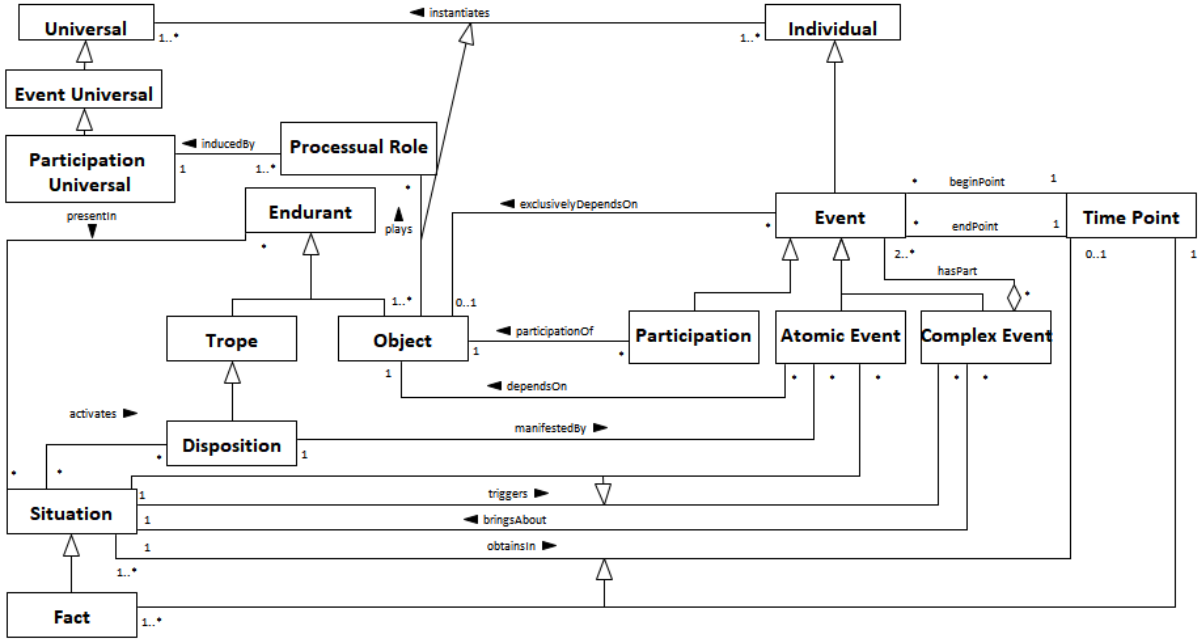



Figura 11 – *Event* e seus elementos. Adaptado de: (5)

Essas categorias da UFO foram usadas para representar as entidades do domínio de Incidente de Segurança da Informação com enfoque em evidenciar o contexto no qual o incidente ocorreu, os participantes dos envolvidos e seus papéis e as consequências do incidente.

### 2.1.3 Teoria multi-nível

Nas últimas décadas, tem havido um crescente interesse no uso de ontologias de fundamentação para fornecer uma base teórica sólida para a modelagem conceitual. Esta abordagem oferece suporte aos tipos cujas instâncias são indivíduos do domínio, porém não provê forma de representar tipos de tipos do domínio ou categorias de categorias. Para suprir esta necessidade foi criada a *Multi-Level Theory* (MLT), uma teoria multi-nível (42).

A MLT considera tipos que possuem outros tipos como instâncias. Para acomodar essas variedades de tipos utiliza noção de ordem de tipos. Tipos tendo indivíduos como instâncias são chamados de tipos de primeira ordem, tipos cujas instâncias são tipos de primeira ordem são chamados de tipos de segunda ordem e assim por diante. Para vincular tipos às entidades que se enquadram nesses tipos, utiliza-se a primitiva instância de (*instance of*), ou simplesmente iof, representada por um predicado ternário iof($e$, $t$, $w$) no qual uma entidade $e$ é uma instância de uma entidade $t$, isto é, um tipo em um mundo $w$. A relação de instanciação com mundos possíveis permite um suporte à classificação dinâmica, admitindo assim tipos que se aplicam eventualmente a suas instâncias. Por exemplo, João é uma instância de estudante em $w$, mas não em $w'$, quando ele se formou.

A noção de tipos e indivíduos é central em MLT. Tipos são entidades predicativas que podem ser aplicadas a múltiplas entidades, incluindo a si mesmas. As entidades, que não são tipos, são considerados indivíduos e são identificadas através da constante *Individual*. Assim, uma entidade é uma instância de *Individual*, se e somente se ele não pode estar relacionada a uma outra entidade através de instanciação. A constante *First-Order Type*, abreviada como 1stOT, caracteriza o tipo que se aplica a todas as entidades cujas instâncias são instâncias de *Individual* (Regra 1). Analogamente, cada entidade cuja extensão possível contenha exclusivamente instâncias de 1stOT é uma instância do *Second-Order Type* (2ndOT) (Regra 2). Sendo assim, *Individual* é a instância de 1stOT que, por sua vez, é a instância de 2ndOT. E os tipos *Individual*, 1stOT e 2ndOT são chamados de tipos básicos da MLT. Cada possível entidade deve ser uma instância de exatamente um dos tipos básicos. No entanto, esse esquema pode ser estendido para considerar quantas ordens forem necessárias (6).

Regra 1: iof(Individual, 1stOT)

Regra 2: iof(1stOT, 2ndOT)

Algumas relações estruturais para suportar a modelagem conceitual são definidas, começando com a especialização comum entre os tipos. Um tipo *t* especializa outro tipo *t'* se, em todos os mundos possíveis, todas as instâncias de *t* também forem instâncias de *t'* (Regra 3). De acordo com essa definição, todo tipo se especializa. Como isso pode ser indesejado em alguns contextos, a relação de especialização própria (*proper specialization*) foi definida para que um tipo *t* seja especialização própria (*proper specialization*) de *t'* se e somente se *t* especializa *t'* e *t* é diferente de *t'*(Regra 4). Essas relações são mantidas somente entre tipos da mesma ordem (6).

Regra 3: $\forall$ t,t' specializes(t,t') $\leftrightarrow (\exists y \ \ iof(y,t) \wedge (\forall e \ \ iof(e,t) \rightarrow$ iof(e,t')))

Regra 4: $\forall t, t$ properSpecializes(t, t') $\leftrightarrow$ (specializes(t, t')$\wedge t \neq$ t')

Todo tipo que não é do tipo básico, como um tipo do domínio, é uma instância de um dos tipos básicos de alta ordem (por exemplo, 1stOT, 2ndOT) e, ao mesmo tempo, é especialização própria (*proper specialization*) do tipo básico no nível imediatamente inferior (respectivamente, *Individual* e 1stOT) (Regra 5 e 6). Por exemplo, considerando uma Pessoa (*Person*) como um indivíduo, ela é instância de 1stOT e uma especialização própria (*proper specialization*) de *Individual*. As instâncias de Fase da pessoa pela idade (*Person Age Phase*) são especializações próprias de Pessoa (*Person*), por exemplo, criança (*Child*) e adulto (*Adult*). Assim, a Fase da Pessoa pela idade (*Person Age Phase*) é uma instância de (*instance of*) de 2ndOT e especialização própria (*proper specialization*) de 1stOT (6).

$Regra\ 5 : \forall t \ \ iof(t, 1stOT) \leftrightarrow specializes(t, Individual)$

$Regra\ 6 : \forall t \ \ iof(t, 2ndOT) \leftrightarrow specializes(t, 1stOT)$

Além das relações de instanciação e especialização, há também a relação de subordinação (*subordinate*). A subordinação entre dois tipos de ordem superior implica especializações entre suas instâncias, ou seja, um tipo *t* é subordinado (*subordinate*) a um tipo *t'* se todas as instâncias de (*iof*) *t* são especializações próprias (*proper specialization*) das instâncias de (*iof*) *t'* (Regra 7). Como subordinação (*subordinate*) implica em especialização própria (*proper specialization*) entre as instâncias dos tipos envolvidos em uma ordem mais baixa, a subordinação só pode ser mantida entre tipos de ordem superior da mesma ordem. Por exemplo, a subordinação pode ser usada para representar a relação entre universais da taxonomia da UFO. Na UFO, cada subtipo (*SubKind*) deve ser especialização de um *Kind* então subtipo (*Subkind*) é subordinado ao (*isSubordinateTo*) *Kind*.

$Regra\ 7 : \forall$ t, t' isSubordinateTo (t, t') $\leftrightarrow (\exists x \ \ iof(x,t) \wedge (\forall$ t" iof(t", t) $\rightarrow$

($\exists$ t"' iof(t"', t') $\wedge$ properSpecializes(t", t"'))))

Os tipos da MLT são representados usando a notação de classe da linguagem de

modelagem unificada (*Unified Modeling Language* - UML), as associações são usadas para representar relações entre instâncias e tipos e setas tracejadas representam relações entre os tipos, com rótulos para denotar os nomes dos predicados que se aplicam. A Figura 12 ilustra esse padrão.

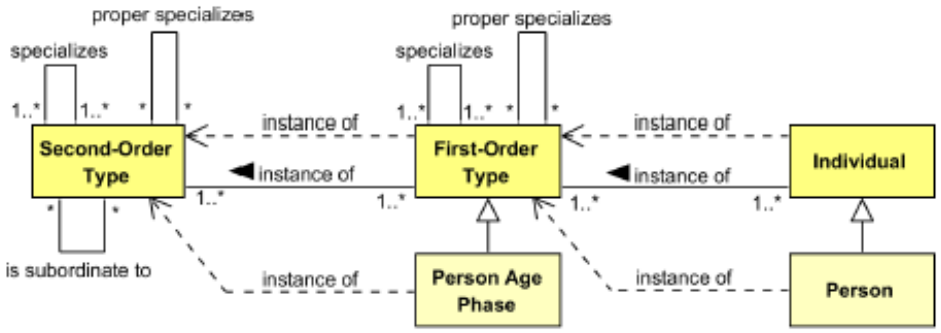



Figura 12 – Ilustrando as relações estruturais intra-níveis usando o padrão da MLT. Fonte: (6)

Até agora, foram apresentadas as relações intra-níveis, ou seja, aquelas que ocorrem entre entidades da mesma ordem. Porém, há relações estruturais de nível cruzado entre os tipos de ordens adjacentes. Essas relações suportam uma análise das noções de supertipo (*powertype*) entre um tipo de ordem mais alta e um tipo de base em uma ordem mais baixa. De forma que, um tipo $t$ é supertipo de um tipo base $t'$ se todas as instâncias de $t$ especializam $t'$ e todas as especializações possíveis de $t'$ são instâncias de $t$. Assim, um tipo $t$ é supertipo de $t'$ quando as instâncias de $t$ são válidas para as instâncias de $t'$, mas $t$ não define critérios de classificação. E, todas as especializações de $t'$, incluindo $t'$ em si são instâncias de $t$. Por exemplo, considere um tipo chamado supertipo de pessoa (*Person Powertype*) de forma que todas as possíveis especializações de pessoa (*Person*) sejam instâncias dele e, inversamente, todas as suas instâncias especializem pessoa (*Person*). Neste caso, *Person Powertype* é o supertipo (*powertype*) de *Person*. Como pessoa (*Person*) é instância de 1stOT, o supertipo de pessoa (*Person Powertype*) é instância de 2ndOT e especialização de 1stOT, conforme ilustrado na Figura 13. De acordo com a definição de supertipo (*powertype*), 1stOT é supertipo de (*isPowertypeOf*) indivíduo (*Individual*) e, analogamente, 2ndOT é supertipo de (*isPowertypeOf*) 1stOT (Regra 8 e 9).

Regra 8: isPowertypeOf(1stOT, Individual)

Regra 9: isPowertypeOf(2ndOT, 1stOT)

Visando representar os critérios de classificação, a relação de categorização entre tipos de níveis adjacentes pode ser modelada. A relação de categorização ocorre entre um tipo de ordem superior $t$ e um tipo de base $t'$ quando todas as instâncias de $t$ especializam $t'$ de acordo com critérios de classificação específicos. Sendo assim, um tipo $t$ categoriza (*categorizes*) um tipo base $t'$ se todas as instâncias de $t$ forem especialização própria

(*proper specialization*) de *t'* (Regra 10). E, se as instâncias de *t* especializam *t'*, mas *t'* não é uma instância de *t* e pode haver outros tipos que especializam *t'* de acordo com outros critérios de classificação, portanto, não são instâncias de *t*. Por exemplo, na Figura 13, o papel da pessoa (*Person Role*), que tem como instâncias gerente (*Manager*) e pesquisador (*Researcher*), categoriza (*categorizes*) pessoa (*Person*), mas não é um supertipo (*powertype*) de pessoa (*Person*), já que existem especializações de pessoa (*Person*) que não são papéis de pessoa (*Person Role*), por exemplo, criança (*Child*) e adulto (*Adult*).

Regra 10: ∀ t,t' categorizes (t,t') ↔ (∃ x iof(x,t) ∧(∀ t" iof(t",t) → properSpecializes(t",t')))

A relação de categorização tem algumas variações úteis para capturar mais restrições em um modelo de vários níveis. Um tipo pode ser considerado categorizado completamente (*completelyCategorizes*) quando o tipo *t* categoriza (*categorizes*) cada instância de *t'* e cada instância de *t'* é instância de, pelo menos, uma instância de *t* (Regra 11). Além disso, *t* categoriza *t'* se somente se cada instância de *t'* é instância de, no máximo, uma instância de *t*, diz-se que *t* categoriza de forma disjunta (*disjointlyCategorizes*) *t'* (Regra 12). Finalmente, um uso comum da noção de supertipo (*powertype*) considera um tipo de ordem superior que, simultaneamente, categoriza completamente (*completelyCategorizes*) e categoriza de forma disjunta (*disjointlyCategorizes*) um tipo de ordem mais baixa. Para capturar essa noção, foi definida a relação de partições (*partitions*). Assim, *t* particiona (*partitions*) *t'* se e somente se *t* categoriza (*categorizes*) *t'* e cada instância de *t'* é instância de exatamente uma instância de *t* (Regra 13). Por exemplo, considere o tipo Fase da pessoa pela idade (*Person Age Phase*) que particiona (*partitions*) Pessoa (*Person*) e tem como instâncias criança (*Child*) e adulto (*Adult*), conforme ilustrado na Figura 13.

Regra 11: ∀ t, t' completelyCategorizes(t, t') ↔ (categorizes(t, t') ∧(∀*e* iof(e,t') → ∃ t" (iof(e,t") ∧ iof(t",t))))

Regra 12: ∀ t, t' disjointlyCategorizes (t, t') ↔ (categorizes(t, t') ∧ ∀ e, t", t"' ((iof(t", t) ∧ iof(t"', t) ∧ iof(e, t") ∧ iof(e, t"')) → t" = t"')))

Regra 13: ∀ t, t' partitions(t, t') ↔ (completelyCategorizes(t, t') ∧ disjointlyCategorizes(t, t'))

Esses padrões de classificação em multi-nível foram usados para representar as categorias de ataque de acordo com critérios de classificação, visando diminuir a ambiguidade conceitual inerente ao domínio.

### 2.1.4 Combinando UFO com MLT

A MLT define novas relações estruturais inspiradas no conceito de supertipo (*powertype*) como a relação de categorização e suas variações para enriquecer a expressi-

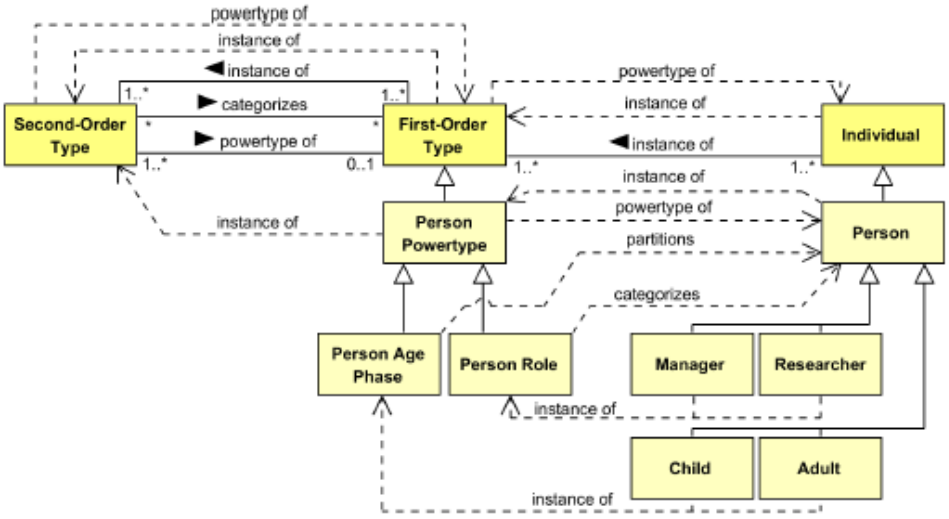



Figura 13 – Ilustrando relação estrutural entre níveis adjacentes usando o padrão da MLT. Fonte:(6)

vidade da modelagem. Nesse sentido, para aumentar os benefícios do uso da ontologia fundamentação para domínios que requerem múltiplos níveis de classificação, a MLT pode ser aplicada em conjunto com a UFO (43). Assim, modelos conceituais construídos com a combinação UFO-MLT têm proveito do sistema de categorias ontológicas empregado pela UFO e dos padrões de classificação em múltiplos níveis formalmente caracterizados na MLT (43).

Para combinar MLT e UFO, em (43) os autores estabeleceram uma hierarquia de modelos conceituais usando os conceitos da UFO-A com a MLT, formando uma camada mais alta. Além dos elementos UFO-A, em (7) os elementos da UFO-B também foram modelados para servir de base para representação de eventos no domínio da segurança da informação. Modelos conceituais construídos com a combinação UFO-MLT têm que seguir as regras de ambas as teorias. Essa combinação possibilita a construção de modelos capazes de expressar as propriedades ontológicas dos tipos que se aplicam aos indivíduos e representam os tipos de tipos específicos de um domínio.

Os conceitos de indivíduos (*Individuals*) da taxonomia UFO são instâncias de 1stOT e especializações de *Individual* da MLT. E, os conceitos universais (*Universals*) da taxonomia de UFO são instâncias de 2ndOT e especializações de 1stOT da MLT. Na UFO, cada entidade na taxonomia de indivíduos, existe uma entidade correspondente na taxonomia dos universais. As instâncias de uma entidade na taxonomia dos universais especializam a entidade correspondente na taxonomia dos indivíduos. Assim na UFO-MLT, *Endurant Universal* categoriza (*categorizes*) *Endurant*, *Event Universal* categoriza (*categorizes*) *Event*, *Moment Universal* categoriza (*categorizes*) *Moment* e assim por diante. A Figura 14 mostra as relações gerais de categorização entre universal (*Universal*) e indivíduo (*Individual*) do fragmento UFO que será usado neste trabalho.

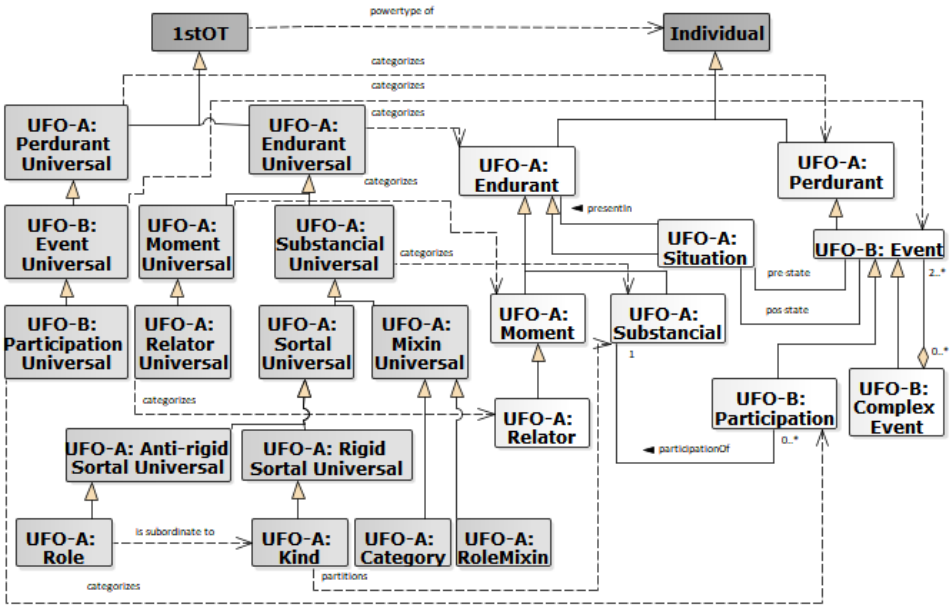



Figura 14 – Modelo conceitual UFO-MLT. Fonte:(7)

Essa estrutura viabiliza especificar melhor as regras de relacionamento entre *Universal* e *Individual*. Na Figura 14, uma vez que cada instância de substancial (*Substantial*) é uma instância de exatamente um *Kind*, seguindo a MLT, *Kind* particiona (*partitions*) substancial (*Substantial*). Além disso, a instância de um papel (*Role*) é especialização própria (*proper specialization*) de um *Kind*. Assim, no termo da MLT, *Role* está subordinado ao (*is subordinate to*) *Kind*.

Para construir modelos conceituais usando a combinação UFO-MLT devem ser seguidas as regras de ambas as teorias. Os tipos de primeira ordem do modelo devem ser especializações de indivíduo (*Individual*) e ser instâncias de um dos elementos folha da taxonomia UFO de universal (*Universal*). Se for necessário definir tipos de segunda ordem do domínio, eles devem ser especializações de uma das categorias taxonômicas de universal. E, para esclarecer qual o tipo de primeira ordem é instancia de um tipo de segunda ordem, cada tipo de segunda ordem do domínio tem um relacionamento de nível cruzado com o tipo de primeira ordem. Desta forma, facilita a explicitação do critério de classificação que suas instâncias devem aplicar para especializar o tipo base. Como resultado, a combinação UFO-MLT pode ser utilizada para fornecer regras e padrões para a introdução de tipos de segunda ordem em modelos de domínio baseados em ontologia (7).

Os benefícios da UFO-MLT apresentados nesta seção podem ser usados para construir um modelo de representação de Incidente de Segurança da Informação mais preciso que supra as necessidades do domínio de ter uma conceitualização clara que seja utilizada em consenso e exprimir as categorias de ataque de acordo com critérios de classificação.

## 2.2 Metodologia para construção de ontologias

Uma grande quantidade de ontologias foi desenvolvida por diferentes grupos, sob diferentes abordagens e usando diferentes métodos e técnicas. No entanto, alguns trabalhos foram publicados sobre como proceder, mostrando as práticas, os critérios de design, as atividades, as metodologias, as ferramentas usadas para construí-las (44). Nesta seção serão apresentadas as metodologias usadas para construir ontologias que inspiraram a elaboração da metodologia DEFESA.

### 2.2.1 Methontology

Methontology constitui-se de uma metodologia bem estruturada para construir ontologias a partir do zero. Ela inclui um conjunto de atividades, técnicas para realizar cada uma delas, e entregáveis a serem produzidos por tais atividades (44).

Antes de construir uma ontologia, a metodologia recomenda o planejamento das principais tarefas a serem realizadas, como elas serão organizadas, quanto tempo será necessário para realizá-las e com quais recursos. E, complementarmente, devem ser definidos os requisitos levando em consideração o motivo pelo qual a ontologia está sendo construída, quais são seus usos pretendidos e os usuários finais.

O ciclo de vida da ontologia é composto pelas seguintes atividades técnicas: especificação, conceituação, formalização, implementação e manutenção. Esta metodologia possui ainda algumas atividades de suporte desempenhadas durante o ciclo de vida da ontologia que são: aquisição do conhecimento, integração, avaliação, documentação e gerenciamento de configuração. Desta forma, o processo de desenvolvimento de ontologias, conforme mostrado na Figura 15, visa oferecer suporte para atender as necessidades que motivaram a construção da ontologia, gerando como produto final uma ontologia documentada, avaliada e codificada em uma linguagem formal.

O ciclo de vida da ontologia se inicia com a especificação. Nesta atividade são definidos o objetivo, a finalidade e escopo da ontologia e produzido um documento de especificação escrito em linguagem natural, usando um conjunto de representações intermediárias ou usando perguntas de competência.

A maior parte da aquisição é feita simultaneamente com a fase de especificação de requisitos, e diminui à medida que o processo de desenvolvimento da ontologia avança. Especialistas, livros, manuais, figuras, tabelas e até outras ontologias são fontes usadas para elucidar o conhecimento usando um conjunto de técnicas, tais como: *brainstorming*, entrevistas, análise formal e informal de textos e outras ferramentas de aquisição de conhecimento.

Esse conhecimento do domínio deve ser estruturado em um modelo conceitual que

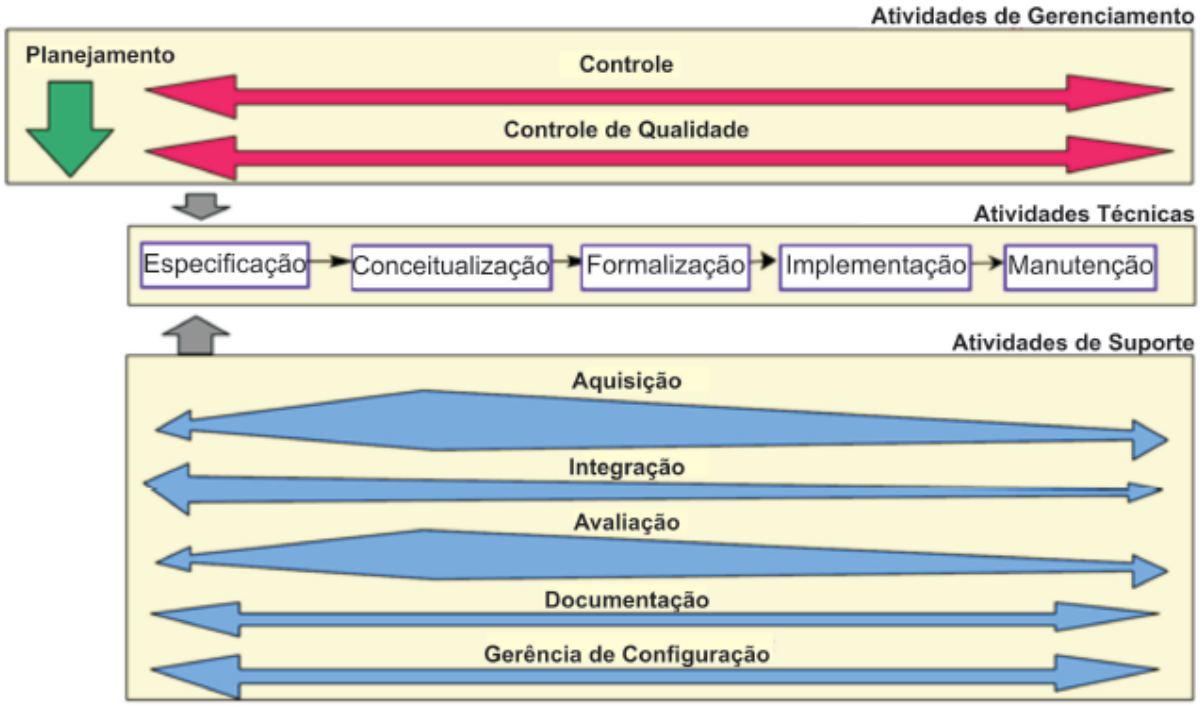



Figura 15 – Metodologia para construção de ontologia Methonlogy. Fonte: (8)

descreve o problema e sua solução em termos do vocabulário de domínio na atividade de conceitualização. Esta atividade envolve: construir glossário de termos; construir taxionomia de conceitos; construir diagrama ilustrando o relacionamento dos termos; construir dicionário de termos; descrever as relações entre os termos; descrever os atributos de instância; descrever os atributos das classes; descrever as constantes; elaborar os axiomas formais; descrever os papéis; e descrever as instâncias. Posteriormente, o modelo conceitual é formalizado e implementado, isto é, a ontologia deve ser codificada em uma linguagem formal. Neste ponto torna-se crucial a avaliação da ontologia. A avaliação consiste em realizar um julgamento técnico da ontologia com base no documento de especificação de requisitos e, na verdade, deve ocorrer durante cada fase e entre as fases do ciclo de vida.

Após a ontologia ser implementada inicia-se a etapa de manutenção, onde as alterações, quando necessárias, são realizadas para possíveis melhorias ou correções.

### 2.2.2 SABiO

A metodologia SABiO abreviação do termo em inglês *Systematic Approach to Build Ontologies*, uma abordagem sistemática para construir ontologias, foi criada em 1997 para apoiar o desenvolvimento de ontologias de referência de domínio. A abordagem parte do pressuposto que as ontologias de domínio devem ser desenvolvidas com base em ontologia de fundamentação. Conceitos e relações em uma ontologia de domínio devem ser previamente analisados à luz de uma ontologia de fundamentação (9). A ideia por trás da

análise ontológica é fornecer uma base sólida para os conceitos de modelagem, e assumir que tais conceitos visam representar a realidade (45).

O processo de desenvolvimento de SABiO compreende cinco fases principais: identificação da finalidade e elicitação de requisitos; captura e formalização da ontologia; projeto; implementação; e teste. Os processos de suporte são executados em paralelo ao processo de desenvolvimento envolvendo atividades de aquisição de conhecimento, documentação, gerenciamento de configuração, avaliação e reutilização (9). A Figura 16 apresenta uma visão geral de SABiO.

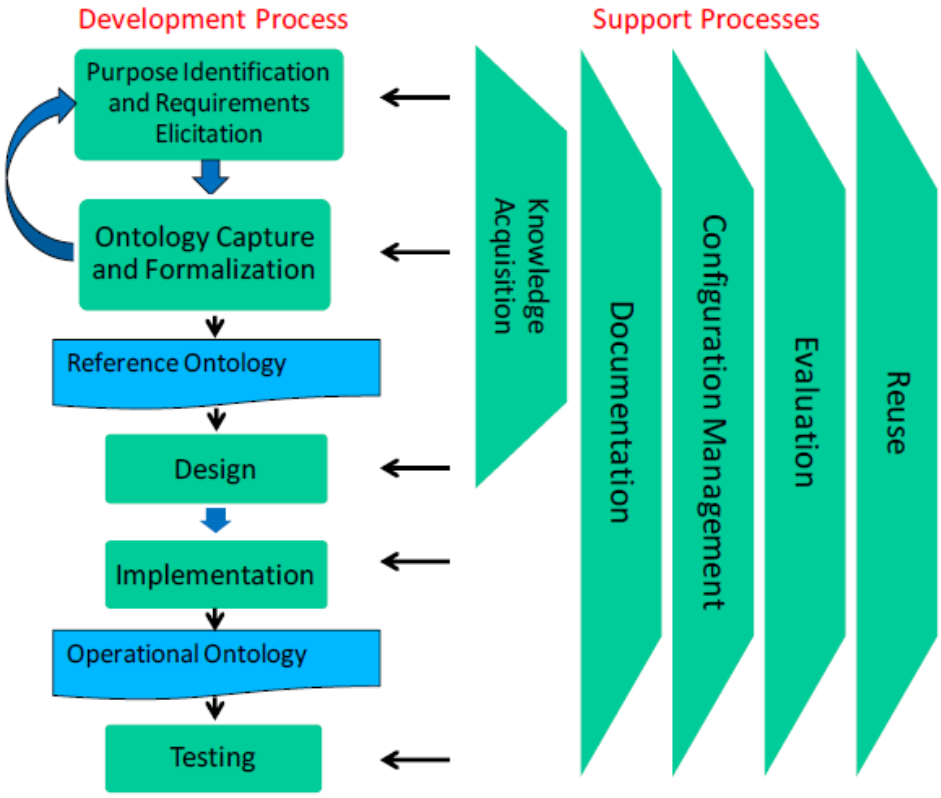



Figura 16 – Metodologia para construção de ontologia SABiO. Fonte: (9)

A primeira fase no processo de desenvolvimento de SABiO é a identificação da finalidade e a elicitação de requisitos. Inicialmente, precisa-se identificar o propósito da ontologia e seus usos pretendidos. Com base na finalidade inicialmente estabelecida para a ontologia e os usos pretendidos elicitados, são elencados os requisitos e as questões de competência são delineadas. O engenheiro de ontologia pode então identificar sub-ontologias e alocar as questões de competência às sub-ontologias identificadas. A crescente compreensão do domínio pode levar a uma melhor compreensão do propósito da ontologia e a identificação de novos usos pretendidos, iniciando um novo ciclo.

Seguindo para segunda fase do processo, a conceituação do domínio será capturada com base nas questões de competência de forma que sejam definidos os elementos da ontologia necessários e suficientes para responder às questões de competência. Essa fase

começa com a modelagem conceitual usando linguagens altamente expressivas para criar ontologias fortemente axiomatizadas que se aproximem o máximo possível da ontologia ideal do domínio. O foco dessas linguagens deve estar na adequação da representação dos principais conceitos e relações no domínio utilizando taxonomias. SABiO sugere que conceitos e relações em uma ontologia de domínio de referência sejam analisados à luz de uma ontologia de fundamentação. E, a formalização ocorre com o uso de axiomas informais escritos em uma linguagem formal.

A captura da ontologia é fortemente apoiada pelo processo de aquisição de conhecimento. O conhecimento pode ser extraído de especialistas do domínio, bem como de fontes de conhecimento consolidado, como livros, padrões internacionais e modelos de referência. Como resultado desta etapa uma ontologia de referência é produzida.

Porém, muitas vezes necessita-se obter uma versão operacional para ser usada por aplicativos de computador. Para alcançar essa versão operacional, precisa-se projetar e implementar em uma linguagem de ontologia legível por máquina. Nessa fase de projeto, a especificação conceitual da ontologia de referência deve ser transformada em uma especificação de projeto, levando em consideração uma série de questões, desde questões arquitetônicas e requisitos tecnológicos não funcionais, até o ambiente de implementação.

Na fase de projeto da ontologia a lista de requisitos tecnológicos não-funcionais deve ser complementada para a ontologia operacional e o ambiente no qual será implementado seja definido. Preparando para a fase de implementação a qual a ontologia será implementada na linguagem operacional escolhida.

Em SABiO, o teste de ontologia refere-se à verificação dinâmica e validação do comportamento da ontologia operacional em um conjunto finito de casos de teste, contra o comportamento esperado em relação às questões de competência. Um caso de teste compreende uma implementação de uma questão de competência como uma consulta no ambiente de implementação escolhido, dados de instanciação do fragmento da ontologia sendo testada e o resultado esperado baseado na instanciação considerada. O teste de validação também pode ser realizado usando a ontologia operacional em aplicações de software reais, de acordo com os usos pretendidos originalmente propostos para a ontologia.

As metodologias apresentadas possuem a mesma finalidade e, consequentemente, possuem muitas atividades semelhantes. O Quadro 1 mostra a correspondência entre as principais atividades das metodologias Methontology e SABiO.

Conforme visto no Quadro 1, as duas metodologias utilizam nomes distintos para as suas atividades, porém se assemelham em muitos pontos. Tal fato, reforça a importância de cada dessas atividades para a construção de uma boa ontologia. Sendo assim, visando desenvolver ontologia que atenda as necessidades do domínio de Incidente de Segurança da Informação as atividades dessas duas metodologias foram usadas para desenvolver uma

Quadro 1 – Correspondência entre as principais atividades das metodologias Methontology e SABiO

| Methontology | SABiO |
|---|---|
| Especificação | Identificação da finalidade e elicitação de requisitos |
| Conceitualização | Captura e formalização da ontologia |
| Formalização | Projeto |
| Implementação | Implementação |
| Manutenção | - |
| - | Teste |
| Aquisição | Aquisição do conhecimento |
| Integração | Reutilização |
| Avaliação | Avaliação |
| Documentação | Documentação |
| Gerenciamento de configuração | Gerenciamento de configuração |

metodologia específica para construir um sistema de apoio à decisão baseado em ontologia.

## 2.3 Sistema de apoio à decisão

As organizações utilizam sistemas transacionais para operacionalizar e manter seus negócios. Porém, quanto há necessidade de avaliar seus processos, faz-se necessário acompanhar o histórico de acontecimentos para identificar pontos que devem ser melhorados e pontos que estão evoluindo positivamente. Para que esta análise ocorra a contento deve ser planejado um processo de obtenção dos dados de sistemas de banco de dados legados e de transações e transformá-los em informações organizadas em um formato amigável, visando facilitar a análise de dados e apoiar a tomada de decisões de negócios baseadas em fatos.

Esse formato amigável é obtido através de um sistema de apoio à decisão que extraia, limpe, transforme e entregue os dados de origem em um armazenamento de dados dimensional, denominado *Data warehouse* (DW). No DW, os dados dos processo a ser analisado são representados através de fatos com seus respectivos descritores, as dimensões. A base para construção dessa estrutura do DW é a modelagem dimensional (MD). Segundo (46), as quatro principais decisões tomadas durante o desenvolvimento de um modelo dimensional incluem:

- Selecionar um processo do negócio;
- Definir a granularidade;
- Identificar as dimensões; e

- Identificar os fatos.

Essas decisões são tomadas considerando as necessidades do negócio e dados disponíveis. Seguindo o processo de negócio, o grão, a dimensão e as declarações de fatos, são determinados pelos nomes de tabela e coluna, os valores de domínio de amostra e as regras de negócios. Os representantes de governança de dados de negócios devem participar dessa atividade de desenvolvimento para garantir a adesão dos negócios.

Os processos de negócios são as atividades operacionais realizadas por uma organização, como fazer um pedido, processar uma reivindicação de seguro, registrar alunos para uma aula ou fazer *snapshots* de todas as contas a cada mês. Eventos de processo de negócio geram ou capturam métricas de desempenho que se traduzem em fatos em uma tabela de fatos. A maioria das tabelas de fatos se concentra nos resultados de um único processo de negócio. Escolher o processo é importante porque define um alvo de desenvolvimento específico e permite que o grão, as dimensões e os fatos sejam declarados. Cada processo corresponde a uma linha na tabela de fatos.

Sendo assim, declarar o grão é o passo crucial em um projeto dimensional. O grão estabelece exatamente o que uma única linha da tabela de fatos representa. A declaração de grãos se torna um contrato obrigatório no desenvolvimento do DW. O grão deve ser declarado antes de escolher dimensões ou fatos, porque toda dimensão ou fato candidato deve ser consistente com o grão.

Essa consistência impõe uma uniformidade em todos os projetos dimensionais que é essencial para o desempenho das consultas analísticas e para facilitar o uso. Grão atômico refere-se ao nível mais baixo em que os dados são capturados por um determinado processo de negócio. Recomenda-se começar com dados de granularidade atômica, pois resiste ao ataque de consultas imprevisíveis de usuários. Cada proposta de tabela de fatos resulta em uma tabela física separada e diferentes grãos não devem ser misturados na mesma tabela de fatos.

As dimensões fornecem o contexto (quem, o quê, onde, quando, por que e como) em torno de um evento de processo de negócio. As tabelas de dimensões contêm os atributos descritivos usados para filtrar e agrupar os fatos. Com o grão de uma tabela de fatos em mente, todas as dimensões possíveis podem ser identificadas. Sempre que possível, uma dimensão deve possuir um único valor quando associada a uma determinada linha de fatos.

A dimensão que existe em todo DW é a dimensão temporal (*temporal dimension*), pois os dados são armazenados numa série de *snapshots*, cada um representando um período de tempo. Então, a dimensão temporal (*temporal dimension*) se faz necessária para descrever quando ocorreu um fato.

Além da dimensão temporal, outras dimensões tipicamente existem em um DW. Por exemplo, a dimensão que descreve os motivos, o porquê da ocorrência do fato, isto

é, revela as condições que levaram o fato acontecer, chama-se dimensão causal (*causal dimension*). Também utiliza-se a dimensão status (*status dimension*) para representar as situações que ajudam a descrever como o fato ocorreu.

Muitas dimensões contêm mais de uma hierarquia natural. Por exemplo, as dimensões de data do calendário podem ter o dia da semana para a hierarquia do período fiscal, bem como uma hierarquia de dia para o mês e para o ano. Nesses casos, recomenda-se que as hierarquias coexistam na mesma tabela de dimensão para dispor de uma maior facilidade de uso e desempenho.

Uma única dimensão física pode ser referenciada várias vezes em uma tabela de fatos e cada referência desempenha um papel logicamente distinto (*playing-role dimension*). Por exemplo, uma tabela de fatos pode conter várias datas, cada uma delas representada por uma chave estrangeira para a dimensão de data. É essencial que cada chave estrangeira se refira a uma visão separada da dimensão de data para que as referências sejam independentes. Essas exibições de dimensão separadas, com nomes de colunas de atributos exclusivos, são chamadas de papéis (*roles*).

Cada fato representado na tabela de fatos corresponde à medida (*measure*) de um evento do processo de negócio que se deseja analisar. Essa medida quase sempre é numérica e, cada linha da tabela de fatos, representa um única medida do evento analisado. Dentro de uma tabela de fatos, apenas fatos consistentes com o grão declarado são permitidos. Por exemplo, em uma transação de vendas no varejo, a quantidade de um produto vendido e seu preço são valores que podem ser usados como medidas, enquanto o salário do gerente da loja não seria um valor adequado para representar este fato por ser um valor influenciado pela montante de venda não só de uma venda e sim de vendas acumuladas durante o mês.

Uma vez definido o modelo dimensional, devem ser estabelecidos os atributos das dimensões e das hierarquias baseados nos dados disponíveis de sistemas de banco de dados legados e de transações, formando o modelo lógico do DW. Posteriormente, os dados devem ser extraído de suas respectivas fontes e armazenados em uma área intermediária onde serão feitas as limpezas e transformações necessárias para que eles sejam carregados no DW.

Finalmente os dados do DW devem ser disponibilizados para serem consultados. Para tal, existem ferramentas de processamento analítico online (Online Analytical Processing – OLAP) que manipulam e analisam um grande volume de dados sob múltiplas perspectivas, oferecendo uma interface amigável para os usuários.

Diante das vantagens do uso sistema de apoio à decisão para análise do negócio, ele será usado para analisar Incidentes de Segurança da Informação.

## 2.4 Modelo dimensional bem fundamentado

A escolha de como representar a informação é extremamente importante para alcançar requisitos analíticos, fazendo da modelagem dimensional (MD) uma tarefa fundamental no ciclo de vida de soluções de um DW. Para isso, necessita-se de um processo de engenharia capaz de capturar a semântica das entidades do negócio e suas relações. Além disso, as necessidades do negócio e as possibilidades oferecidas pelos dados existentes devem ser avaliadas para definir a melhor forma de organizar as informações para o processamento analítico (47). Tendo em vista que as ontologias são aplicadas como um mecanismo para melhorar a expressividade semântica das representações de domínio, uma abordagem ontológica para a derivação de conceitos e esquemas MD a partir de categorias da UFO foi usada por Moreira et al.(47) para classificar os dados de origem do domínio durante a MD. E, Amaral e Guizzardi(48) usou padrões ontológicos, fundamentados na UFO, para melhorar a expressividade semântica dos modelos dimensionais.

Moreira et al.(47) utiliza uma análise ontológica usando UFO para estabelecer regras de mapeamento para derivar os conceitos MD a partir das categorias da UFO. Na modelagem dimensional, o fato corresponde a um evento físico observável (46) e, na UFO, os eventos complexos são manifestações de relacionamentos que envolvem vários participantes. Sendo assim, Moreira et al.(47) definiu uma regra de mapeamento que propõe a derivação de fatos a partir de eventos complexos e os objetos participantes do evento foram vistos como perspectiva de análise do fato, então foi estabelecida a regra de transformação desses objetos participantes em dimensões.

Em MD uma dimensão pode ser referenciada várias vezes pela mesma tabela de fatos, para cada um dessas referências a dimensão desempenha um papel diferenciado e diz-se que a dimensão desempenha papéis (*role-playing dimension*). A modelagem de dimensões que desempenham papéis pode se beneficiar do uso do padrão de papéis (*Role*) da UFO, pois ele pode ser usado para representar os diferentes papéis desempenhados por uma dimensão, ao mesmo tempo em que torna explícito que a mesma entidade desempenha diferentes papéis nesse contexto específico (48).

A dimensão de tempo pode ser derivada pelo intervalo de tempo dos eventos (47). Além disso, os eventos representam possíveis estados de coisas de uma realidade para outra, estes estados na UFO são representados como situações. Para analisar as mudanças de situação causadas por eventos, tais situações são definidas como dimensões no modelo dimensional (47).

Outro fato recorrente na MD, são as dimensões que representam entidades como um todo integral, composta por membros que desempenham o mesmo papel no coletivo. Em muitos casos, em modelos dimensionais, é importante distinguir a conceituação do todo da conceituação das partes, porque é necessário relacionar o todo com um fato que

se aplica ao coletivo e as partes a um fato aplicável apenas aos indivíduos. Para tal, os tipos da UFO usados para classificar indivíduos que compartilham propriedades comuns, tais como, categoria (*Category*) e mistura de papéis (*Role Mixin*), podem ser usados para explicitar a existência de um relacionamento uniforme que se mantém entre as partes que são representadas por tais dimensões (48).

Há dimensões que representam entidades que possuem diferentes níveis de classificação que geralmente a relação entre os diferentes níveis de classificação não é explícita nos modelos. Amaral e Guizzardi(48) propõem o uso da MLT para explicitar a relação entre entidades e seus tipos.

Visando oferecer suporte para criação de um sistema de apoio à decisão de qualidade será realizada uma modelagem dimensional bem fundamentada.

# 3 TRABALHOS RELACIONADOS

A busca por medidas de segurança que reduzam a quantidade de Incidentes de Segurança da Informação tem motivado diversos estudos. Muitas dessas estratégias de defesa cibernética usam ontologia com o objetivo explicitar melhor o domínio, como os trabalhos apresentados na Seção 3.1.

Além dos trabalhos que usam ontologias, outros apresentam as vantagens da criação de um ambiente para integração e análise das notificações de incidentes de forma a oferecer suporte à tomada de decisão (Seção 3.2). E, na Seção 3.3 serão citados trabalhos que geraram o modelo dimensional com base em ontologia com objetivo de prover um melhor suporte à tomada de decisão.

Sendo assim, este capítulo reúne as diversas abordagens serão usadas para desenvolver esse trabalho.

## 3.1 Ontologia de Incidente de Segurança da Informação

Devido à facilidade do uso de ontologias e devido à sua representatividade semântica, elas têm sido usadas como ferramentas para modelar tarefas relacionadas à segurança da informação. Algumas abordagens de representação do domínio focam na representação das elementos do domínio e seus tipos, como Moreira(10), apresentou um modelo de tratamento de incidentes, descrito como uma ontologia, com o objetivo de facilitar a integração com outros modelos e simplificar o processo de transferência de informações e conhecimento, dentro de um contexto de ciberdefesa colaborativa, conforme ilustrado na Figura 17. O modelo representa elementos, tais como, incidente (*Incident*) e ativo (*Asset*) sem correlacioná-las. E, mesmo representando os tipos de evento (*Event*) e de ativo (*Asset*) não fornece os critérios usados para estabelecer tal classificação.

Outra abordagem que também modela elementos e tipos foi proposta por Ping, Haifeng e Guoqing(11) ao desenvolver uma ontologia para um sistema de resposta a incidentes. A ontologia, apresentada na Figura 18, é usada para descrever as propriedades de cada novo caso de incidente para que ele seja submetido a um algoritmo que sugira a forma mais apropriada de resposta ao incidente. Para tal, a ontologia foca na definição de elementos, tais como, incidente (*Incident*) e vulnerabilidade, ou seja, a fraqueza identificada com a ocorrência do incidente (*leak arose incident*). E, o modelo oferece suporte à representação de tipos de ferramentas maliciosas (*intrusion tech*) usadas no ataque e o tipo de dano causado (*incident effect descript*) pelo incidente, porém não captura as características associadas a cada tipo e os critérios de classificação que auxiliam na distinção entre os tipos.

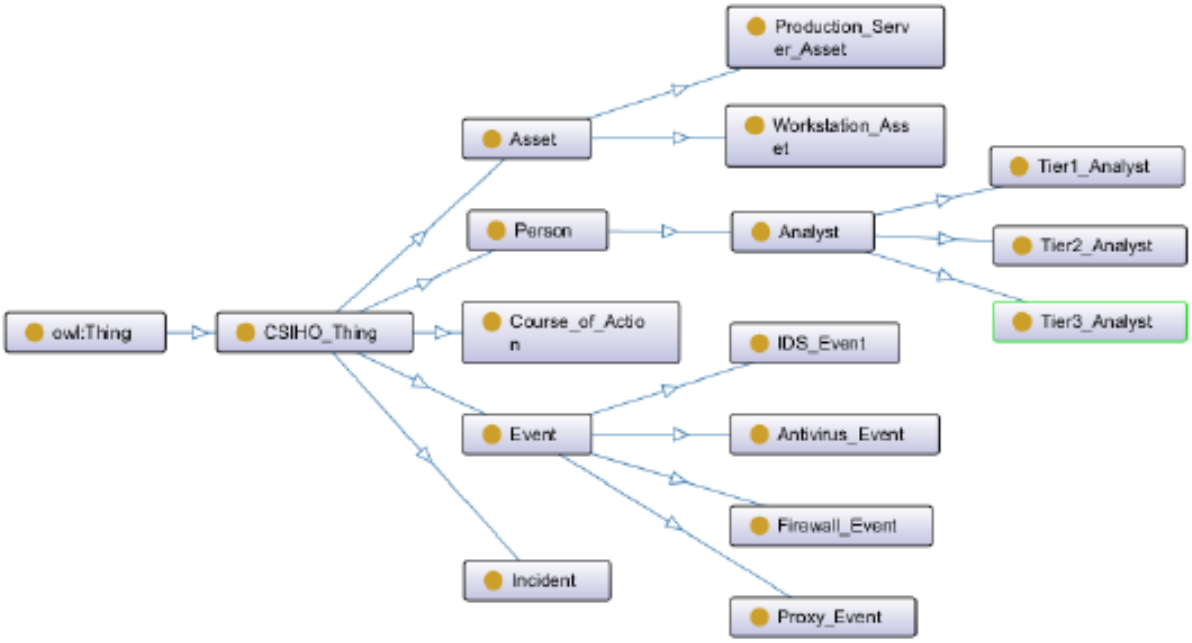



Figura 17 – Ontologia para tratamento de Incidente de Segurança da Informação. Fonte: (10)

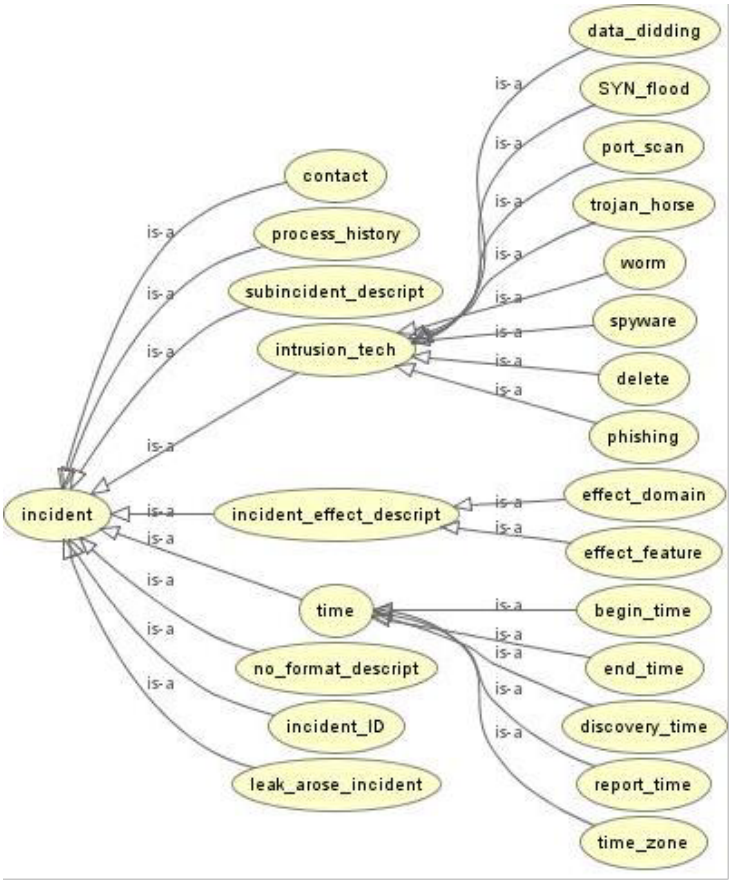



Figura 18 – Ontologia de Incidente de Segurança da Informação. Fonte: (11)

Há ontologias que modelam exclusivamente a hierarquia de classes. Li e Tian(12), para correlacionar alertas de sistemas de detecção de intrusão, criaram uma ontologia de hierarquia de classes de ataque, conforme ilustrado na Figura 19.

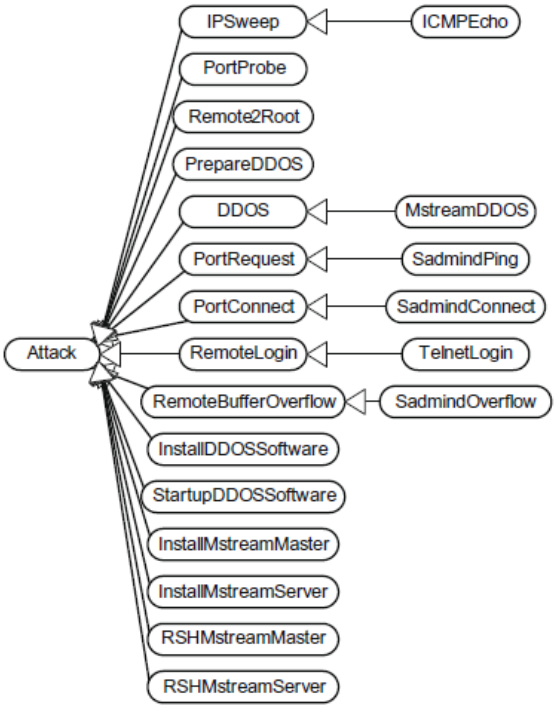



Figura 19 – Ontologia de hierarquia de classes de ataque. Fonte: (12)

Swimmer(13) projetou um modelo ontológico de classes de *malware* com o objetivo de facilitar a comunicação dos seus tipos. A ontologia consiste em duas partes, a primeira é uma hierarquia de características que são atribuídas aos *malware*. Com base nessas características, foi desenvolvida uma classificação hierárquica de *malware* com propriedades que fazem referência às características para viabilizar a diferenciação entre as classes. A Figura 20 ilustra o modelo construído na intenção de ser utilizado universalmente em alertas, sistemas de resposta de detecção de intrusão, entre outros.

Outro exemplo, modela a hierarquia de ataques de negação de serviço distribuído (*Distributed Denial of Service* - DDoS) (14) da base de dados de enumeração e classificação comum de padrão de ataque (*Common Attack Pattern Enumeration and Classification* – CAPEC). Este modelo, ilustrado na Figura 21, mostra a relação entre os tipos de ataque DDoS e, não retrata as características que distinguem os tipos de suas próprias especializações.

As ontologias demonstradas enfatizam a representação de alguns elementos do domínio, seus tipos ou classes hierárquica de tipos para prover uma solução para um ambiente específico, conforme apresentado no Quadro 2.

Em virtude dessas ontologias terem sido elaboradas para fins específicos, nenhuma

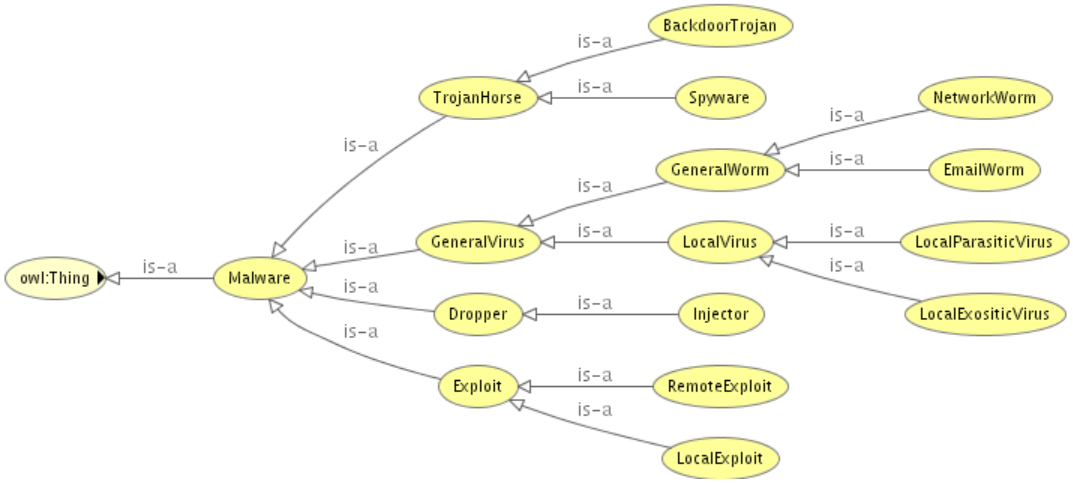



Figura 20 – Ontologia de classes hierárquicas de *malware*. Fonte: (13)

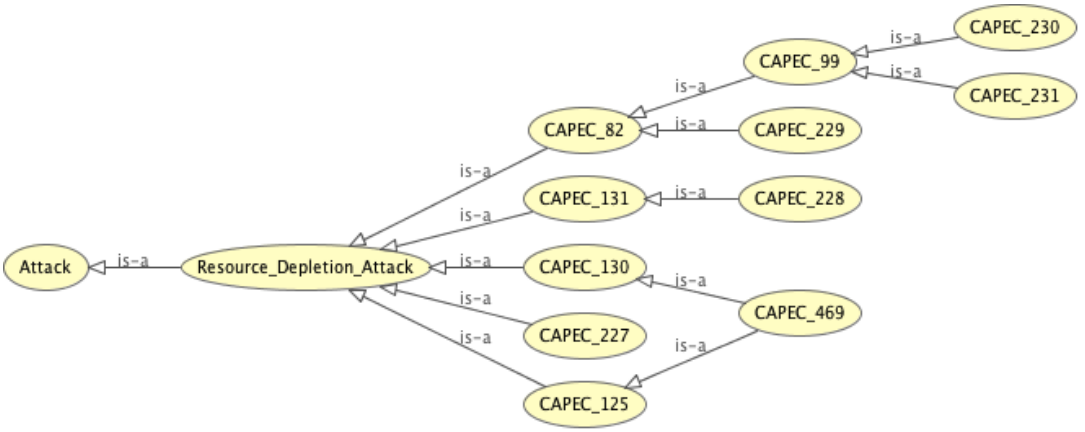



Figura 21 – Ontologia de hierarquia de tipos de ataque DDoS. Fonte: (14)

Quadro 2 – Comparação entre as ontologias de Incidente de Segurança da Informação

| Ontologia | Elemento | Tipo | Hierarquia de tipo |
|---|---|---|---|
| Moreira(10) | X | X | |
| Ansarinia et al.(14) | | X | X |
| Ping, Haifeng e Guoqing(11) | X | X | |
| Li e Tian(12) | | X | X |
| Swimmer(13) | | X | X |

delas reúne elementos, seus tipos e hierarquia de tipos com objetivo de oferecer uma abordagem aberta e genérica para a interoperabilidade. Para atingir esse objetivo, faz-se necessário que a ontologia represente o evento incidente e o contexto em que ele ocorre, os seus participantes e seus respectivos papéis, bem como seus impactos. Além disso, o próprio esquema de categorização é importante nesse domínio. Essa estrutura gera hierarquias de tipos nos quais os tipos mais específicos geralmente formam uma partição de um tipo mais geral, distinguindo instâncias de acordo com critérios de classificação.

## 3.2 Sistema de apoio à decisão de Incidentes de Segurança da Informação

Os sistemas de apoio à decisão permitem cruzamentos e manipulações dos dados com vistas a apoiar análises. Eles oferecem flexibilidade para visualizar dados sob diferentes perspectivas e têm sido utilizados em diversos domínios, inclusive no domínio de Incidente de Segurança da Informação. Pois, neste domínio, quanto mais rapidamente se consegue reconhecer, analisar e responder a um incidente, menores serão os custos de recuperação e os danos.

Um exemplo de uso de sistema de apoio à decisão de Incidentes de Segurança da Informação foi apresentado por Fuertes et al.(15). Os autores projetaram um sistema de apoio à decisão de incidentes para aumentar o potencial de defesa cibernética de um Grupo de Resposta a Incidentes de Segurança em Computadores (*Computer Security Incident Response Team* - CSIRT) acadêmico. O sistema usa um *data warehoure* (DW) para armazenar dados de *logs* obtidos de sensores de IDS *Snort* e *Passive Vulnerability Scanner* que possuem diferentes formatos. Para viabilizar esse armazenamento foi desenvolvido um modelo dimensional, apresentado na Figura 22, que permitiu armazenar os dados de forma homogênea, melhorar a visualização de eventos, como detecção de ataque e varredura de portas, e permitir às instituições membros do CSIRT tomarem ações e decisões. Os autores afirmam que houve um aumento do nível de segurança das instituições membros do CSIRT acadêmico que usam o sistema.

As vantagens obtidas por Fuertes et al.(15) ao desenvolver um sistema de apoio à decisão de Incidentes de Segurança para correlacionar dados de IDS de tipos específicos poderia ser ampliada para armazenar dados de outras fontes. Vislumbrando essa possibilidade, este trabalho irá propor um projeto de especificação de um ambiente de suporte à tomada de decisão que viabilize armazenar dados com formato distintos.

## 3.3 Modelagem dimensional baseada em Ontologia

A tarefa de desenvolvimento do DW considera os requisitos do usuário e as fontes de dados da organização. Os métodos de projeto mais atuais disponíveis exigem requisitos altamente expressivos como entrada para realizar a exploração e análise dos dados. No entanto, a tarefa de extrair os requisitos do usuário final pode ser complexa. Nesse sentido, algumas pesquisas propuseram o uso de ontologia para auxiliar na modelagem dimensional do DW.

Romero e Abelló(49) desenvolveram um método automático para identificar conceitos com probabilidade de desempenhar funções multidimensionais a partir de uma ontologia, denominado *Automating Multidimensional Design from Ontologies* (AMDO). O

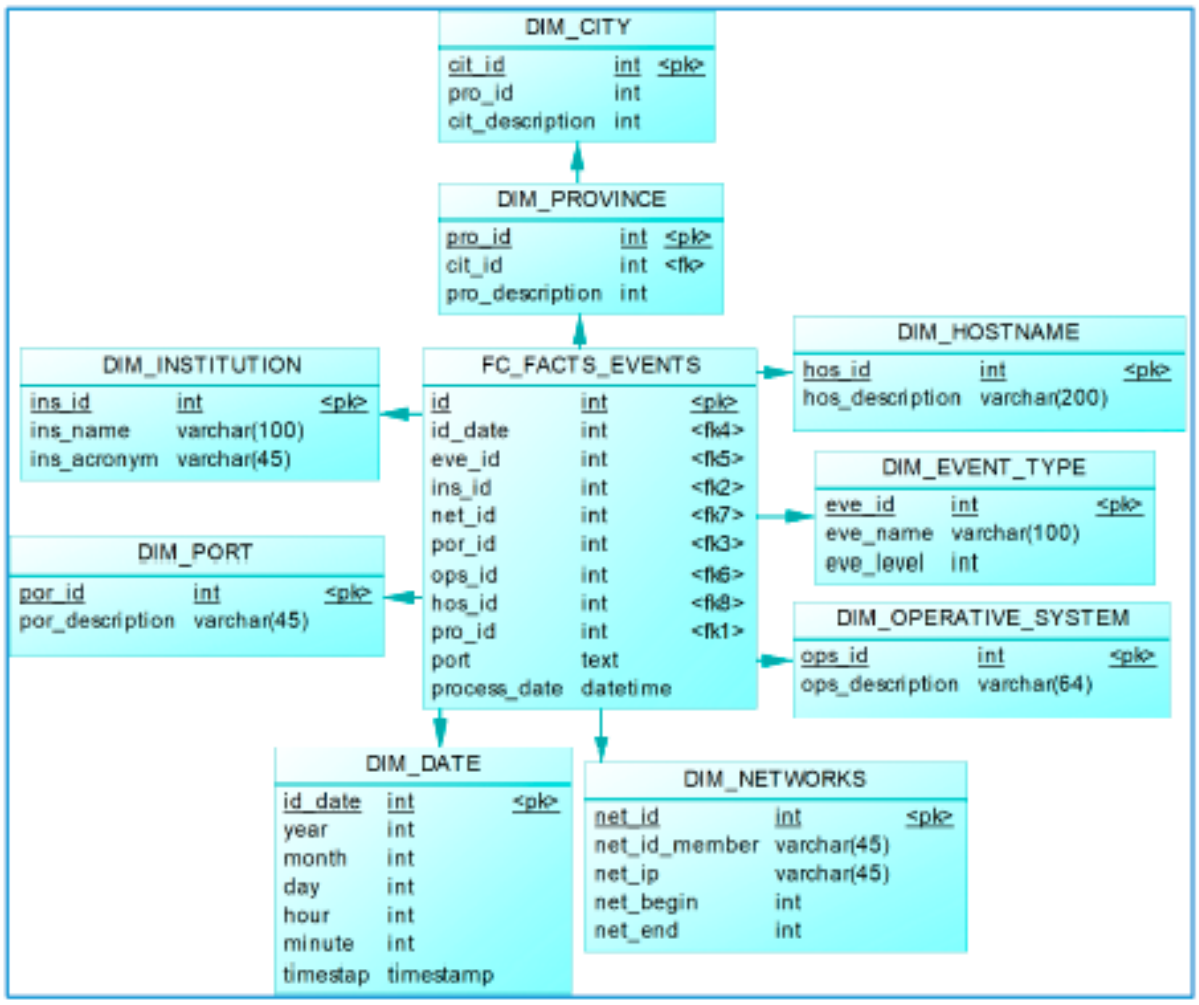



Figura 22 – Modelo dimensional de eventos de segurança. Fonte: (15)

principal objetivo do AMDO, ilustrado na Figura 23, é facilitar a elicitação de requisitos e as tarefas de desenvolvimento do DW. A entrada da abordagem é uma ontologia (*Ontology*) que contém informações sobre as fontes de dados e os vocabulários do domínio. O método estabelece que os conceitos com medidas associadas e que possuem possibilidades de serem analisados sob várias perspectivas provavelmente sejam considerados como possíveis fatos. O AMDO, ao percorrer a ontologia, identifica tais conceitos e apresenta uma lista de possíveis fatos com seus conceitos dimensionais (*Discover Dimensions*) e medidas associados (*Discover Measures*). Essa lista deve ser apreciada pelo usuário para a escolha dos fatos (*Find out Facts*). Para cada fato escolhido, o AMDO identifica caminhos de agregação relevantes para relacionamentos típicos parte-todo, organizando os conceitos dimensionais em hierarquias de dimensões (*Point out Dimension Hierarchies*). Assim, a saída do AMDO será um esquema dimensional para cada fato identificado (*Multidimensional Schemas*). Os autores afirmam que esse método captura os fatos de forma mais expressiva, provendo modelos mais fáceis de entender e usar.

Thenmozhi e Vivekanandan(50) estavam em busca de um forma de liberar os

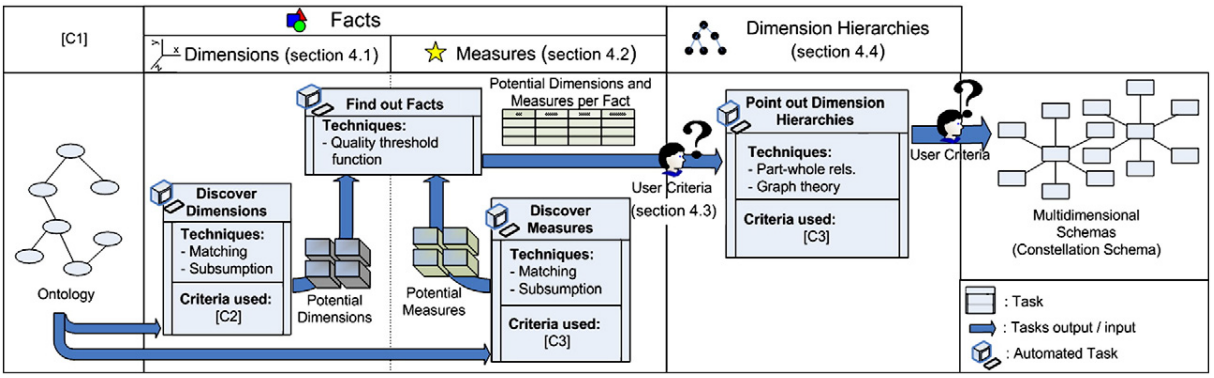



Figura 23 – Método AMDO

especialistas de realizar a modelagem dimensional do DW. Então, eles desenvolveram uma ferramenta que usa ontologia para derivar automaticamente o modelo dimensional para o DW (*Ontology based data warehouse schema design* - OBDWSD), ilustrada na Figura 24. Os objetivos, o contexto e as medidas para o DW levantados na análise de requisitos são fornecidos através de um GUI e transformados na ontologia de requisitos (*DWRequeriment Ontology*). Para captar o significado semântico dos dados, cada fonte de dados é transformada em uma ontologia que são associadas através de uma ontologia de mesclagem (*Ontology Merging*), formando uma ontologia global (*Global Ontology*). Após, faz-se uma correspondência entre a ontologia de requisitios e a ontologia global, resultando em uma nova ontologia denominada *Ontology Matching*. Um algoritmo (*Schema Transformation*) percorre os conceitos dessa ontologia em busca dos conceitos que tem um número de propriedades numéricas acima de um limite estipulado previamente na ferramenta. Esses conceitos são identificados como fato, suas propriedades numéricas são identificadas como medidas e suas propriedades não numéricas, se tiverem um relacionamento muitos para um com o conceito classificado como fato, são identificadas como dimensões, formando o modelo dimensional (*Local Schema*) com significado semântico.

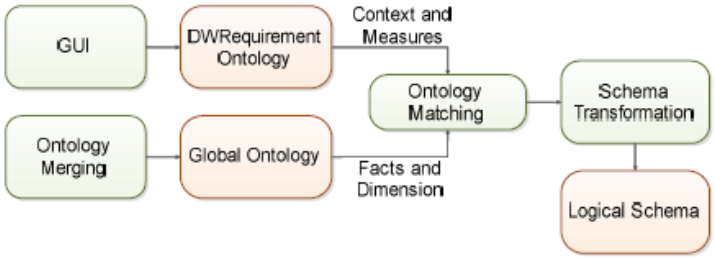



Figura 24 – Ferramenta OBDWSD

Bouchra et al.(16) fornecem um método para o desenvolvimento do DW através da combinação das necessidades dos usuários e dos dados do negócio. O método ocorre em duas etapas, conforme ilustrado na Figura 25. A primeira, interativa e semi-automática, consiste em identificar os futuros usuários do DW, as fontes de dados (*Data Source*) que

eles usam e levantar suas necessidades (*Decisional Needs*). Com base nesse levantamento, consultas as fontes de dados são realizadas e transformadas em visões materializadas (*Materialized Views*). Em virtude das visões materializadas terem sido derivadas de fontes dados distintas, elas são potencialmente heterogêneas. No entanto, para que todas as informações sejam reunidas em um mesmo DW, esses conflitos semânticos devem ser resolvidos. Para tal, o método usa ontologia (*Ontology Transformation*) como meio de representação e integração semântica das visões materializadas. A transformação de cada uma das visões materializadas em uma ontologia é feita por um algoritmo. Posteriormente, um outro algoritmo de alinhamento de ontologia (*Alignment*) integra todos os conceitos de todas as ontologias locais gerando uma ontologia global (*Global Ontology*). Cada conceito da ontologia global será considerado um potencial fato se estiver relacionado outros conceitos por um relacionamento muitos para um. As propriedades numéricas do fato são identificadas como medidas. Cada conceito da ontologia relacionado ao conceito que foi transformado em fato através de um relacionamento um para muitos é considerado como uma dimensão. E, conceitos da ontologia com relacionamentos muitos para um com os conceitos definidos como dimensões são considerados como nível. Como resultado, uma ontologia decisional (*Decisional Ontology*) é gerada.

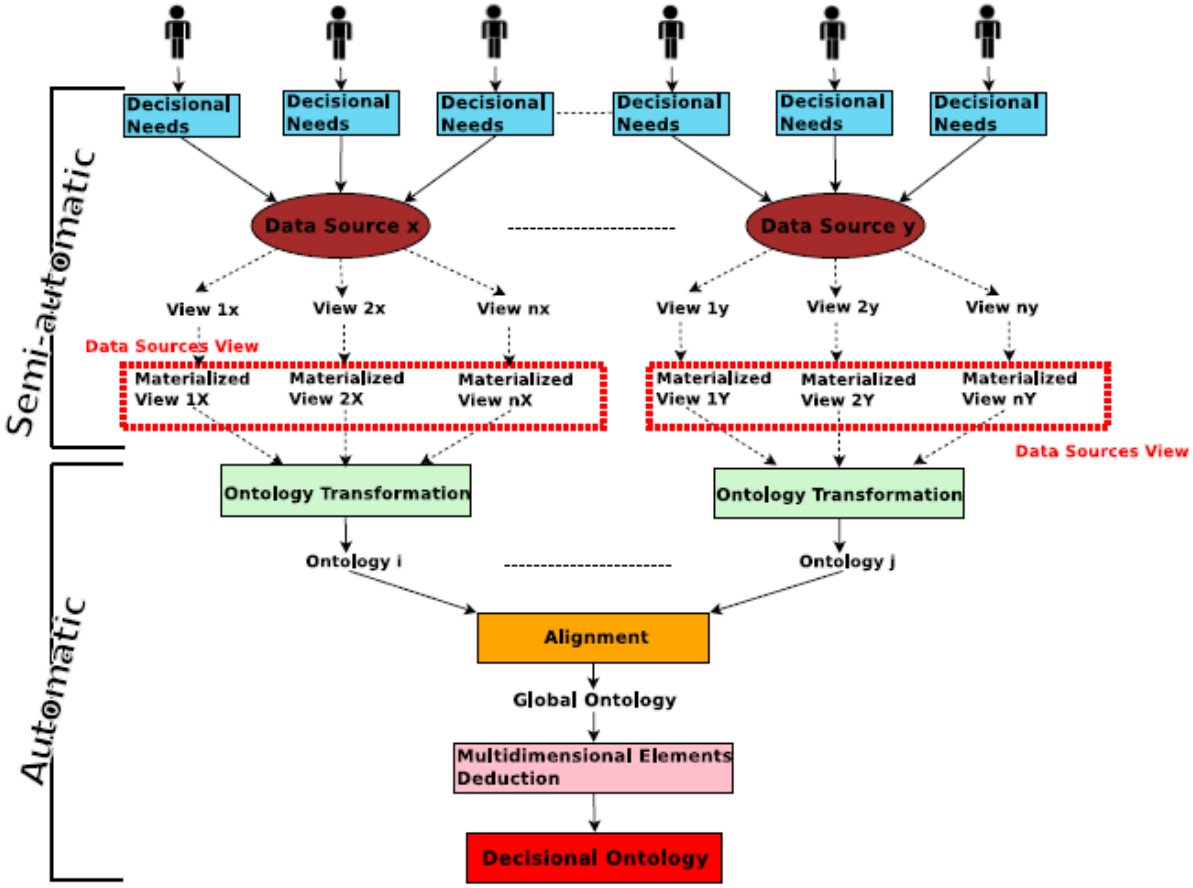



Figura 25 – Método modelagem dimensional baseado em ontologia de (16)

Ren, Wang e Lu(17) estabeleceram um método de modelagem dimensional de DW baseado em ontologia com intuito de eliminar a heterogeneidade semântica de fontes de dados com informações médicas. O método, ilustrado na Figura 26, constrói a ontologia de domínio (*domain ontology*) baseada em metadados (*metadata*) de fonte de dados heterogêneas (*data source*) para fornecer o significado semântico dos metadados e resolver o problema da heterogeneidade semântica entre as fontes de dados. Posteriormente, um algoritmo percorre a ontologia de domínio para identificar os conceitos que são potenciais fatos, dimensões e medidas (*potencial facts/ dimensions /measures*). Esses fatos, dimensões e medidas são considerados a estrutura inicial do modelo dimensional. Paralelamente, uma análise de requisitos (*business requeriments*) orientada nos objetivos dos usuários (*user*) é realizada para obter um modelo decisional (*decisional model*) dos requisitos médicos do negócio. Os conceitos do modelo decisional são comparados com os conceitos dos fatos, dimensões e medidas potenciais para gerar a estrutura final do modelo dimensional. Experimentos mostraram que o uso desse método melhorou consideravelmente a eficiência da modelagem dimensional do DW médico.

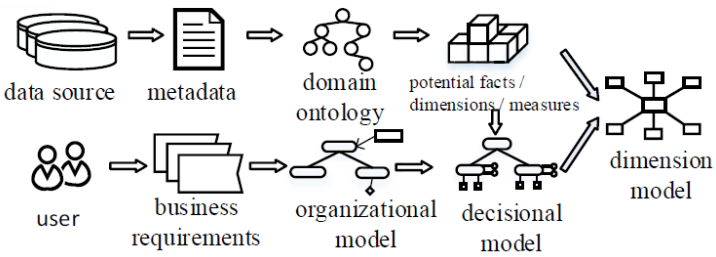



Figura 26 – Método de modelagem dimensional baseado em ontologia de Ren, Wang e Lu(17)

Esses trabalhos mostraram como tirar proveito da ontologia para desenvolver um modelo conceitual. Eles utilizam a ontologia para otimizar a análise de requisitos, resolver o problema da heterogeneidade semântica nos requisitos de negócios e os conflitos de representação inerentes à consolidação de fontes de dados distintas. E, com base na ontologia, conseguiram desenvolver métodos para identificar fatos, dimensões e medidas, formando modelos dimensionais de DW expressos em um formato semântico padronizado e consistente. Visando obter esses benefícios, este trabalho desenvolve um método para construir um sistema de apoio à decisão para analisar Incidentes de Segurança da Informação através da elaboração de modelo dimensional baseado em ontologia bem fundamentada na UFO-MLT.

# 4 METODOLOGIA DEFESA

Há poucas garantias de que uma organização seja imune a ocorrências de Incidentes de Segurança da Informação. A defesa cibernética possui um caráter mais dinâmico do que as técnicas de segurança da informação, sendo elas caracterizadas por serem estáticas e preventivas, embora importantes, são insuficientes para prover a defesa cibernética. A preparação da defesa será possibilitada pela existência de uma gestão que evidencie os elementos de um sistema complexo (51).

Para que incidentes sejam evitados ou, ao menos, minimizados a capacidade de aprender e de absorver erros e acertos deve ser explorada, através de uma análise acerca dos incidentes ocorridos e o que possibilitou o sucesso deles (25). Porém, o domínio carece de uma representação aberta e genérica que seja capaz de reduzir a heterogeneidade semântica.

Em outros domínios que enfrentavam desafios semelhantes, conforme relacionado na Seção 3.2, as ontologias foram usadas para remover ambiguidades conceituais e facilitar a integração de dados de diversas fontes para serem analisados em um sistema de apoio à decisão. Então, para análise de ocorrências de Incidentes de Segurança de Informação foi desenvolvida uma metodologia para **D**efinir, **E**specificar e **F**ormalizar os conceitos do domínio com **Ê**nfase em oferecer **S**uporte à tomada de decisão e **A**umentar a expressividade semântica (**DEFESA**).

A metodologia DEFESA tem o propósito de ser abrangente com relação à descrição dos conceitos e específica em oferecer suporte à tomada de decisão formando uma arquitetura em camadas, baseada na estrutura descrita na Seção 2.1.1. Na camada fundacional fica a UFO-MLT para que os seus conceitos, relações e categorias embasem a ontologia do domínio. Essa ontologia fica no centro da arquitetura e alicerça o modelo dimensional com expressividade semântica para atender à demanda de construção do sistema de apoio à decisão, como mostra a Figura 27.

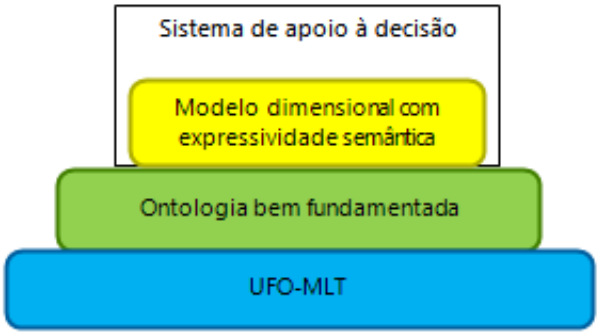



Figura 27 – Metodologia DEFESA - Camadas arquiteturais

Esse arcabouço ontológico integrado da metodologia DEFESA compõe-se dos macroprocessos de *Modelar ontologia de domínio bem fundamentada* e de *Construir sistema de apoio à decisão*. Este último é composto pelo processo de *Elaborar modelo dimensional com expressividade semântica* e *Desenvolver ambiente analítico de dados*, conforme ilustrado na Figura 28.

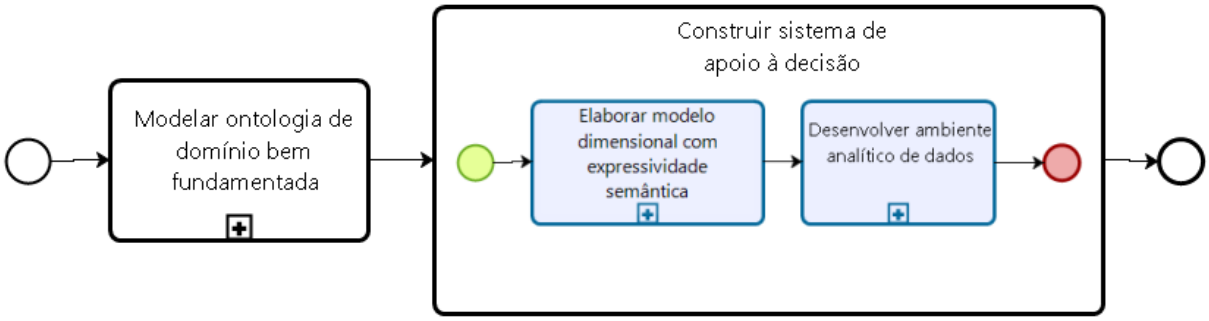



Figura 28 – Metodologia DEFESA - Macroprocessos

Esses macroprocessos contêm um conjunto de atividades definidos com base nas metodologias para construção de ontologias Methontology e SABiO descritas na Seção 2.2. O Quadro 3 mostra a correspondência entre as atividades da metodologia DEFESA e as atividades das metodologias Methontology e SABiO. Vale destacar que o macroprocesso *Construir sistema de apoio à decisão* tem o diferencial de detalhar atividades específicas para a construção de um sistema de apoio à decisão. Os próximos parágrafos detalham cada uma das atividades da metodologia DEFESA.

Quadro 3 – Correspondência entre as atividades da metodologia DEFESA e as atividades das metodologias Methontology e SABiO

| DEFESA | Methontology | SABiO |
|---|---|---|
| **Modelar ontologia de domínio bem fundamentada** | | |
| Realizar o levantamento dos termos | Aquisição | Aquisição do conhecimento |
| Definir formalmente os conceitos do domínio | Conceitualização | Captura e formalização da ontologia |
| Avaliar os termos levantados<br>Avaliar os conceitos definidos | Avaliação | Avaliação |
| Objetos de dados de saída | Documentação | Documentação |
| **Construir sistema de apoio à decisão** | | |
| Elaborar modelo dimensional com expressividade semântica | Formalização | Projeto |
| Desenvolver ambiente analítico de dados | Implementação | Implementação |

O macroprocesso de *Modelar ontologia de domínio bem fundamentada* inicia com a atividade de *Especificar o propósito da ontologia*. Ela norteia a atividade de *Realizar o levantamento dos termos* do domínio e a atividade de *Definir formalmente os conceitos do domínio*. Pois, no final de cada uma dessas atividades é realizada uma avaliação para verificar se os objetos de dados produzidos atendem ao propósito da ontologia. A Figura 29 ilustra essas atividades.

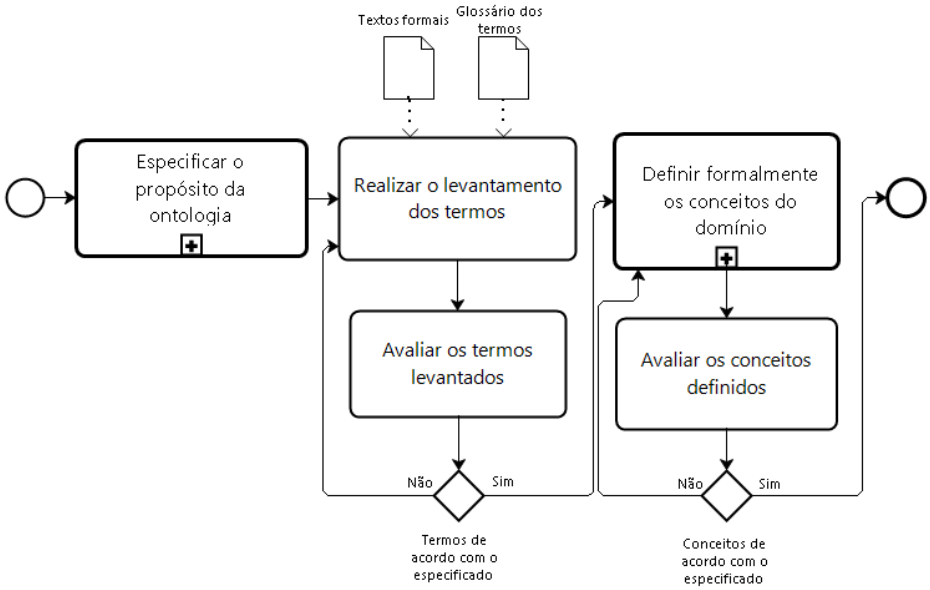



Figura 29 – Metodologia DEFESA - Atividades do processo *Modelar ontologia de domínio bem fundamentada*

A atividade de *Especificar o propósito da ontologia* constitui-se de um processo que detalha o objetivo da construção da ontologia e seu escopo para que as características essenciais da ontologia e a granularidade fiquem evidenciadas. Além disso, documenta o uso pretendido com a abordagem e para qual tipo de usuários finais. Todas essas informações são registradas em um documento de especificação escrito em linguagem natural e, com base nesses requisitos são definidas questões de competência. A Figura 30 ilustra esta atividade que contém as seguintes tarefas e os objetos de dados de saída:

Tarefas:

- Definir o objetivo da ontologia;
- Definir o escopo da ontologia;
- Definir o cenário de uso da ontologia;
- Definir os potenciais usuários da ontologia.

Objetos de dados de saída:

- Documento de especificação;
- Questões de competência da ontologia.

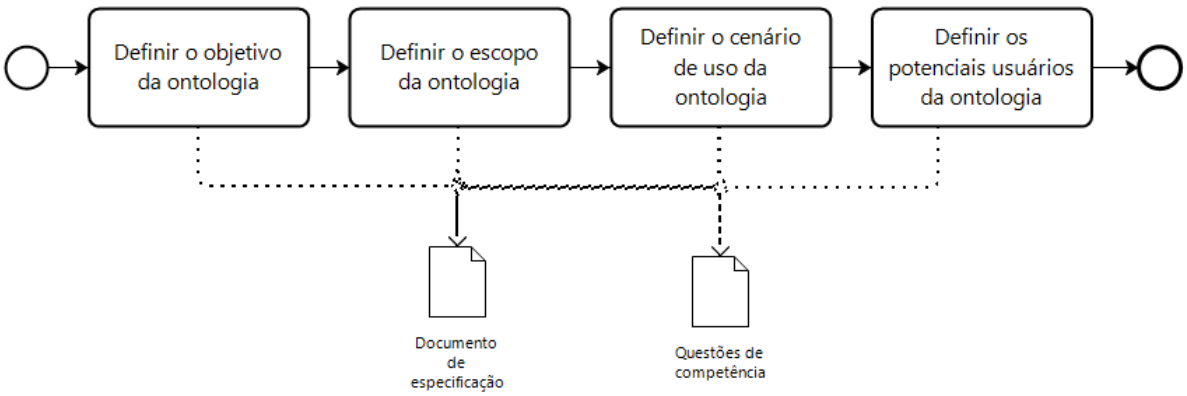



Figura 30 – Metodologia DEFESA - Atividades do processo *Especificar o propósito da ontologia*

Após especificar o propósito da ontologia, inicia-se a tarefa de *Realizar o levantamento dos termos* para identificar o vocabulário do domínio. Esses termos podem ser extraídos de textos formais sobre o assunto, como normas, padrões e artigos científicos e reunidos em um glossário de termos, conforme ilustrado na Figura 28. No final dessa tarefa, há uma avaliação para verificar se os termos contidos no glossário são necessários e suficientes para atender ao proposto na especificação. Essa tarefa tem os seguintes objetos de dados de entrada e saída:

Objeto de dados de entrada:

- Textos formais.

Objeto de dados de saída:

- Glossário de termos.

Em virtude dessas informações terem sido coletadas de diversas fontes, possivelmente, o glossário de termos conterá o mesmo termo com diferentes significados e vice-versa, e alguns termos com falta de expressividade semântica. Esses conflitos de representação podem ser reduzidos através da atividade de *Definir formalmente os conceitos do domínio*.

Nesta atividade, ilustrada na Figura 31, os termos são analisados ontologicamente para elucidar as entidades do domínio, descobrir distinções relevantes entre elas e suas relações, com o propósito prático de desambiguar termos com diferentes interpretações em diferentes contextos. As entidades serão conceitualizadas e relacionadas utilizando o

sistema de categorias da UFO. Além disso, as entidades são classificadas em níveis de acordo com a MLT e representadas através de uma ontologia baseada na UFO-MLT. As tarefas, os objetos de dados de entrada e de saída dessa atividade são:

Tarefas:

- Analisar ontologicamente os termos usando a UFO;
- Classificar as entidades em níveis de acordo com a MLT;
- Construir a ontologia baseada na UFO-MLT.

Objetos de dados de entrada:

- Sistema de categorias da UFO;
- Padrões de classificação em múltiplos níveis da MLT.

Objetos de dados de saída:

- Ontologia baseada na UFO-MLT.

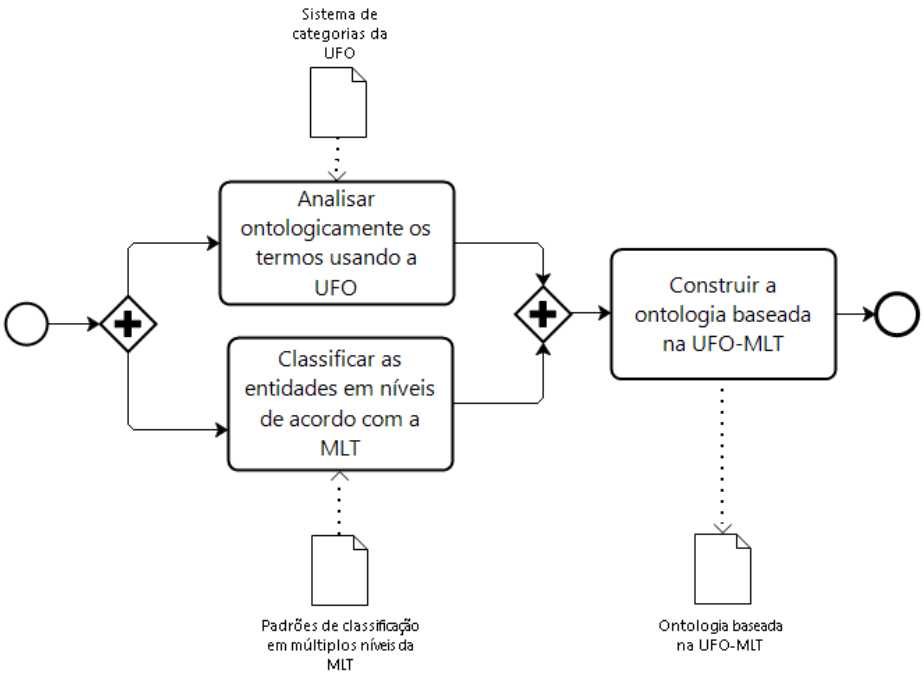



Figura 31 – Metodologia DEFESA - Atividades do processo *Definir formalmente os conceitos do domínio*

Ao concluir a atividade de *Definir formalmente os conceitos do domínio*, uma avaliação é realizada para verificar as questões de competência tem possibilidade de serem respondidas usando a ontologia baseada na UFO-MLT.

As atividades do macroprocesso *Modelar ontologia de domínio bem fundamentada*, reunidas na Figura 32, enfatizam o aumento a expressividade semântica do domínio. O suporte à tomada de decisão será oferecido pelo macroprocesso de *Construir sistema de apoio à decisão*. A construção do sistema de apoio a decisão envolve as atividades de *Elaborar modelo dimensional com expressividade semântica* e *Desenvolver ambiente analítico de dados*.

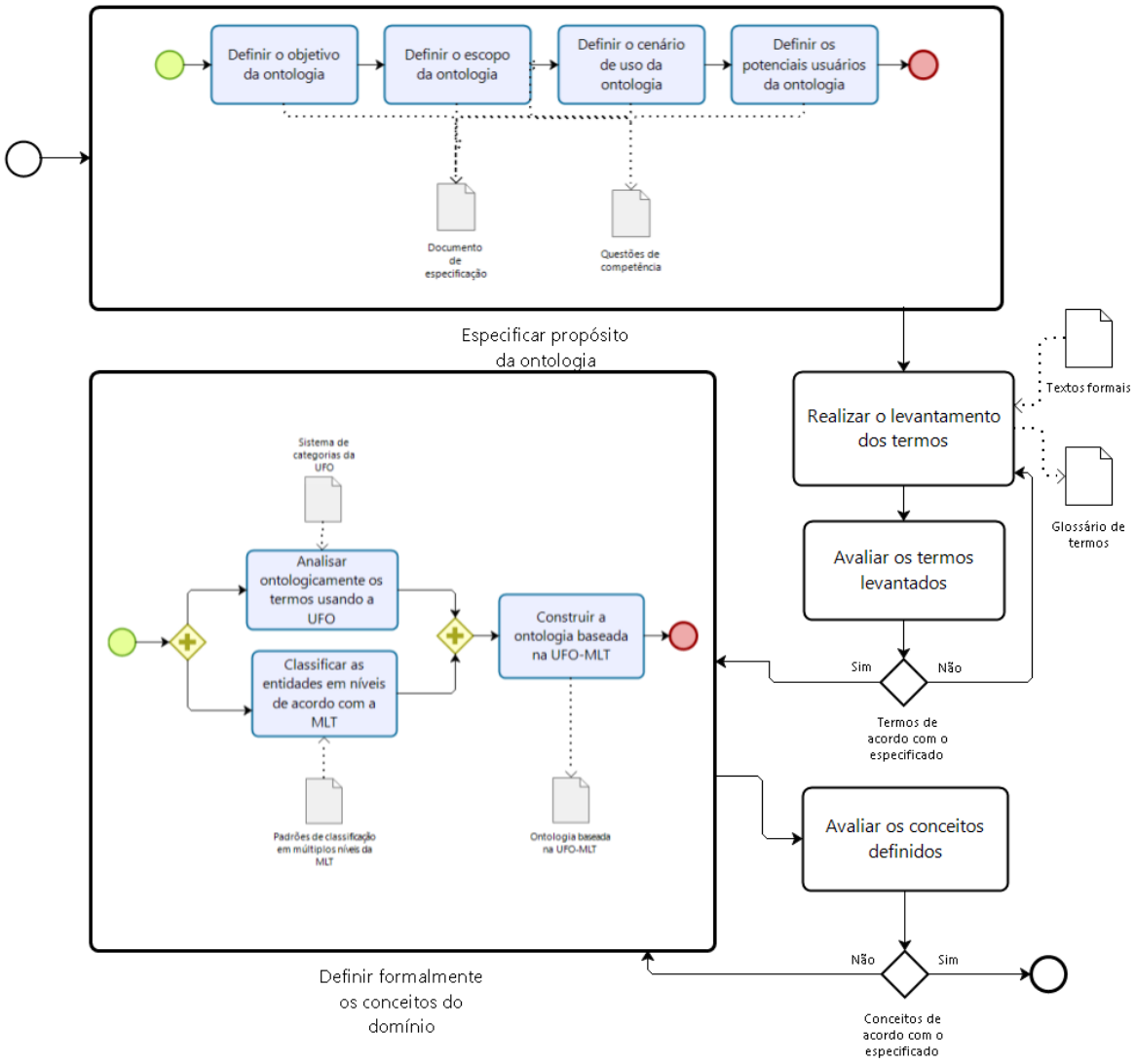



Figura 32 – Metodologia DEFESA - Macroprocesso de *Modelar ontologia de domínio bem fundamentada*

O processo de *Elaborar modelo dimensional com expressividade semântica* segue as etapas de selecionar o processo do negócio, definir a granularidade, identificar as dimensões e identificar os fatos, conforme detalhado na Seção 2.3. O processo do negócio e a granularidade foram previamente registrados no documento de especificação e expressos na ontologia baseada na UFO-MLT. Então, esse modelo conceitual bem fundamentado será usado como base para definir o propósito da análise expresso através de questões

analíticas, conforme ilustrado na Figura 33. Essa atividade consiste de uma tarefa com os seguintes objetos de dados de entrada e saída:

Objeto de dados de entrada:

- Ontologia baseada na UFO-MLT

Objeto de dados de saída:

- Questões analíticas.

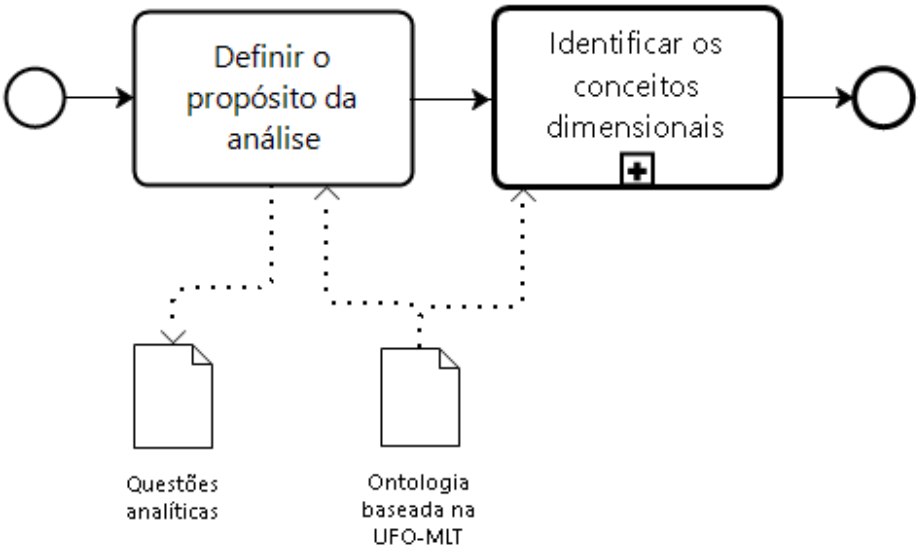



Figura 33 – Metodologia DEFESA - Atividades do processo *Elaborar modelo dimensional com expressividade semântica*

A próxima atividade consiste em *Identificar os conceitos dimensionais*. As ontologias por representarem os conceitos do domínio com maior expressividade semântica são boas fontes de derivação de conceitos dimensionais, conforme detalhado na Seção 2.4. Sendo assim, para identificar os conceitos dimensionais serão aplicadas regras para transformar cada elemento da ontologia, categorizado usando a UFO-MLT, em fato, dimensão ou hierarquia. Conforme ilustrado na Figura 34, esta atividade tem as seguintes tarefas e objetos de dados de entrada e saída:

Tarefas:

- Aplicar regras de transformação para identificar fatos;
- Aplicar regras de transformação para identificar dimensões;
- Aplicar regras de transformação para identificar hierarquias.

Objeto de dados de entrada:

- Ontologia baseada na UFO-MLT.

Objeto de dados de saída:

- Modelo dimensional.

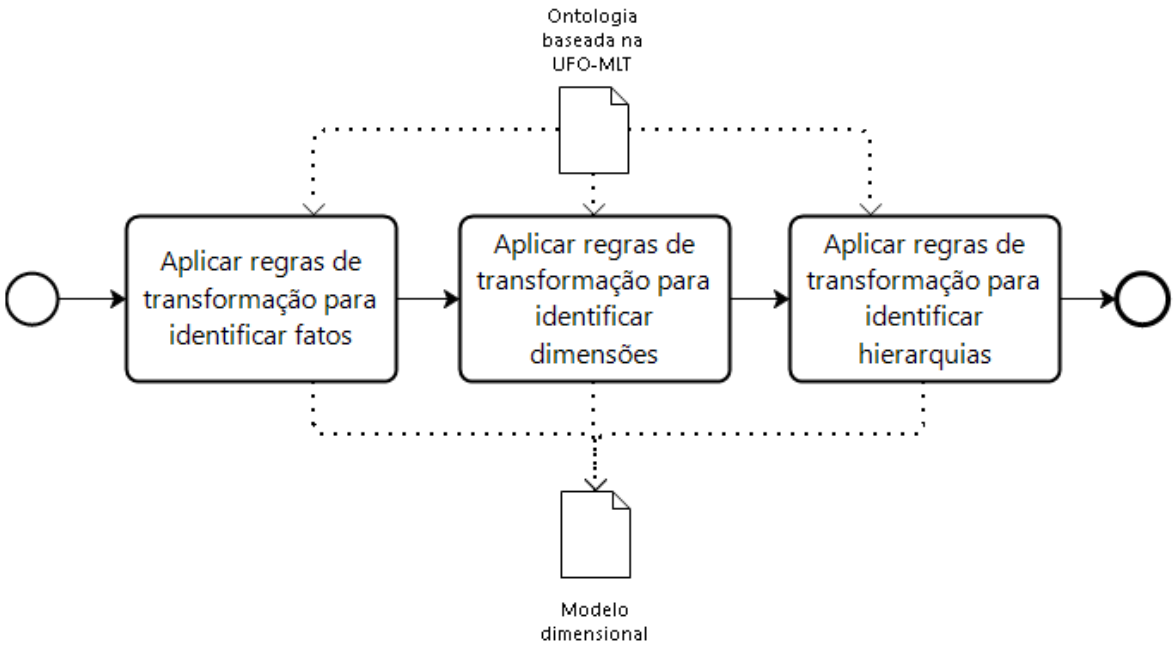



Figura 34 – Metodologia DEFESA - Processo *Identificar conceitos dimensionais*

As atividades até então elaboradas levavam em consideração as necessidades do negócio. Porém, as análises são feitas através de consultas aos dados. As próximas atividades do processo de desenvolvimento do ambiente analítico de dados consideram as fontes de dados disponíveis, conforme detalhado na Figura 35.

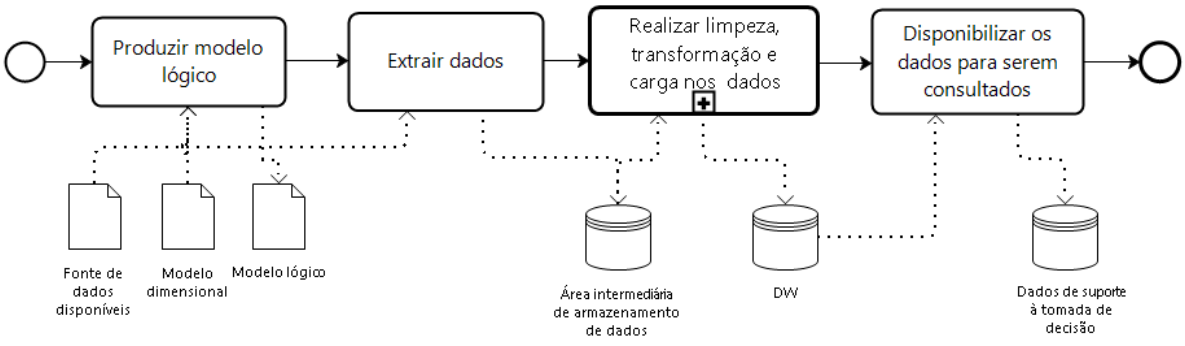



Figura 35 – Metodologia DEFESA - Processo *Desenvolver ambiente analítico de dados*

A atividade de *Produzir modelo lógico* considera os dados disponíveis do domínio para definir os atributos das dimensões, medidas e hierarquias resultando em um modelo lógico. Esta tarefa tem como objetos de dados de entrada e saída:

Objeto de dados de entrada:

- Fonte de dados disponíveis.

Objeto de dados de saída:

- Modelo lógico.

Antes dos dados serem carregados no *data warehouse* (DW), eles precisam ser transformados para se tornarem valores dos atributos definidos no modelo lógico. Para tal, os dados serão extraídos de suas fontes e armazenados em uma área de armazenamento intermediária para serem tratados. A Figura 35 ilustra essa atividade que tem os seguintes os objetos de dados de entrada e de saída:

Objeto de dados de entrada:

- Fonte de dados disponíveis.

Objeto de dados de saída:

- Dados armazenados na área intermediária.

Na área de armazenamento intermediária, os dados são limpos e transformados para serem carregados no DW, conforme detalhado na Figura 35. Este processo contém as seguintes tarefas e objeto de dados de saída:

Tarefas:

- Limpar dados;
- Transformar dados;
- Armazenar dados no DW.

Objeto de dados de saída:

- Data warehouse.

Os dados do DW já estão no formato adequado para realizar consultas analíticas. Existem ferramentas de processamento analítico on-line (*Online Analytical Processing* –

*OLAP*) para auxiliar na manipulação e consultas do grande volume de dados do DW. Sendo assim, a última tarefa do processo de *Construir o sistema de apoio à decisão* é *Disponibilizar os dados para serem consultados* em uma ferramenta OLAP. Os objetos de entrada e saída dessa tarefa são:

Objeto de dados de entrada:

- DW;

Objeto de dados de saída:

- Dados de suporte à tomada de decisão.

A Figura 36 reúne todas as atividades do macroprocesso *Construir sistema de apoio à decisão* e a Figura 37 mostra todas as atividades da metodologia DEFESA.

A metodologia DEFESA foi estabelecida inspirada nas metodologias para construção de ontologias apresentadas na Seção 2.2. E, tem o diferencial de detalhar uma forma de implementação, através do processo de desenvolver sistema de apoio à decisão. Tais características possibilitam que DEFESA tenha a flexibilidade de ser aplicada em diversos domínios apesar dela ter sido desenvolvida especialmente para a análise de Incidentes de Segurança da Informação.

Os próximos capítulos são dedicados à utilização da metodologia DEFESA no domínio de Incidente de Segurança de Informação. No capítulo 5, o macroprocesso *Modelar ontologia de domínio bem fundamentada* foi usado para construir sCuDO, a ontologia de domínio de Incidente de Segurança da Informação, que foi utilizada para representar ocorrências de incidentes no capítulo 6. No capítulo 7, o processo *Elaborar modelo dimensional com expressividade semântica* foi usado para produzir sCuD$^2$O, o modelo dimensional de Incidentes de Segurança da Informação. Um ambiente analítico de dados foi desenvolvido no Capítulo 8.

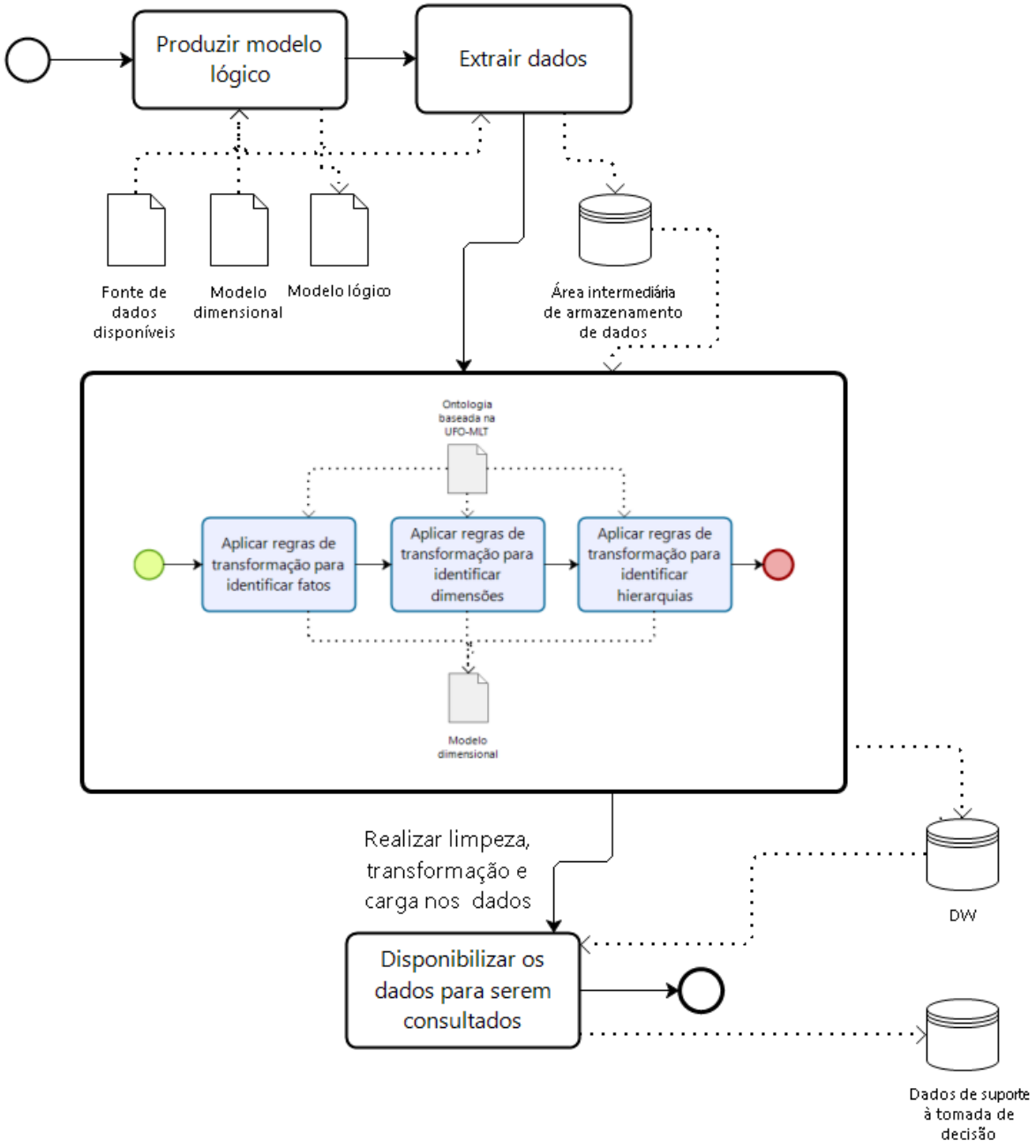



Figura 36 – Metodologia DEFESA - Macroprocesso *Construir sistema de apoio à decisão*

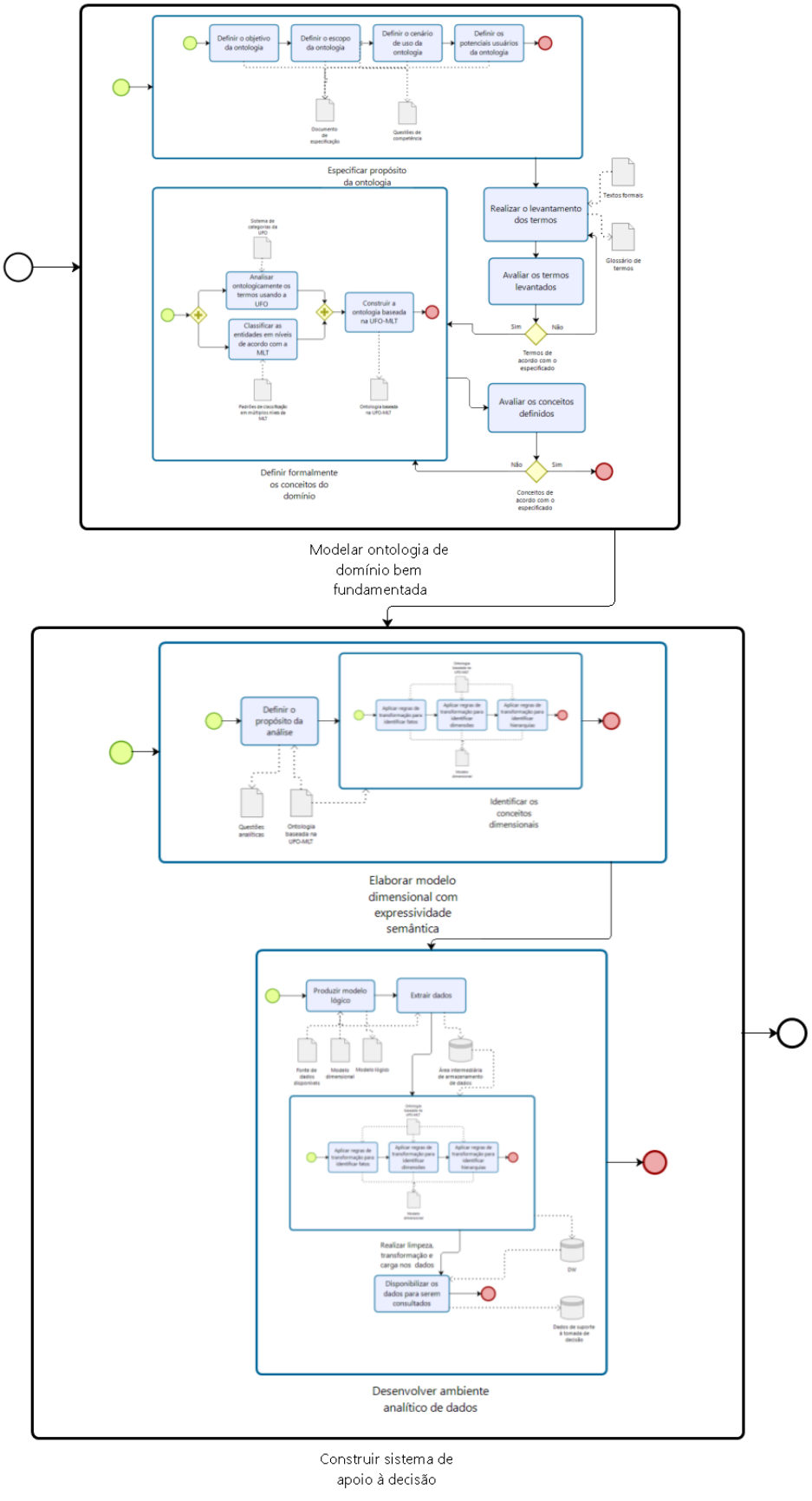



Figura 37 – Todas as atividades da Metodologia DEFESA

# 5 APLICAÇÃO DA METODOLOGIA DEFESA PARA CONSTRUIR SCUDO

Este capítulo apresenta sCuDO, a ontologia de domínio de Incidente de Segurança da Informação. Para o desenvolvimento dessa ontologia, foram utilizadas as atividades do processo de *Modelar ontologia de domínio bem fundamentada* da metodologia DEFESA, conforme ilustrado na Figura 32.

## 5.1 Especificar o propósito da ontologia

Seguindo a metodologia DEFESA, o primeiro processo para modelar ontologia de domínio bem fundamentada consiste em *Especificar o propósito da ontologia*, conforme representado na Figura 30, através das seguintes tarefas:

**Objetivo**: representar o domínio de Incidente de Segurança da Informação como um evento, explicitando suas causas, seus participantes e os danos causados através de entidades tipificadas, correlacionadas e categorizadas com critérios de classificação definidos.

**Escopo**: estabelecer uma conceitualização comum sobre incidente que possa ser utilizada em consenso.

**Cenário de uso**: modelo de referência para representar ocorrências de incidentes apoiando o compartilhamento e integração dessas informações.

**Potenciais usuários:** integrantes de Grupos de Resposta a Incidentes de Segurança em Computadores (Computer Security Incident Response Team - CSIRT) e profissionais da área de segurança.

Baseado nesse propósito sCuDO deverá ser capaz de responder as seguintes questões de competência:

**Questões de competência:**

QC1- Quais são as características de um evento para classificá-lo como incidente?

QC2- Qual foi o dano causado pelo incidente?

QC3- Quem causou o incidente?

QC4- Quem sofreu o incidente?

QC5- Quando ocorreu o incidente?

QC6- Qual foi a causa do incidente?

QC7- Por que o incidente ocorreu?

QC8- Como o incidente ocorreu?

QC9- Como o incidente foi classificado?

## 5.2 Realizar o levantamento dos termos

Baseado no propósito especificado, foi realizado um levantamento dos termos do domínio, conforme sugerido na Figura 29, com os principais órgãos nacionais relacionados a segurança de informação, visando reunir o conhecimento fragmentado sobre os elementos envolvidos no Incidente de Segurança da Informação. Complementarmente, foram consultadas normas e trabalhos científicos relacionados a segurança da informação. Tais termos com suas respectivas fontes estão descritos abaixo.

Segundo (52), um Incidente de Segurança da Informação é identificado com um simples ou uma série de eventos de segurança da informação indesejados ou inesperados, que tenham uma grande probabilidade de comprometer as operações do negócio e ameaçar a segurança da informação. Esses eventos de segurança são identificados através de um sistema, serviço ou estado de rede indicando uma possível violação da política de segurança da informação, falha de controles ou uma situação anteriormente desconhecida que pode ser relevante para a segurança. A segurança da informação se refere às ações que objetivam viabilizar e assegurar a disponibilidade, a integridade, a confidencialidade e a autenticidade de dados e informações (53); adicionalmente, outras propriedades, tais como autenticidade, responsabilidade, não repúdio e confiabilidade, também podem estar envolvidas (52).

Com o aumento da interconectividade, a informação está agora exposta a um crescente número e a uma grande variedade de ameaças e vulnerabilidades (54). A lista de possíveis vulnerabilidades é tão numerosa que empresas renomadas como Apple, HP, IBM, NIST e Red Hat se uniram e elaboraram uma lista formal de vulnerabilidades. Essa lista, denominada *Common Weakness Enumeration* (CWE), representa vulnerabilidades potenciais mais gerais agrupadas em classes comuns em uma estrutura acessível para ser consultada. O CWE possui o registro de cerca de 808 vulnerabilidades que podem ser visualizadas de forma hierárquica de três formas distintas: organizadas por conceitos de pesquisa, de desenvolvimento e arquiteturais (55).

Os ativos de informação, sejam eles meios de armazenamento, transmissão e processamento de dados e informação, os equipamentos necessários a isso, tais como, computadores, equipamentos de comunicação e interconexão; os sistemas utilizados para tal e os sistemas de informação de um modo geral, bem como os locais onde se encontram esses meios e as pessoas que a elas tem acesso, possuem fraquezas tornando-os vulneráveis a ação de pessoas maliciosas (53). E, essas fraquezas dos ativos de informação podem

ser exploradas por uma ou mais ameaças causando um incidente indesejado, que pode resultar em danos a um sistema ou organização, tais como, perda financeira, política e de reputação (56) (57).

Todas as informações e ativos associados com os recursos de processamento da informação devem possuir um proprietário designado. O termo proprietário identifica uma pessoa ou organismo com responsabilidade autorizada para controlar a produção, o desenvolvimento, a manutenção, o uso e a segurança dos ativos. O termo proprietário não significa que a pessoa realmente tenha qualquer direito de propriedade ao ativo (52) e, sim aquela que usa o ativo em determinado evento.

Às vezes, um evento que ocorre em um computador ou rede é parte de uma série de etapas destinadas a resultar em algo que não está autorizado a acontecer, tais como, perda de recurso, aumento do número de acesso, divulgação de informação, corrupção de informação, negação de serviço etc (58) (57). Qualquer tentativa, bem ou mal sucedida, de uma pessoa acessar ou usar, sem autorização, um serviço, computador ou rede é considerada um ataque (19). E, a pessoa responsável pela realização de um ataque é um atacante. O atacante pode ser um terrorista, um hacker, um espião, um vândalo, um profissional do crime ou até mesmo um invasor corporativo (57).

Um ataque tem vários elementos, a começar com uma série de ações tomadas pelo atacante. O atacante usa alguma ferramenta para explorar a vulnerabilidade de um alvo com o objetivo de obter um resultado não autorizado, visto da perspectiva da pessoa responsável pelo ativo de informação alvo. Para tal, o atacante executa uma série de ações mal intencionadas. Isso diferencia um ataque de algo que é inadvertido. Para ser bem sucedido, um ataque deve encontrar caminhos que possam ser conectados (ataques), talvez simultaneamente ou repetidamente (57).

Quando o atacante comete uma série de ataques com o mesmo objetivo, diz-se que esses ataques são parte de um incidente. O que os torna um grupo distinto é uma combinação de fatores. Primeiro, pode haver apenas um atacante ou vários atacantes relacionados de alguma forma. Os atacantes podem usar ataques semelhantes, ou eles podem estar tentando alcançar objetivos distintos. Além disso, os alvos envolvidos nos ataques e o momento dos ataques podem ser os mesmos ou estar relacionados. Quando um grupo de ataques que ocorrem em momentos próximos, causados pelo mesmo atacante, com o mesmo objetivo e atingem um ativo de informação constitui-se um incidente (57).

Há várias técnicas usadas pelo atacantes para explorar vulnerabilidades, as técnicas mais reproduzidas tornam-se padrões de ataque que são tratados pela comunidade de segurança. Desde de 2007, o departamento de segurança interna dos Estados Unidos, vem catalogando os padrões de ataque comuns em uma base denominada *Common Attack Pattern Enumeration and Classification* (CAPEC). A base CAPEC continua a evoluir com participação pública e contribuições para formar um mecanismo para identificar, coletar,

refinar e compartilhar padrões de ataque entre a comunidade de segurança cibernética. Atualmente, a sua estrutura contém 516 padrões de ataque estruturados por mecanismo de ataque e por domínio de ataque que vem sendo usado como referência tanto na indústria quanto na pesquisa. A representação dos mecanismos de ataque organiza os padrões de ataque hierarquicamente com base nos mecanismos que são frequentemente empregados ao explorar uma vulnerabilidade. A representação de domínios de ataque organiza os itens pelos domínios de destino para cada padrão de ataque (59).

Todos esses conceitos levantados foram reunidos no glossário de termos apresentado na Tabela 1.

Tabela 1 – Glossário de Termos

| Termo | Descrição |
|---|---|
| Ameaça | Causa potencial de um incidente indesejado, que pode resultar em danos a um sistema ou organização (56), tais como, perda financeira, política e de reputação (56) (57) |
| Atacante | Pessoa responsável pela realização de um ataque (19). O atacante pode ser um terrorista, um hacker, um espião, um vândalo, um profissional do crime ou até mesmo um invasor corporativo (57). |
| Ataque | Qualquer tentativa, bem ou mal sucedida, de acesso ou uso não autorizado de um serviço, computador ou rede (19). Isto é, uma série de passos dados por um atacante para produzir um resultado não autorizado (57). Há várias técnicas usadas pelo atacantes, as técnicas mais reproduzidas tornam-se padrões de ataque catalogados em uma base denominada *Common Attack Pattern Enumeration and Classification* (CAPEC) (59). |
| Ativo de informação | Meios de armazenamento, transmissão e processamento de dados e informação, os equipamentos necessários a isso (computadores, equipamentos de comunicações e de interconexão), os sistemas utilizados para tal, os sistemas de informação de um modo geral, bem como os locais onde se encontram esses meios e as pessoas que a eles têm acesso (53). |
| Evento de segurança de informação | Uma ocorrência identificada de um estado de sistema, serviço ou rede, indicando uma possível violação da política de segurança da informação ou falha de controles, ou uma situação previamente desconhecida, que possa ser relevante para a segurança da informação (56). |

| | |
|---|---|
| Incidente de segurança | Um simples ou uma série de eventos de segurança da informação indesejados ou inesperados, que tenham uma grande probabilidade de comprometer as operações do negócio e ameaçar a segurança da informação (52). Quando um grupo de ataques que ocorrem em momentos próximos, causados pelo mesmo atacante, com o mesmo objetivo e atingem um ativo de informação constitui-se um incidente (57) |
| Proprietário | Uma pessoa ou organismo que tenha responsabilidade autorizada para controlar a produção, o desenvolvimento, a manutenção, o uso e a segurança dos ativos (52). |
| Segurança de informação | Ações que objetivam viabilizar e assegurar a disponibilidade, a integridade, a confidencialidade e a autenticidade das informações (53). Adicionalmente, outras propriedades como autenticidade, responsabilidade, não-repúdio e confiabilidade também podem estar envolvidas (52). |
| Vulnerabilidade | Fraqueza de um ativo ou controle que pode ser explorada por uma ou mais ameaças (52). Essa condição, quando explorada por um atacante, pode resultar em uma violação de segurança (19). A lista de das vulnerabilidades potenciais mais gerais foram agrupadas em classes comuns em uma estrutura acessível denominada *Common Weakness Enumeration* (CWE) (55). |

## 5.3 Avaliar os termos levantados

As descrições dos termos levantados foram avaliadas para verificar se eas têm informações suficientes para serem usadas para definir formalmente os conceitos do domínio, conforme recomentado na Figura 29. Como resultado, para cada questão de competência, definidas na Seção 5.1, foram verificados os termos cujo significados norteiam a sua resposta. Essa associação encontra-se no Quadro 4.

## 5.4 Definir formalmente os conceitos do domínio

Cada um dos termos do glossário foi analisado conceitualmente utilizando como referencial teórico a UFO-MLT, conforme sugerido no processo *Definir formalmente os conceitos do domínio* da metodologia DEFESA (Figura 31). Tal abordagem tem o objetivo de desambiguar os conceitos, tipificar as entidades do domínio, explicitar suas restrições

Quadro 4 – Associação entre as questões de competência e os termos do domínio

| Questão | Termo(s) |
|---|---|
| QC1- Quais são as características de um evento para classificá-lo como incidente? | O **Incidente de Segurança da Informação** é um simples ou uma série de **Eventos de Segurança da Informação** indesejados ou inesperados, que tenham uma grande probabilidade de comprometer as operações do negócio e ameaçar a **Segurança da Informação**. |
| QC2- Qual foi o dano causado pelo incidente? | **Ameaça** é a causa potencial de um incidente indesejado, que pode resultar em danos a um sistema ou organização, tais como, perda financeira, política e de reputação. |
| QC3- Quem causou o incidente? | O **Atacante** é a pessoa responsável pela realização de um **Ataque**. Quando um grupo de ataques que ocorrem em momentos próximos, causados pelo mesmo atacante, com o mesmo objetivo e atingem um ativo de informação constitui-se um **Incidente**. |
| QC4- Quem sofreu o incidente? | O **Proprietário** com responsabilidade autorizada para controlar a produção, o desenvolvimento, a manutenção, o uso e a segurança dos **Ativos**. |
| QC5- Quando ocorreu o incidente? | Quando um **Evento de segurança da informação** é identificado. |
| QC6- Qual foi a causa do incidente? | **Ameaça** é a causa potencial de Incidente indesejado, que pode resultar em danos a um sistema ou organização. |
| QC7- Por que o incidente ocorreu? | A **Vulnerabilidade** de um ativo ou controle pode ser explorada por uma ou mais **Ameaças**. Essa condição, quando explorada por um **Atacante**, pode resultar em uma violação de **Segurança**. |
| QC8- Como o incidente ocorreu? | O **Incidente de Segurança da informação** consistem em um simples ou uma série de **Eventos de segurança da informação** indesejados ou inesperados, que tenham uma grande probabilidade de comprometer as operações do negócio e ameaçar a segurança da informação. Quando um grupo de **Ataques** que ocorrem em momentos próximos, causados pelo mesmo atacante, com o mesmo objetivo e atingem um ativo de informação constitui-se um incidente. |
| QC9- Como o incidente foi classificado? | O **Incidente de Segurança da informação** é composto por um grupo de **Ataque**. Os padrões de ataque são catalogados na base *Common Attack Pattern Enumeration and Classification* (CAPEC). |

e representar o relacionamento entre elas. Além disso, prover uma caracterização formal dos tipos e subtipos que são fundamentais nesse domínio. Em virtude dos termos da UFO-MLT serem em inglês, todos os elementos dos diagramas estão em inglês, inclusive os elementos do domínio. Para tal, a ferramenta *Enterprise Architect* foi usada para a construção dos esteriótipos da UFO e da MLT. E, para facilitar a identificação o emprego desses esteriótipos na representação dos elementos do domínio, os diagramas seguem os padrões de cores apresentados na Figura 38. As entidades do domínio que são instâncias de algum tipo da UFO seguem o padrão de cores para elementos específicos da UFO (Specific elements of UFO) exibido na parte superior da figura. E nos diagramas da UFO-MLT, os elementos da MLT são identificados pela cor amarela, os conceitos universais (*Universals*) da taxonomia de UFO pela cor cinza, os tipos que são especializações próprias da UFO e suas instâncias são identificados pela cor branca. Quando esses tipos forem especializados em muitos níveis cada nível tem a borda de uma cor diferente (azul, verde, vermelho, laranja, roxo) para facilitar a identificação.

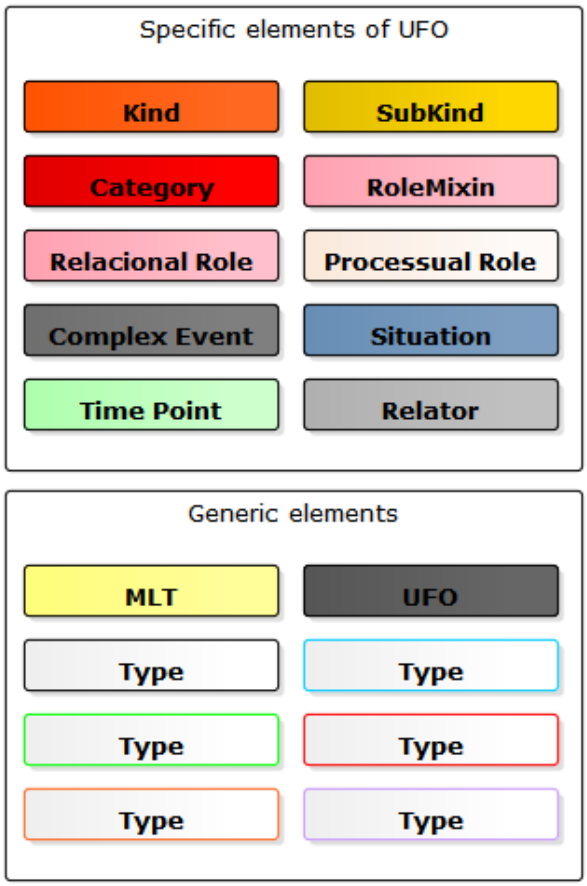



Figura 38 – Padrões de cores dos modelos

O Incidente de Segurança da Informação ocorre em um ambiente computacional e pode envolver uma diversidade de elementos, como computadores, equipamentos de rede, software entre outros. Esses ativos de informação são usados para diversos fins, inclusive

como ferramenta para execução de tarefas ou para prática de atividades maliciosas. Analisando ontologicamente, o ativo de informação representa uma categoria (*category*) de recursos computacionais de diferentes princípios de identidade. Cada um deles pode ser de um tipo (*kind*), tais como, computador (*Computer*), equipamento de rede (*Network Equipament*) e *software*. A Figura 39 modela a categoria (*category*) ativo de informação (*Information Asset*).

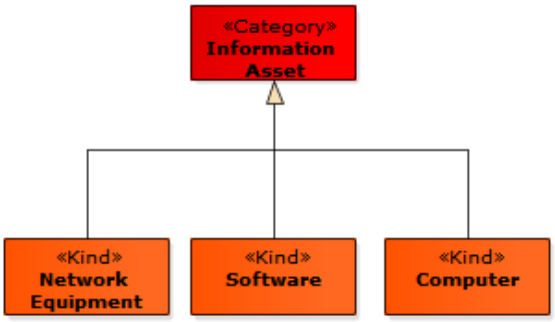



Figura 39 – Modelo baseado na UFO da categoria ativo de informação

Todos os ativos de informação devem possuir um proprietário designado, isto é, uma pessoa que ao usá-lo torna-se responsável por ele. Desta forma, um tipo (*kind*) de pessoa (*Person*), podendo ser do subtipo (*subkind*) pessoa física (*Natural Person*) ou pessoa jurídica (*Juridical Person*), tem uma relação material (*relator*) de aquisição de posse (*Ownership acquisition*) com um ativo de informação (*Information Asset*). Neste relacionamento a pessoa desempenha o papel relacional (*Relational Role*) de responsável (*Responsible*) por um ativo de informação e, consequentemente, qualquer tipo de ativo de informação desempenha o papel relacional (*RoleMixin*) de tutelado (*Tutored*), conforme demonstrado na Figura 40.

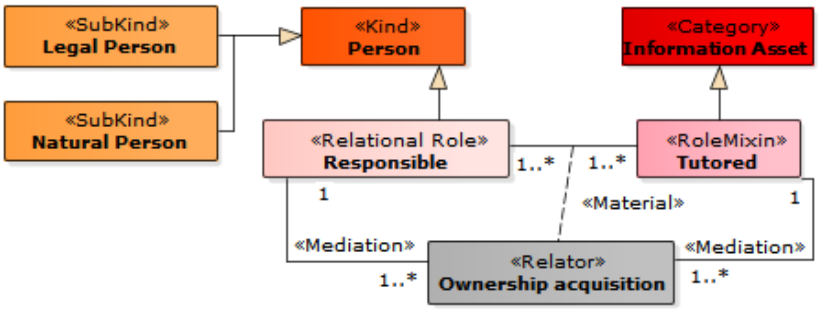



Figura 40 – Modelo baseado na UFO do relacionamento entre pessoa e ativo de informação

O Incidente de Segurança de Informação ocorre quando uma pessoa realiza um simples ou uma série de eventos que provoquem algum dano a uma outra pessoa. O incidente (*Incident*) consiste em um evento complexo (*Complex Event*) que tem a participação de (*participation of*) uma pessoa no papel processual (*Processual Role*) de atacante (*Attacker*)

e outra pessoa no papel processual (*Processual Role*) de vítima (*Victim*), o atacante em um incidente leva a vítima a uma situação (*situation*) de dano (*Damage*). A Figura 41 representa o momento (*Incidente Moment*) em que a vítima sofre um dano causado por um atacante.

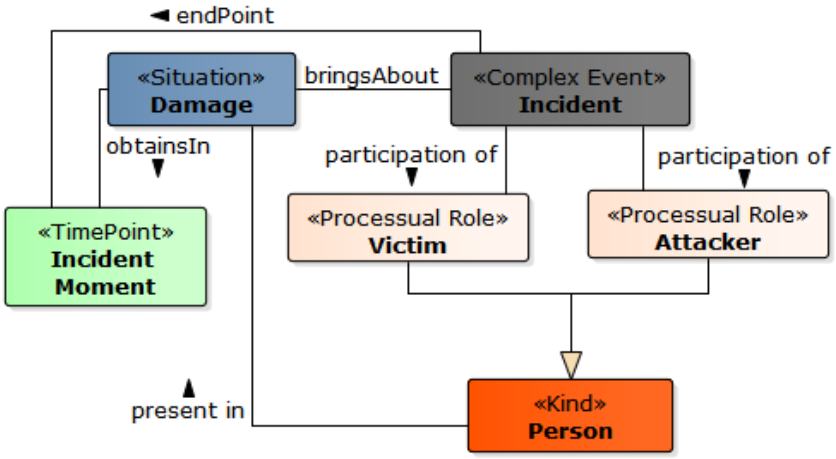



Figura 41 – Modelo baseado na UFO do evento incidente

Para que o atacante consiga causar um dano à vítima, ele usa um ativo de informação de forma maliciosa para realizar uma série de ataques ao ativo de informação da vítima. Cada ataque explora uma vulnerabilidade de um ativo de informação alvo para acessá-lo e realizar um uso não autorizado. Sendo assim, o incidente (*Incident*) é um evento do tipo complexo (*Complex Event*) composto por ataques, conforme mostrado na Figura 42. Cada ataque tem a participação do (*participation of*) ativo de informação (*Information Asset*) desempenhando o papel processual (*Processual Role*) de malicioso (*Malicious*) e a participação do (*participation of*) ativo de informação (*Information Asset*) desempenhando o papel processual (*Processual Role*) de alvo (*Target*). O ataque somente ocorre se o ativo de informação alvo manifestar disposições que façam com que ele esteja (*triggers*) em uma situação (*Situation*) vulnerável (*Vulnerable*) à ação do ativo de informação malicioso e, após o ataque, este leva (*bringsAbout*) o ativo de informação alvo a apresentar um resultado não autorizado (*Unauthorized Result*). Logo, o ataque se concretiza no momento (*Attack Moment*) em que o ativo de informação alvo passa a apresentar um resultado não autorizado. Em virtude do propósito ser a representação do incidente que já ocorreu, e não da prevenção de incidentes, o modelo criado não contempla as manifestações das disposições do ativo de informação alvo durante o ataque. No sentido estender o modelo para contemplar essa parte, recomenda-se consultar o trabalho de Duarte et al.(60).

As principais entidades ontológicas e relações definidas até agora foram reunidas em sCuDO conforme ilustrado na Figura 43.

A Figura 43 apresenta o contexto em que o incidente ocorre, os participantes e seus respectivos papéis, bem como seus impactos. Porém, para oferecer maior expressividade o

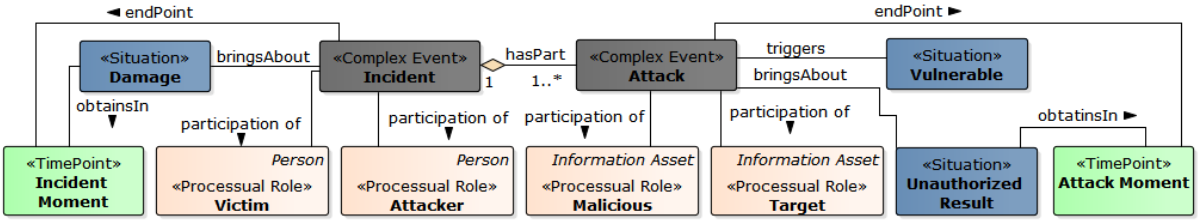



Figura 42 – Modelo baseado na UFO do evento incidente e do evento ataque

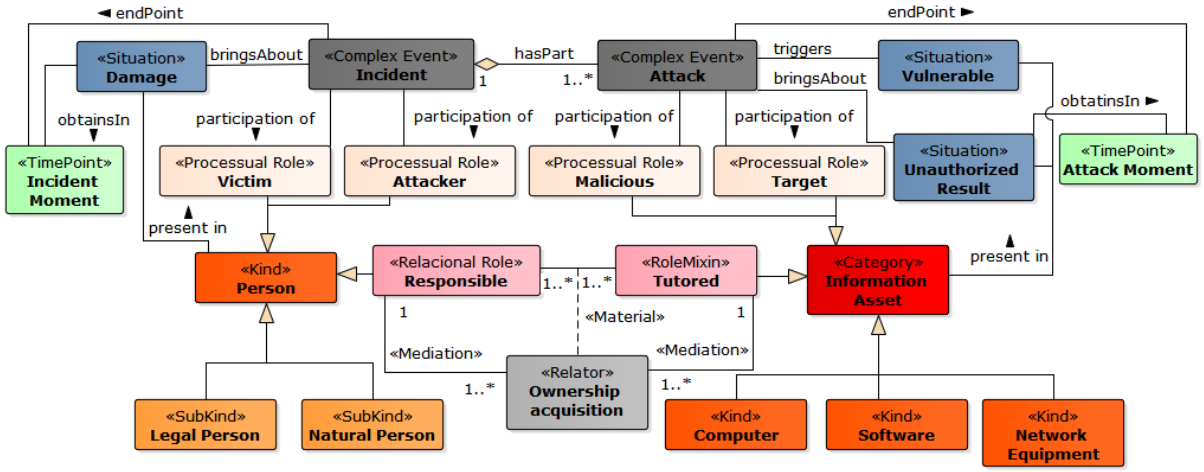



Figura 43 – Modelo baseado na UFO de Incidente de Segurança da informação

modelo precisa representar a hierarquia de tipos das entidades para dirimir ambiguidades, pois, o esquema de categorização em si é importante neste domínio. Essa estrutura dá origem a hierarquias de tipos nos quais os tipos mais específicos geralmente formam uma partição de um tipo mais geral, distinguindo instâncias de acordo com um critério de classificação específico. O sistema de categorização de sCuDo é baseado na MLT. O primeiro passo foi transformar o modelo conceitual baseado na UFO em um modelo multi-nível, conforme a Figura 44.

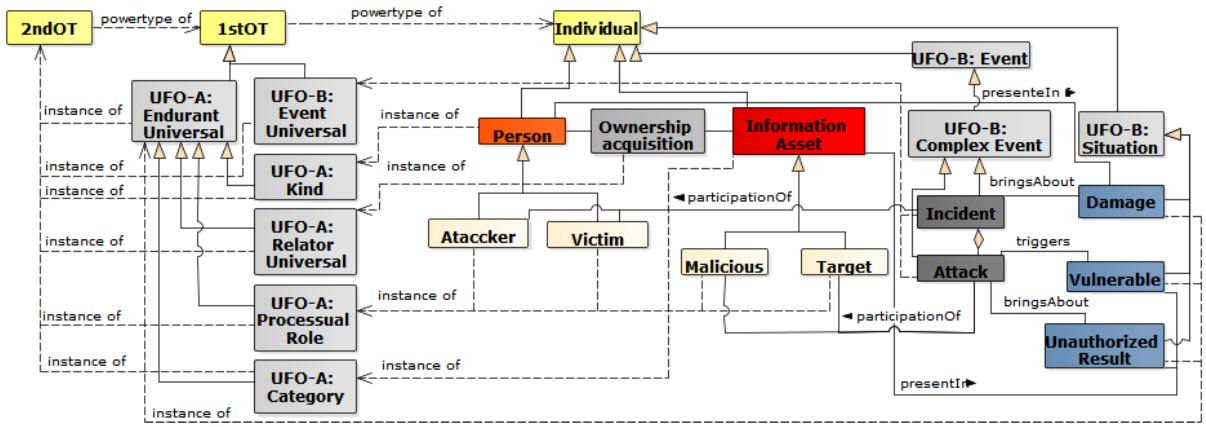



Figura 44 – Modelo baseado na UFO-MLT de Incidente de Segurança da informação

O uso de UFO-MLT permite que as entidades sejam classificadas em tipos utilizando os esteriótipos da UFO e os critérios de classificação da MLT. As entidades de sCuDo são próprias especializações de *Individual* e instâncias de um dos elementos da árvore de *Universal* da UFO. E, os tipos de segunda ordem do domínio são especializações de uma das categorias taxonômicas de *Universal*.

Desta forma, a entidade Pessoa (*Person*) é uma especialização própria de *Individual* e uma instância de *UFO-A: Kind*. Pessoa pode ser especializada de acordo com o aspecto legal em Pessoa Física e Pessoa Jurídica. O subtipo de pessoa de acordo com o aspecto legal (*Person legal aspect SubKind*) é uma especialização de *UFO-A: SubKind* que particiona (*partitions*) pessoa (*Person*). Sendo assim, suas instâncias (*instance of*) pessoa física (*Natural Person*) e pessoa jurídica (*Legal Person*) são especializações de pessoa.

A pessoa é responsável por um ativo de informação logo, ela desempenha o papel relacional (*Relational Role*) de responsável. Então, responsável (*Responsible*) é uma instância de papel relacional de pessoa (*Person Relational Role*) e uma especialização própria de pessoa (*Person*). As pessoas, responsáveis por ativos de informação, quando participam de um incidente desempenham o papel de atacante ou de vítima. Então, o papel de pessoa no evento incidente (*Person Processual Role*) especializa *UFO-A: Processual Role*, categoriza (*categorizes*) a pessoa e tem como instâncias as especializações próprias de pessoa: atacante (*Attacker*) e vítima (*Victim*). Por fim, o atacante pode ser um terrorista (*Terrorist*), um hacker (*Hacker*), um espião (*Spy*), um vândalo (*Vandal*), um profissional do crime (*Professional Criminal*) e até mesmo um invasor corporativo (*Corporate Raider*) (57). Esses tipos de atacante (*Attacker Type*) são especializações próprias do papel desempenhado pela pessoa no incidente (*Person Processual Role*). A Figura 45 ilustra a pessoa e seus tipos.

A entidade ativo de informação é uma especialização própria de *Individual* e uma instância de *UFO-A: Category*. A categoria Ativo de Informação generaliza as diferentes propriedade de diversos tipos de ativo de informação, isto é, o tipo de ativo de informação (*Information Asset Kind*) é uma instância de segunda ordem que especializa *UFO-A: Kind*, particiona (*partitions*) ativo de informação e tem como instâncias computador (*Computer*), equipamento de rede (*Network equipment*) e *software* (57). O ativo de Informação que participa de um ataque desempenha o papel processual de malicioso ou de alvo. Para tal, o papel processual do ativo de informação no ataque (*Information Asset Processual Role*) foi o tipo de segunda ordem definido como especialização de *UFO-A: Processual Role* que categoriza (*categorizes*) o ativo de informação (*Information Asset*) e tem como instâncias malicioso (*Malicous*) e alvo (*Target*). Todo ativo de informação tem uma pessoa responsável por ele. Neste relacionamento, o ativo de informação desempenha o papel relacional de tutorado. Logo, o papel relacional desempenhado por todos os tipos de ativo de informação (*Information Asset RoleMixin*) é uma especialização de *UFO-A:RoleMixin*

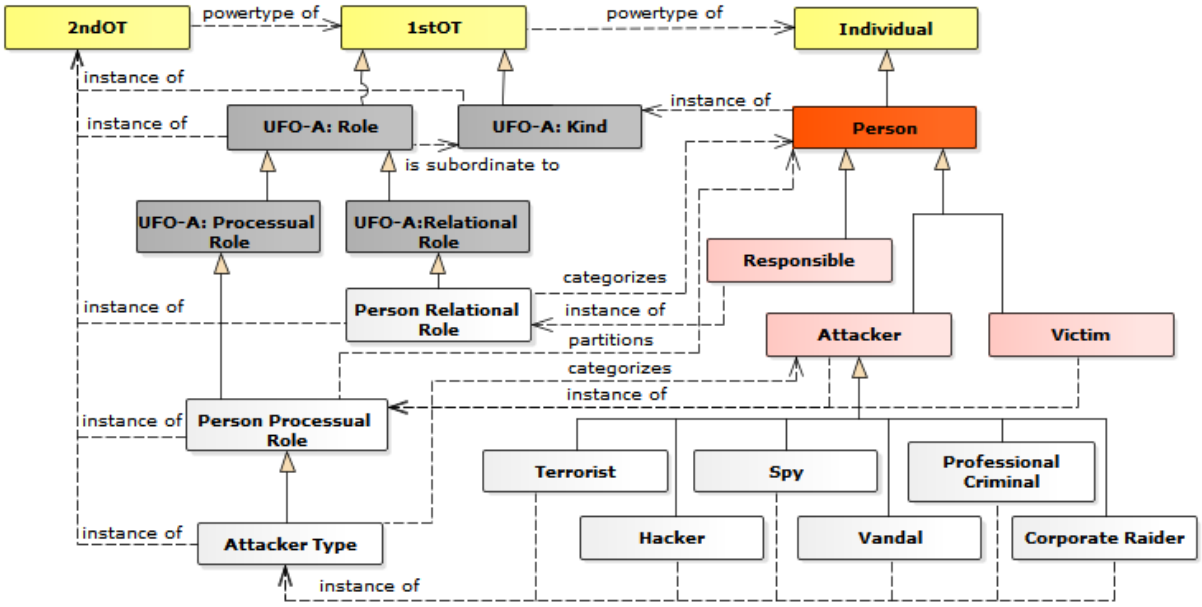



Figura 45 – Modelo baseado na UFO-MLT de pessoa e seus tipos

que categoriza (*categorizes*) ativo de informação (*Information Asset*) e tem como instância tutorado (*Tutored*). A Figura 46 ilustra os tipos de ativo de informação.

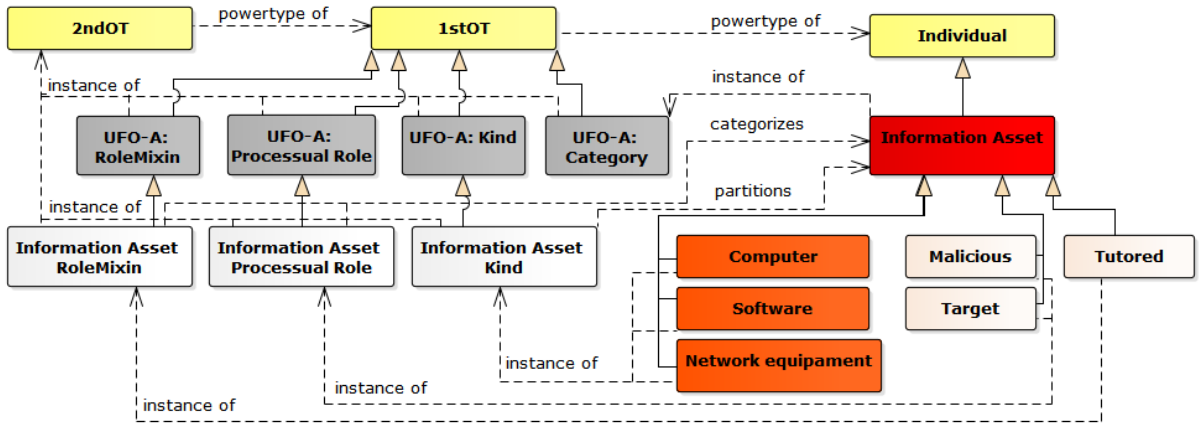



Figura 46 – Modelo baseado na UFO-MLT de ativo de informação e seus tipos

Os eventos de sCuDo também são representados utilizando UFO-MLT. O Incidente, evento composto por um ou mais ataques, é uma especialização de *UFO-B: Complex Event* e instância de (*instance of*) *UFO-B: Event Universal*. O Ataque por possuir vários participantes, também é uma especialização de *UFO-B: Complex Event* e instância de (*instance of*) *UFO-B: Event Universal*. Os eventos causam situações que em que os substanciais estão presentes, no caso o incidente, uma pessoa desempenhando papel de vítima sofre um dano. O dano (*Damage*) é instância de *UFO-A: Situation Universal* e especialização própria de *UFO-A: Situation*, situação esta que a pessoa (*Person*) fica presente (*present in*) após um incidente (*bringsAbout*). O evento ataque muda o estado

de um ativo de informação (*Information Asset*) alvo (*Target*) de vulnerável (*triggers Vulnerable*) e para resultado não autorizado (*bringsAbout Unauthorized Result*).

As situações podem ser especializadas para tornar a representação mais detalhada. O dano causado por um incidente pode ser político, financeiro ou na reputação, dando origem a instância de segunda ordem tipo de Dano (*Damage Type*) que é uma especialização de *UFO-A: Situation Universal* que categoriza (*categorizes*) dano e tem como instância perda financeira (*Financial*), política (*Political*) e de Reputação (*Reputation*) (57). A Figura 47 ilustra os tipos de situação de dano.

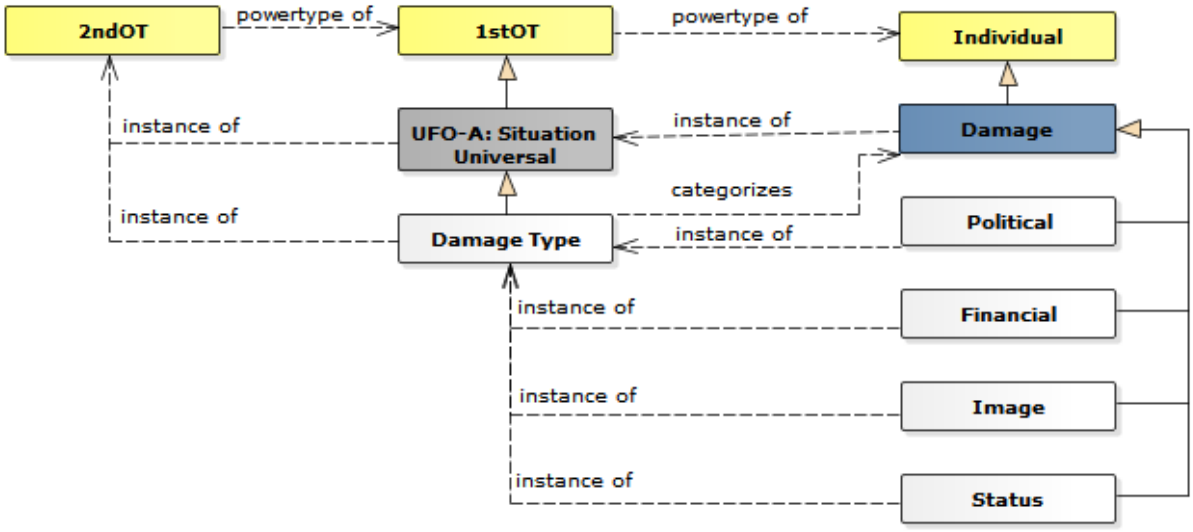



Figura 47 – Modelo baseado na UFO-MLT da situação de dano e seus tipos

A vulnerabilidade, situação propícia ao ataque, pode ser causada por falhas na implementação, no desenvolvimento ou na configuração do ativo de informação etc. A lista de vulnerabilidades da base de dados do CWE é utilizada para categorização das vulnerabilidades de sCuDO.

O CWE possui o registro de cerca de 808 vulnerabilidades que podem ser visualizadas de forma hierárquica de três formas distintas: organizadas por conceitos de pesquisa, de desenvolvimento e arquiteturais. Em sCuDo, a classificação do CWE (*Vulnerable Type by CWE Classification*) é usada para categorizar (*categorizes*) *UFO-A Situation Universal* que tem como especialização própria o tipo de situação vulnerável por conceitos de pesquisa (*Vulnerable Type by Research Concepts*), de desenvolvimento (*Vulnerable Type by Development Concepts*) e arquitetural (*Vulnerable Type by Architectural Concept*). Esses tipos categorizam (*categorizes*) a situação vulnerável (*Vunerable*) que tem como especializações próprias as instâncias deles. Para demonstrar esta categorização, a Figura 48 ilustra as instâncias do tipo de situação vulnerável por conceitos de pesquisa (*Vulnerable Type by Research Concepts*).

Os tipos de situação vulnerável por conceitos de pesquisa, desenvolvimento e arqui-

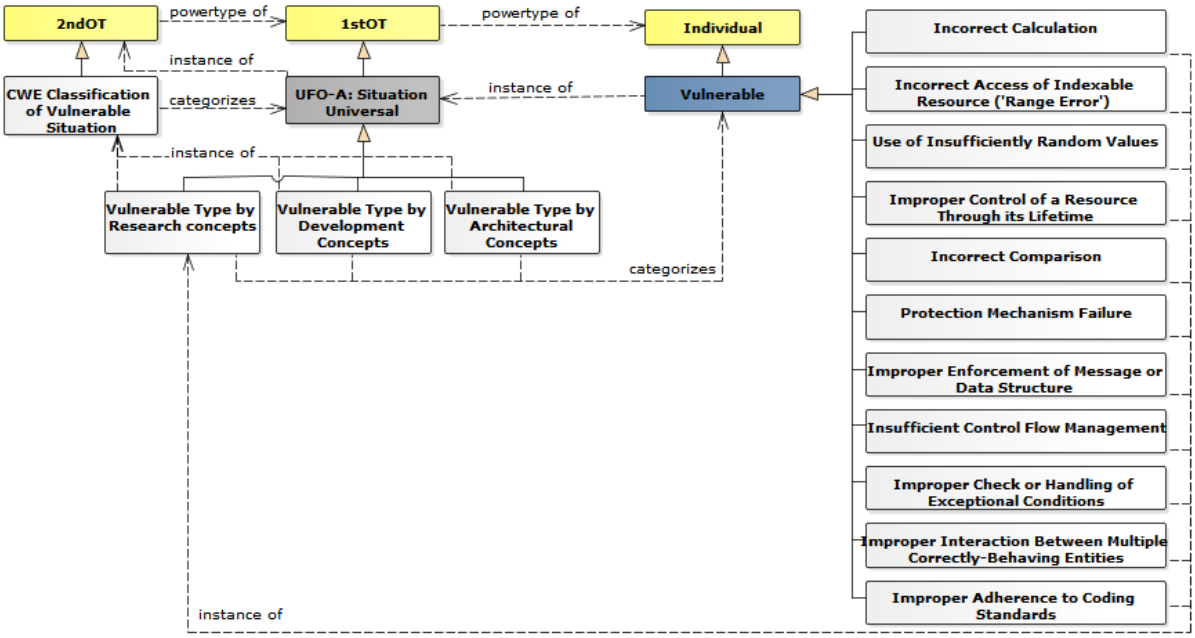



Figura 48 – Modelo baseado na UFO-MLT da situação vulnerável e seus tipos

teturais tem muitas instâncias, por isso, na Figura 48 somente estão sendo representadas as 11 instâncias do tipo de situação vulnerável por conceitos de pesquisa (*Vulnerable Type by Research Concept*). E, um extrato da especialização das categorias *Improper Control of a Resource Through its Lifetime* e *Protection Mechanism Failure* é mostrado na Figura 49. A categoria *Improper Control of a Resource Through its Lifetime* é especializada em vários tipos, como por exemplo, o *Improper Access Control* e *Exposure of Resource to Wrong Sphere*. E, este último tipo, tem o subtipo *Insufficiently Protected Credentials*.

Conforme ilustrado na Figura 49, o tipo *Improper Access Control* é uma especialização de *Improper Control of a Resource Through its Lifetime* e de *Protection Mechanism Failure*. O tipo *Improper Access Control* tem vários subtipos, dentre eles, o *Improper Authentication* que é especializado em *Insufficiently Protected Credentials*. Alguns outros tipos de situação vulnerável encontram-se no Apêndice C.

Quando o ativo de informação em situação vulnerável é atacado, ele passa a apresentar um resultado não autorizado. Esta situação pode ser de vários tipos, tais como, perda de recurso, aumento do número de acesso, divulgação de informação, corrupção de informação, negação de serviço etc (57). O tipo de resultado não autorizado (*Unauthorized Result Type*) é uma especialização de *UFO-A: Situation Universal* que categoriza (*categorizes*) o resultado não autorizado (*Unauthorized Result*) e tem como instância (*instance of*) perda de recursos (*Theft of Resources*), aumento no número de acesso (*Increased Access*), divulgação de informação (*Disclosure of Information*), corrupção de informação (*Corruption of Information*), negação de serviço (*Denial of Service*). A Figura 50 ilustra os tipos de resultado não autorizado.

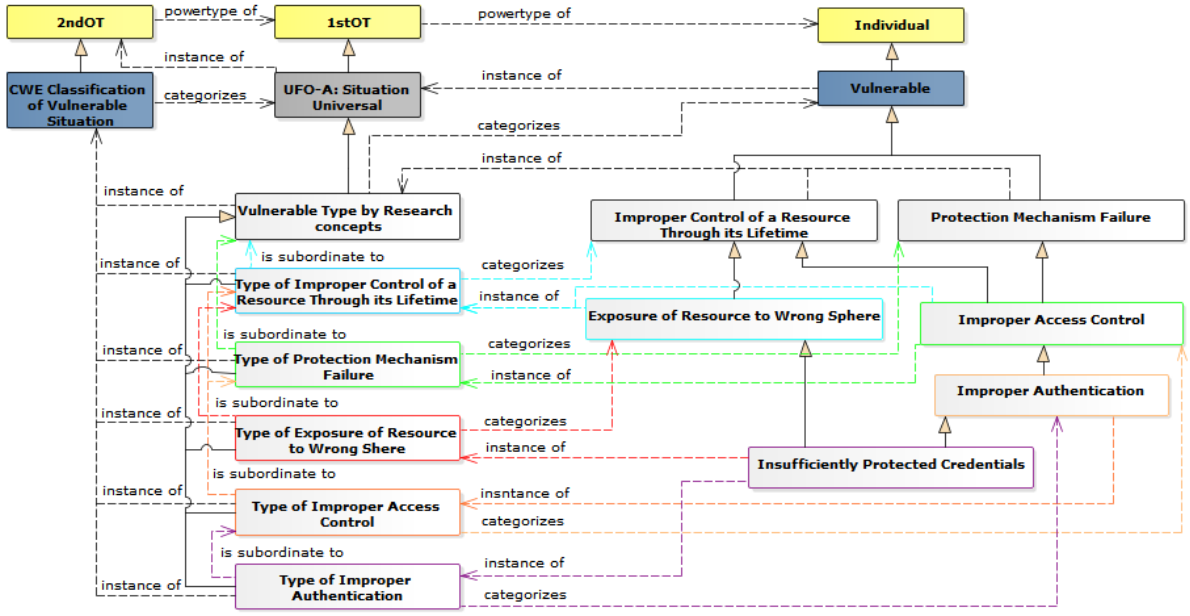



Figura 49 – Modelo baseado na UFO-MLT de subtipos de situação vulnerável *Improper Control of a Resource Through its Lifetime*

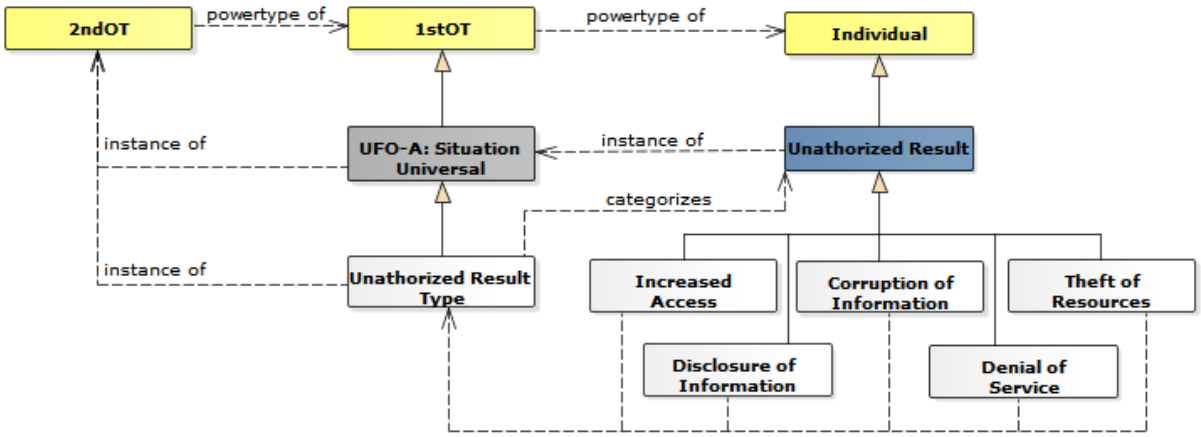



Figura 50 – Modelo baseado na UFO-MLT da situação resultado não autorizado e seus tipos

Os incidentes geralmente são relatados e classificados de acordo com o tipo de ataque que os ocasionaram. Sendo assim, a representação de tipos de ataque é importante no sCuDo. Devido à variedade de tipos e formas de classificação, na construção do modelo os tipos de ataque foram hierarquizados de acordo com a base de dados de enumeração e classificação comum de padrão de ataque (*Common Attack Pattern Enumeration and Classification – CAPEC*).

O CAPEC organiza 516 padrões de ataque por mecanismo de ataque e por domínio de ataque. Em sCuDO, o tipo de ataque por mecanismo de ataque (*Attack Type by Mechanism*) e o tipo de ataque por domínio de ataque (*Attack Type by Domain*) são

instâncias de segunda ordem de classificação do tipo de ataque usando a base de dados do CAPEC (*Attack Type by CAPEC Classification*) e especializações próprias de *UFO-B: Event Universal* que categorizam (*categorizes*) ataque (*Attack*). A Figura 51 ilustra as categorias de ataque e seus tipos.

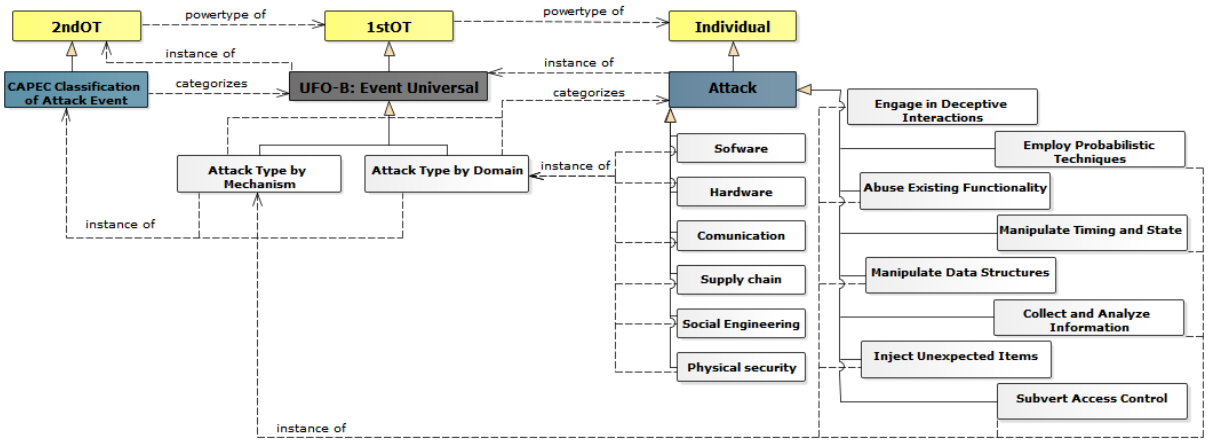



Figura 51 – Modelo baseado na UFO-MLT de ataque e seu tipos

Os tipos de ataque são especializados em outros subtipos e assim sucessivamente. A Figura 52 ilustra os subtipos do tipo de ataque *Subvert Access Control* por mecanismo de ataque. O mecanismo de ataque *Subvert Access Control* tem como uma de suas especializações o tipo *Exploitation of Trusted Credentials* que pode ser especializado em *Use of Known Domain Credentials*. E, este subtipo pode ser especializado em *Remote Services with Stolen Credentials*. Alguns outros tipos de ataque encontram-se no Apêndice B.

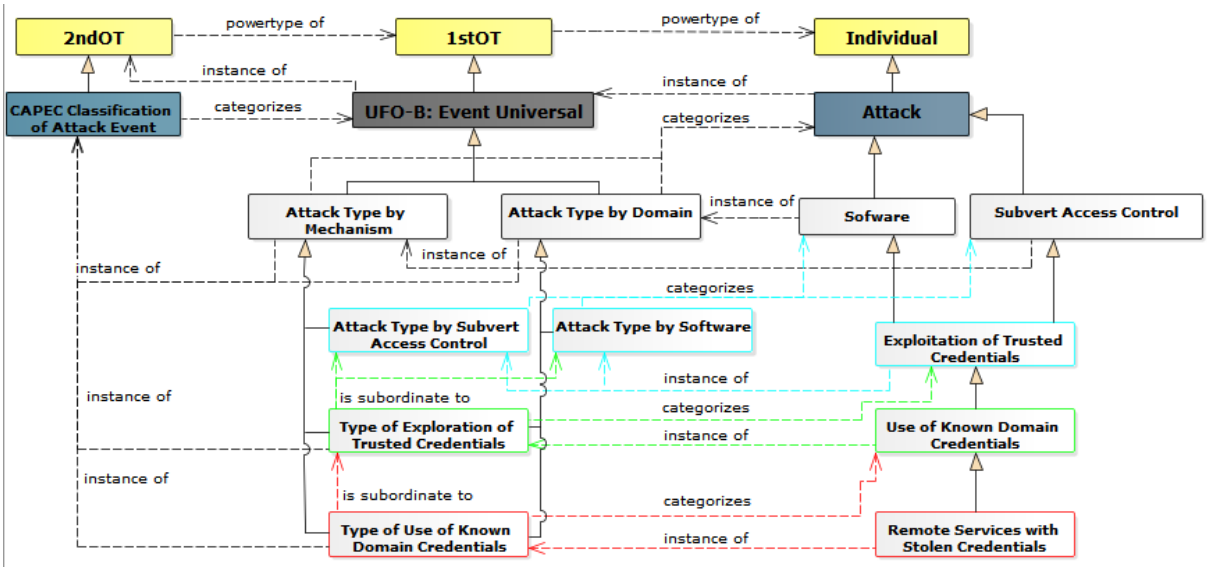



Figura 52 – Modelo baseado na UFO-MLT de tipo de ataque *Subvert Access Control* categorizado por mecanismo de ataque

Conforme ilustrado na Figura 52, o tipo de ataque *Exploitation of Trusted Credentials* também é uma especialização própria do tipo de ataque por *Software* que é uma instância do tipo de ataque por domínio de ataque (*Attack Type by Domain*). Isto é, esse tipo de ataque é um dos muitos tipos que podem ser categorizados tanto pela estrutura de tipos de ataque por domínio de ataque quanto pela estrutura de tipos de ataque por mecanismo de ataque.

Uma das principais vantagens de utilizar o CAPEC e o CWE para categorizar respectivamente os tipos de ataque e os tipos de situação vulnerável é que eles são originalmente correlacionados. Dos 516 padrões da ataque da base do CAPEC, 374 deles têm pelo menos um tipo de vulnerabilidade do CWE associada. E, esse relacionamento é representado em sCuDO. Por exemplo, o tipo de ataque *Use of Known Domain Credentials*, citado na Figura 52, é causado quando há o tipo de situação vulnerável *Insufficiently Protected Credentials*, citado na Figura 49. A Figura 53 mostra a relação entre esse tipo de ataque e esse tipo de situação vulnerável.

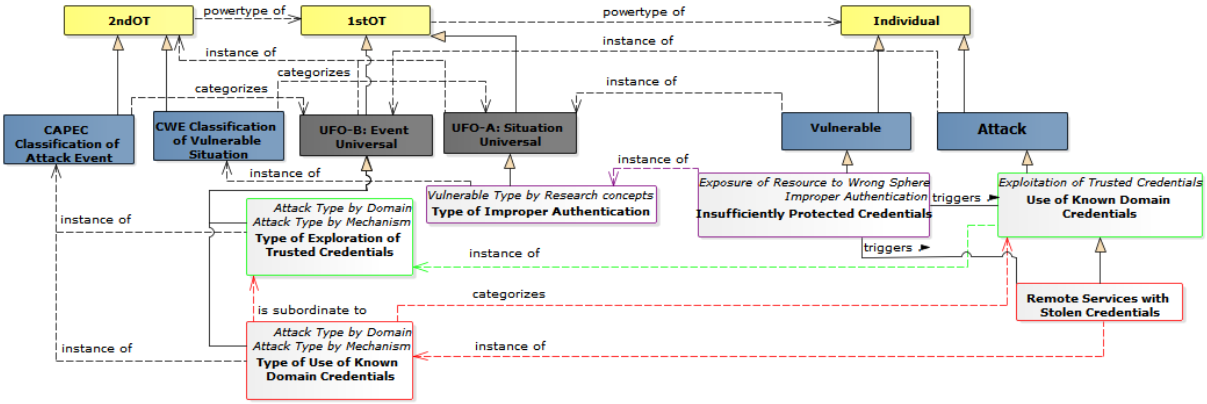



Figura 53 – Exemplo da relação entre tipo de ataque e situação vulnerável

Ao longo dessa seção os conceitos de Incidente de Segurança da Informação foram apresentados através de uma ontologia baseada no UFO-MLT. O sCuDO representa conceitos de domínio, como em (10) e (11), mas o uso de estereótipos da UFO como instâncias de tipos de segunda ordem promoveu maior expressividade semântica. Por exemplo, a entidade incidente (*Incident*), no sCuDO, é uma instância do *UFO B: Event Universal*, especialização própria de *UFO-B: Complex Event* e composto de ataque (*Attack*). O esquema de categorização de tipos de sCuDO se assemelha ao proposto em (11), (12) e (14) na identificação dos tipos de ataque. No entanto, o sCuDo tem o diferencial de elucidar as características de um tipo e o critério de especialização. Essa classificação de ataque permite inferir que o *HTTP flood* é uma especialização própria de *Flooding* que, por sua vez, é uma especialização própria de abuso de funcionalidade existente (*Abuse existing funcionality*)(ver Apêndice B). Esse tipo de relacionamento facilita a comunicação e o intercâmbio de informações.

Dessa forma, o sCuDO representa tipos e subtipos de ataque, vulnerabilidade, resultado não autorizado, atacante e dano com foco em evidenciar o critério de classificação para facilitar a correlação entre ocorrências de incidentes originárias de fontes distintas ou que usaram diferentes critérios de classificação. Por exemplo, supondo que houve dois incidentes semelhantes ao incidente descrito na seção 1.1, ou seja, alguns ataques tornaram alguns *sites* indisponíveis. E se um desses incidentes fosse registrado usando a ontologia de Ping, Haifeng e Guoqing(11) (ilustrada na Figura 18), o incidente seria registrado como causado pela técnica de intrusão (*intrusionTech*) *SYNflood*. E, se a segunda ocorrência de incidente tivesse sido relatada usando a ontologia Ansarinia et al.(14) (ilustrada na Figura 21), o ataque seria do tipo *HTTP Flood* (CAPEC-469). Olhando exclusivamente para os tipos atribuídos, as duas ocorrências podem ser interpretadas como de tipos distintos, no entanto, usando o sCuDO, seria possível associar o tipo *HTTP Flood* de Ansarinia et al.(14) com a *HTTP Flood* do sCuDO que é a especialização própria de *Flood*, que na ontologia de Ping, Haifeng e Guoqing(11) é chamado de *SYNFlood*, possibilitando correlacionar ambas as ocorrências.

## 5.5 Avaliar os conceitos definidos

Uma vez que sCuDO foi modelado deve ser verificado se as entidades expressas no modelo atendem ao especificado. Para isso, a Tabela 2 mostra a correspondência entre as questões de competência e os conceitos do domínio.

Tabela 2 – Associação entre as questões de competência e os conceitos do domínio

| QC1- Quais são as características de um evento para classificá-lo como incidente? |
|---|
| O incidente (*Incident*) consiste de um evento complexo (*Complex Event*) que tem a participação de (*participation of*) uma pessoa no papel processual (*Processual Role*) de atacante (*Attacker*) e outra pessoa no papel processual (*processual role*) de vítima (*Victim*), o atacante em um incidente leva a vítima a uma situação (*situation*) de dano (*Damage*) (ver Figura 43). |
| QC2- Qual foi o dano causado pelo incidente? |
| O dano causado por um incidente pode ser político, financeiro ou na reputação, dando origem a instância de segunda ordem tipo de Dano (*Damage Type*) que é uma especialização de *UFO-A: Situation Universal* que categoriza (*categorizes*) dano e tem como instância perda financeira (*Financial*), política (*Political*) e de Reputação (*Reputation*) (ver Figura 47). |
| QC3- Quem causou o incidente? |

| |
|---|
| O incidente é causado por uma pessoa, podendo ser pessoa física ou pessoa jurídica, desempenhado o papel processual de atacante. O atacante pode ser um terrorista (*Terrorist*), um hacker (*Hacker*), um espião (*Spy*), um vândalo (*Vandal*), um profissional do crime (*Professional Criminal*) e até mesmo um invasor corporativo (*Corporate Raider*). Esses tipos de atacante (*Attacker Type*) são especializações próprias do papel desempenhado pela pessoa no incidente (*Person Processual Role*) que categorizam (*categorizes*) atacante (*Attacker*). (ver Figura 45). |
| QC4- Quem sofreu o incidente? |
| Uma pessoa (*Person*), podendo ser pessoa física (*Natural Person*) ou pessoa jurídica (*Juridical Person*), desempenhando papel processual de vítima (*Victim*) sofre um dano no incidente (ver Figura 43). |
| QC5- Quando ocorreu o incidente? |
| O incidente ocorre no momento em que a vítima sofre o dano (ver Figura 43). |
| QC6- Qual foi a causa do incidente? |
| Para que o atacante consiga causar um dano a vítima, ele usa um ativo de informação de forma maliciosa para realizar uma série de ataques ao ativo de informação da vítima. Cada ataque explora uma vulnerabilidade de um ativo de informação alvo para acessá-lo e realizar um uso não autorizado. Sendo assim, o incidente (*Incident*) é um evento do tipo complexo (*Complex Event*) composto por ataques. Cada ataque tem a participação do (*participation of*) ativo de informação (*Information Asset*) desempenhando o papel processual (*Processual Role*) de malicioso (*Malicious*) e do (*participation of*) ativo de informação (*Information Asset*) desempenhando o papel processual (*Processual Role*) de alvo (*Target*). O ataque somente ocorre se o ativo de informação alvo deve estiver (*triggers*) em uma situação (*Situation*) vulnerável (*Vulnerable*) a ação do ativo de informação malicioso e, após o ataque (*bringsAbout*) o ativo de informação alvo passa a apresentar um resultado não autorizado (*Unauthorized Result*) (ver Figura 43). |
| QC7- Por que o incidente ocorreu? |
| A vulnerabilidade, situação propícia ao ataque, pode ser causada por falhas na implementação, no desenvolvimento ou na configuração do ativo de informação etc. A lista de vulnerabilidades CWE é utilizada de base para categorização das vulnerabilidades de sCuDO. (ver Figura 48). |
| QC8- Como o incidente ocorreu? |
| Quando o ativo de informação em situação vulnerável é atacado, ele passa a apresentar um resultado não autorizado. Esta situação pode ser de vários tipos, tais como, perda de recurso, aumento do número de acesso, divulgação de informação, corrupção de informação, negação de serviço etc. A vítima, proprietária, de um ativo de informação alvo de ataque sofre um dano (ver Figura 43). |

| QC9- Como o incidente foi classificado? |
| --- |
| Os incidentes geralmente são relatados e classificados de acordo com o tipo de ataque que os ocasionaram. Os tipos de ataque foram hierarquizados de acordo com a base de dados do CAPEC (ver Figura 51). |

# 6 CENÁRIO DE APLICAÇÃO DE SCUDO

Este capítulo tem o objetivo de usar sCuDO para representar ocorrências de incidentes em três perspectivas distintas. Primeiramente sCuDO será usado para representar um incidente ocorrido no Irã. O incidente foi instanciado baseado em sCuDO explicitando o ativo de informação malicioso utilizado pelo atacante para atacar, os ataques envolvidos no incidente, os alvos atingidos e o dano causado à vítima. Após o incidente ser representado usando sCuDO, será verificado se através do modelo consegue-se responder as questões de competência especificadas. Posteriormente, um outro incidente, envolvendo uma série de ataques, será detalhado usando sCuDO com enfoque na aplicabilidade da hierarquização de tipos de sCuDO. E, por fim, uma base de dados de ocorrências de incidentes de um Grupo de Resposta a Incidentes de Segurança em Computadores (*Computer Security Incident Response Team* - CSIRT) será usada para mostrar uma possibilidade de instanciação de sCuDO usando os valores de seus campos.

## 6.1 Representação do Incidente do Irã

Para demonstrar um uso prático de sCuDO para representar um incidente e verificar se este modelo tem a completude especificada, um incidente histórico ocorrido no Irã será modelado.

Durante os protestos contra as eleições presidenciais iranianas em 2009, o *software* Slowloris foi criado para ser utilizado de forma maliciosa para atacar *sites* administrados pelo governo iraniano, como o 'www.leader.ir' e o 'www.president.ir'. Slowloris foi desenvolvido por Robert 'Rsnake' Hanser usando a linguagem perl para causar negação de serviço (61). Ele cria um fluxo de solicitações TCP SYN e o envia para um alvo e as mantém abertas o maior tempo possível. Para fazer isso, envia continuamente solicitações HTTP parciais e não conclui nenhuma delas. O alvo recebe as solicitações, abre conexões e aguarda a conclusão de cada solicitação. Por fim, o *pool* de conexões simultâneas máximo do alvo é preenchido e tentativas legítimas de conexão são negadas (62).

De acordo com o sCuDo, o ataque ao *site* 'www.leader.ir' e o ataque ao *site* 'www.president.ir' compõem o incidente. Robert 'Rsnake' Hanser é a pessoa que desempenha o papel de atacante e o governo iraniano a vítima. Durante o incidente, o atacante Robert 'Rsnake' Hanser usou o *software* Slowloris, que é um tipo de ativo de informação, para causar danos políticos em protesto as eleições presidenciais iraniana. O Slowloris foi usado para inundar computadores hospedeiros dos *sites* com solicitações HTTP (*HTTP Flood*). Cada computador que hospeda um *site* é capaz de aceitar solicitações HTTP. Esses ataques ocorreram porque os computadores hospedeiros estavam na situação vulnerável de

permitir a alocação de recursos sem limites (*Allocation of Resources Without Limits or Throttling*). Esse fato permitiu que eles aceitassem solicitações HTTP parciais até estourar a área de armazenamento do *pool* de conexões. Quando os *sites* foram atacados, eles se tornam indisponíveis (Denial of Service), que é um resultado não autorizado. A Figura 54 representa o Incidente do Irã utilizando sCuDO.

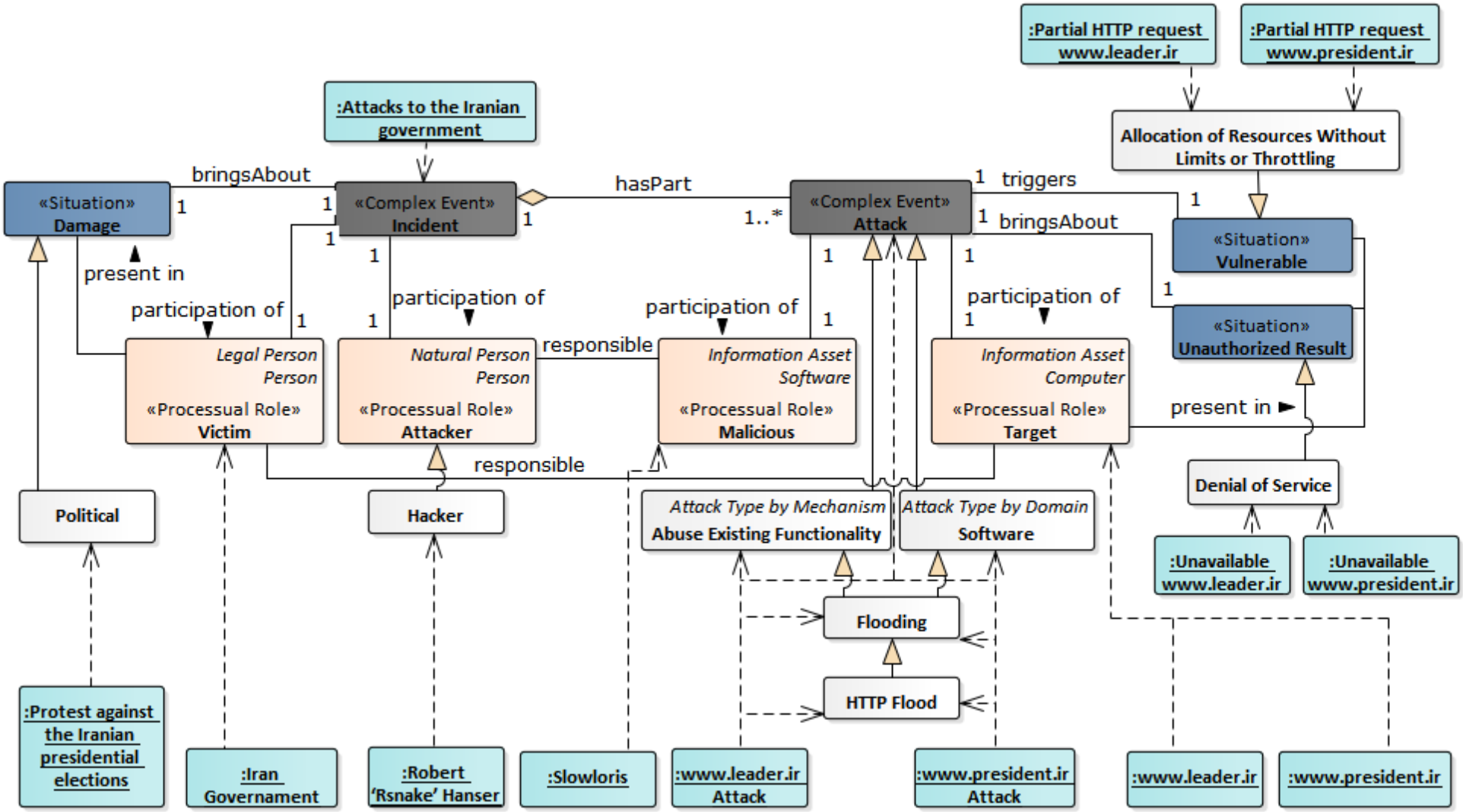



Figura 54 – Representação do Incidente do Irã usando sCuDO

Conforme ilustrado na Figura 54, os ataques usando sCuDO podem ser tipificados de formas distintas dependendo do critério utilizado para classificação. Sob o ponto de vista do mecanismo utilizado para atacar, os ataques foram classificados como ataques do tipo que abusam de funcionalidades existentes (*Abuse Existing Functionality*). Considerando o domínio de ataque, os ataques foram classificados como ataques que utilizam *software*. Analisando com mais detalhe os ataques, eles enviam muitas requisição ao alvo consumindo seus recursos, logo também podem ser classificados como do tipo *Flooding*. E, como para atacar o Slowloris usou solicitações HTTP, neste caso os ataque podem ser classificados como do tipo *HTTP Flood*.

Essa estrutura hierárquica de tipos e subtipos representada em sCuDO pode ajudar a evitar inconsistências e ambiguidades. Por exemplo, considerando que esses ataques, na época da ocorrência, estivessem sido registrados de acordo com sCuDO, como mostra a Figura 54. Além disso, suponha que outro incidente tenha ocorrido no Irã e fosse registrado utilizando sCuDO como do tipo *Flooding*. Com os dois incidentes registrados de acordo com o sCuDO, seria possível identificar que os dois incidentes têm características semelhantes, tais como, o abuso de funcionalidades existentes (*Abuse Existing Functionality*), o uso de software (*Software*) e o consumo de recursos do alvo (*Flooding*). Sendo assim, utilizando qualquer uma destas características os incidentes poderiam ser correlacionados.

Uma vez representado o incidente utilizando sCuDO as questões de competência definidas na Seção 5.1 podem ser respondidas, conforme Tabela 3.

Tabela 3 – Associação entre as questões de competência e as informações do Incidente do Irã representadas com sCuDO

| |
|---|
| QC1- Quais são as características de um evento para classificá-lo como incidente? |
| Os ataques ao Governo Iraniano (*Attacks to the Iranian government*) consiste de um evento com a participação do atacante (*Attacker*) Robert "Rsnake" Hanser e da vítima (*Victim*) Governo Iraniano (*Iranian government*), no qual o atacante causou dano (*Damage*) político (*Political*) a vítima em protesto as eleições presidenciais (*Protest against the Iranian elections*). |
| QC2- Qual foi o dano causado pelo incidente? |
| Dano (*Damage*) político (*Political*) em protesto as eleições presidenciais (*Protest against the Iranian elections*) |
| QC3- Quem causou o incidente? |
| O incidente foi causado pelo *hacker* Robert 'Rsnake' Hanser. |
| QC4- Quem sofreu o incidente? |
| A vítima do incidente foi o governo iraniano (*Iranian government*) |
| QC5- Quando ocorreu o incidente? |
| Não se aplica |

| QC6 – Qual foi a causa do incidente? |
|---|
| Os ataques aos *sites* www.leader.ir" e "www.president.ir" |
| QC7- Por que o incidente ocorreu? |
| O *hacker* Robert 'Rsnake' Hanser usou o *software* slowloris para enviar solicitações parciais HTTP (*Partial HTTP request*) para os *sites* 'www.leader.ir' e 'www.president.ir', abusando de funcionalidades existentes (*Abuse Existing Functionality*) neles de permitir a alocação de recursos sem limites (*Allocation of Resources Without Limits or Throttling*). Tal ação maliciosa levou a negação de serviço (*Denial of Service*) desses *sites*, tornando-os indisponíveis. |
| QC8 – Como o incidente ocorreu? |
| Quando os *sites* 'www.leader.ir' e 'www.president.ir' tiveram seus serviços negados (*Denial of Service*) e ficaram indisponíveis (*Unavailable* 'www.leader.ir', *Unavailable* 'www.president.ir'). |
| QC9 – Como o incidente foi classificado? |
| Considerando o mecanismo de ataque (*Mechanism of Attack*) o incidente é do tipo abuso de funcionalidade existente (*Abuse Existing Functionality*) e ao considerar o domínio de ataque (*Domain of Attack*) ele é do tipo de *software*. E, devido o *software* slowloris abusar da funcionalidades existentes nos *sites* inundando-os com requisições HTTP esse incidente também pode ser classificado como do subtipo *Flooding* ou ainda mais especificamente como *HTTP Flood*. |

Conforme demonstrado acima, sCuDO tem abrangência para responder a todas as questões de competência. A única questão que não foi respondida foi a QC5 por falta de informação, porém no modelo há entidade para representação.

## 6.2 Representação do Incidente WannaCry

As diferentes formas de categorizar incidentes representam um dos principais desafios para troca e compartilhamento de informações no domínio de Incidente de Segurança da Informação. Aplicando a metodologia DEFESA foi criado sCuDO para representar a hierarquia de tipos das entidades no intuito de dirimir ambiguidades. A estrutura de hierarquia de tipos de sCuDO nos quais os tipos mais específicos formam uma partição de um tipo mais geral, distinguindo instâncias de acordo com um critério de classificação específico.

Para demonstrar o uso prático da hierarquia de tipos criadas em sCuDO, foi escolhido *ransomware* WannaCry que causou um surto de incidentes em maio de 2017 afetando tanto pessoas quanto organizações governamentais, hospitais, universidades, empresas ferroviárias, firmas de tecnologia e operadoras de telecomunicações em mais de

150 países (63). Os incidentes causados pelo WannaCry se caracterizam por um conjunto de ataques associados ao um conjunto de vulnerabilidades que foram modeladas usando sCuDO.

O WannaCry foi criado por um grupo intitulado como Grupo Lazarus da Coréia do Norte (64). Ele é um *ransomware* que tem um módulo *worm* que ao ser executado instala um módulo no ativo de informação infectado que criptografa os seus arquivos e solicita o pagamento de um resgate em troca da chave de decriptografia.

Para causar esse dano financeiro à vítima, o incidente geralmente se inicia com um e-mail enviado pelo atacante para vítima contendo o *worm*, na expectativa da vítima, ao recebê-lo, o baixar e o executar. O *worm* explora a vulnerabilidade *EternalBlue* do ativo de informação alvo da vítima, esta vulnerabilidade é um tipo de validação de entrada imprópria (*Improper Input Validation*) devido a habilitação do serviço SMBv1 do Windows (1- *Windows SBMv1 enabled*) que possibilita a execução do módulo worm (2- *Worm module execution*).

Ao executar esse código alternativo (*Leverage Alternate Encoding*) caracteriza-se a ocorrência do primeiro ataque (65). Esse código se aproveita da possibilidade de acessar diretamente o endereçamento de memória (3- *Allow direct addressing of memory locations*) para exceder números inteiros suportada pelo *buffer* (*Integer Overflow to Buffer Overflow*) do alvo através da execução de códigos arbitrários (4- *Worm runs arbitrary code on systems*) para forçar estouro de número inteiro (*Forced Integer Overflow*). Essa restrição imprópria de operações dentro dos limites de um *buffer* de memória (*Improper Restriction of Operations within the Bounds of a Memory Buffer*) faz com que *worm* aloque mais memória do que o ativo de informação tem disponível (5 - *Less memory to be allocated tham expected*), causando estouro de buffer (6 - *Worm causes buffer overflow*).

A falta de gerenciamento apropriado de privilégios (7 - *Not properly manage privileges*) possibilita que o *worm* se aproprie de um encadeamento de privilégios de um processo do sistema (*Hijacking a Privileged Thread of Execution*) e ganhe privilégios (8- *Worm module gains privileges*). Embora a probabilidade de exploração de um segmento de execução privilegiado seja baixa, sua gravidade é crítica e resulta na execução de comandos não autorizados, obtendo privilégios ou assumindo identidade (66).

O ativo de informação alvo permitiu o controle externo de configurações do sistema (9 - *Allow external control of system settings*), possibilitando que o *worm* obtivesse o controle da configuração (10 - *Worm controls the configuration*). Além disso, o alvo também permitiu que fosse incluída uma funcionalidade executável (11 - Allow include executable functionality) que viabilizou o registro do *worm* com um serviço e instalação do módulo *ransomware* (12 - *Worm registers as a service and installs ransomware*) como um arquivo de sistema. O módulo *worm*, em seguida, baixa uma ferramenta para conexão com uma rede anônima do atacante e se conecta a alguns domínios dessa rede. Estes domínios são

utilizados apenas para rastrear as infecções e prover um endereço de pagamento *bitcoin* único, bem como as chaves de decriptografia, caso a vítima pague pelo resgate (67).

O módulo *ransomware*, ao ser executado, instala-se como um serviço que é executado sempre que o computador for reiniciado. Em virtude do alvo atacado permitir acesso as principais pastas e arquivos (13 - *Allow access to major folders and files*), o *ransomware* acessa os nomes das pastas e de serviços da máquina atacada e utilizam uma sequencia aleatória de caracteres para criptografá-los e exibir uma mensagem de solicitação de pedido de pagamento de resgate em troca de chave de decriptografia (14 - *Ransomware encrypts files and folders*). A Figura 55 mostra os ataques e as situações vulneráveis envolvidas no Incidente WannaCry (67).

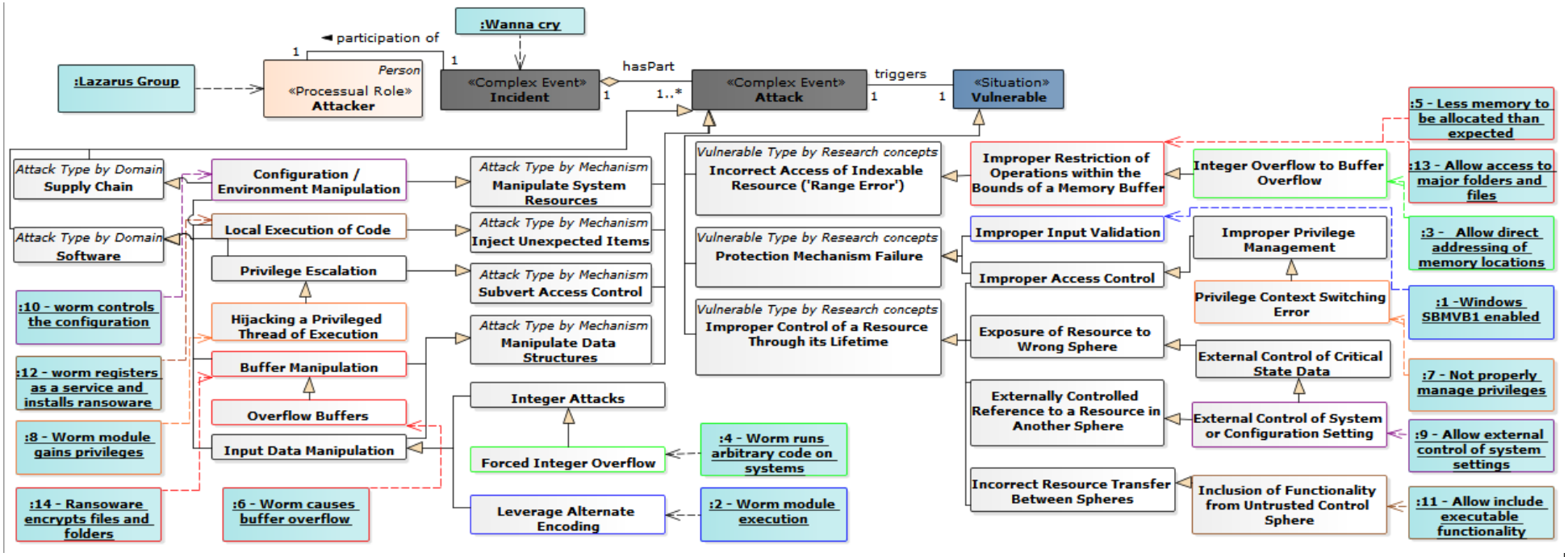



Figura 55 – Representação do Incidente WannaCry usando sCuDO

Conforme ilustrado na Figura 55, o incidente ocorreu em virtude de várias situações vulneráveis do ativo de informação alvo que viabilizaram a ocorrência de uma sequência de ataque. Como a figura tem muitos elementos sendo representados, a associação entre situação vulnerável que possibilita a ocorrência do ataque foi representada através de numeração das instâncias. As instâncias, retângulo de cor verde, numeradas com número ímpar representam situações vulneráveis e o respectivo ataque que ocorreu em virtude da exploração dessa situação vulnerável está numerado com o número par subsequente. Por exemplo, a situação vulnerável de deixar o serviço SMBv1 no Windows habilitado (1 - *Windows SBMVB1 enabled*) foi atribuída o número 1 e, ela viabilizou a ocorrência do primeiro ataque que foi a execução do módulo *worm* (2 - *Worm module execution*) que foi atribuído o número 2 e, assim, sucessivamente.

Além da relação entre situação vulnerável e o ataque, os seus tipos e subtipos estão representados na Figura 55. Os ataques são tipificados por domínio de ataque (*Domain of Attack*) e mecanismo de ataque (*Mechanism of Attack*) e as situações vulneráveis são tipificadas de acordo com os conceitos de pesquisa (*Resarch concepts*). Usando essa representação foi possível responder algumas questões de competência, conforme mostrado no Tabela 4.

Tabela 4 – Associação entre as questões de competência e as informações do Incidente Wannacry representadas com sCuDO

| |
|---|
| QC1- Quais são as características de um evento para classificá-lo como incidente? |
| O Incidente Wannacry consiste de uma série de ataques causados pelo atacante Grupo Lazarus da Coréia do Norte (*Lazarus Group*) causando danos financeiros à vítima. |
| QC2- Qual foi o dano causado pelo incidente? |
| A série de ataques resultam na criptografa os arquivos do ativo de informação alvo e na solicitação de pagamento de um resgate em troca da chave de decriptografia (14 - *Ransomware encrypts files and folders*), causando dano financeiro à vítima. |
| QC3- Quem causou o incidente? |
| O incidente foi causado pelo Grupo Lazarus da Coréia do Norte (*Lazarus Group*). |
| QC4- Quem sofreu o incidente? |
| Houveram inúmeras ocorrências desse mesmo tipo de incidente, afetando tanto pessoas quanto organizações governamentais, hospitais, universidades, empresas ferroviárias, firmas de tecnologia e operadoras de telecomunicações em mais de 150 países. Como o enfoque era representar os tipos de ataque e de situação vulnerável envolvidos neste tipo de incidente, nenhuma vítima específica foi representada na Figura 55. |
| QC5- Quando ocorreu o incidente? |

| |
|---|
| O surto desse tipo de incidente foi em maio de 2017. Em virtude de não ter sido representada uma ocorrência específica não foi representado o momento do incidente. |
| QC6 – Qual foi a causa do incidente? |
| O incidente foi causado devido aos resultados não autorizados dos seguintes ataques: execução do módulo *worm* (2 - *Worm module execution*), *worm* executa códigos arbitrários nos sistemas (4- *Worm runs arbitrary code on systems*), *worm* causa estouro de buffer (6 - *Worm causes buffer overflow*), *worm* ganha privilégios (8- *Worm module gains privileges*), *worm* controla a configuração (10 - *Worm controls the configuration*), *worm* se registra como um serviço e instala módulo *ransomware* (12 - *Worm registers as a service and installs ransomware*) e o *ransomware* criptografa arquivos e pastas (14 - *Ransomware encrypts files and folders*). |
| QC7- Por que o incidente ocorreu? |
| Porque o ativo de informação alvo apresentava as seguintes situações vulneráveis: serviço SMBv1 do Windows habilitado (1- *Windows SBMv1 enabled*), permitia acesso direto ao endereçamento de memória (3- *Allow direct addressing of memory locations*), permitia a alocação de mais memória do que o ativo de informação tinha disponível (5 - *Less memory to be allocated tham expected*), não havia gerenciamento apropriado de privilégios (7 - *Not properly manage privileges*), permitia o controle externo de configurações do sistema (9 - *Allow external control of system settings*), permitia que fosse incluída funcionalidade executável (11 - Allow include executable functionality) e permitia o acesso as principais pastas e arquivos (13 - *Allow access to major folders and files*). |
| QC8 – Como o incidente ocorreu? |
| Os resultados não autorizados não foram representados no diagrama. |
| QC9 – Como o incidente foi classificado? |
| O incidente envolve os seguintes tipos de ataque: aproveitamento de código alternativo (*Leverage Alternate Encoding*), forçar o estouro de número inteiro (*Forced Integer Overflow*), forçar o estouro do buffer (*Forced Integer Overflow Buffers*), apropriação de um segmento de execução privilegiado (*Hijacking a privileged thread of execution*), manipulação de configuração ou ambiente (*Configuration / Environment Manipulation*), execução de código local (*Local Execution of Code*) e manipulação de buffer (*Buffer Manipulation*). Em virtude da variedade de tipos de ataque envolvidos e da grande quantidade de ocorrências, o incidente compostos por esses tipos de ataque são apelidados de Wannacry. |

## 6.3 Representação da base de incidentes de um CSIRT

Há várias organizações que utilizam sistema de detecção de intrusão (IDS) para auxiliar na segurança. Os arquivos de *logs* produzidos por essas ferramentas são boas fontes de dados para análise de incidentes ocorridos e, tais informações, podem ser representadas utilizando sCuDO.

Nesse sentido, foram associados os campos da base de incidentes de um CSIRT as entidades de sCuDO. A base de incidentes do CSIRT contém uma massa de dados de 5 meses de ocorrências de incidentes extraídas do *log* do sistema de detecção de intrusão Network Security Manager da McAfee. Nela, cada incidente é composto por ataques do mesmo tipo, realizados por um mesmo atacante usando o mesmo ativo de informação malicioso e atingindo o mesmo ativo de informação da vítima. Cada registro contém informação de um incidente através dos seguintes campos:

- Time
- Attacker IP Address
- Attacker Port
- Target IP Address
- Target Port
- Domain
- Device
- Interface
- Name
- Attack Count
- Direction
- Result
- Severity
- BTP
- Attack Category
- Attacker Risk
- Target Risk

- Application

O campo *Time* contém a data/hora do registro da ocorrência e foi usado para representar o momento do incidente (*Incident Moment*). Os campos (*Attacker IP Address*, *Attacker Port*) contém informações do ativo de informação (*Information Asset*) malicioso (*Malicious*) usado pelo atacante no incidente para atacar. Os campos (*Target IP Address*, *Target Port*) contém informações do ativo de informação (*Information Asset*) alvo (*Target*) do ataque. Os campos *Domain*, *Device* e *Interface* representam as informações de vítima (*Victim*) do incidente. O campo *Name* tem informação do nome do tipo dos ataques (*Attacks*) do incidente. E, o campo *Attack Count* tem informação da quantidade de ataques. A Figura 56 ilustra essa correlação entre os campos da base de incidentes do CSIRT e as entidades de sCuDO.

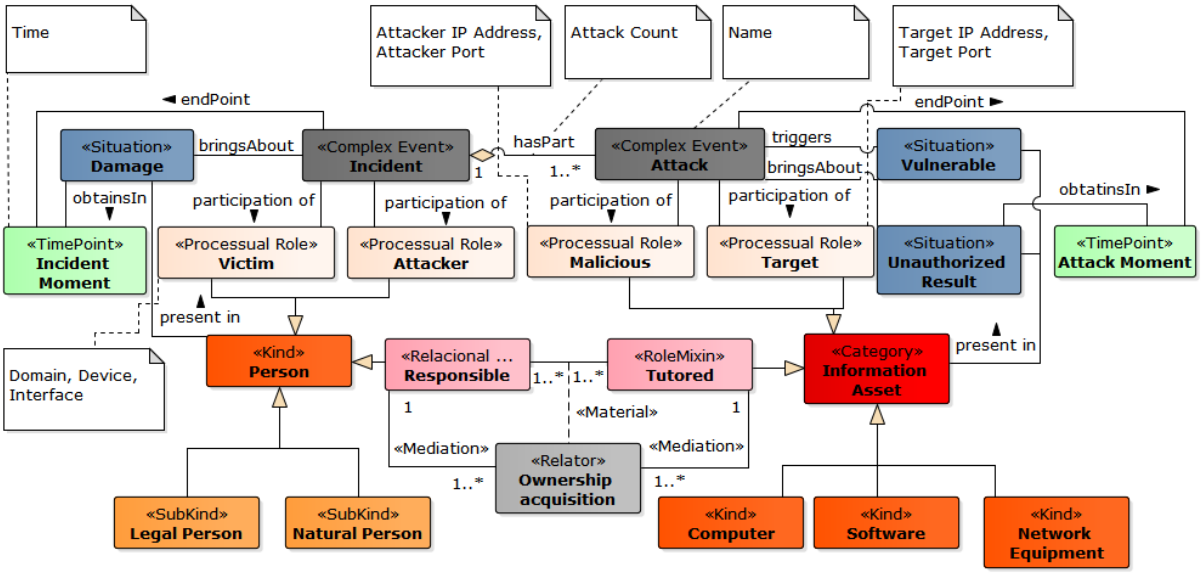



Figura 56 – Correlação entre os campos da base de incidentes do CSIRT e as entidades de sCuDO

Essa associação, ilustrada na Figura 56, foi usada no Capítulo 8 para auxiliar no desenvolvimento do ambiente analítico.

# 7 APLICAÇÃO DA METODOLOGIA DEFESA PARA CONSTRUIR SCUD$^2$O

sCuDO foi criado para facilitar a troca e o compartilhamento de informações sobre Incidentes de Segurança de Informação porém, a análise e comparação de milhares de ocorrências necessita de uma ferramenta apropriada. Para redução da quantidade de incidentes, o ideal seria que os dados de incidentes oriundos de logs, IDS e relatórios fossem reunidos de forma organizada e adequada para que eles sejam facilmente consultados e medidos. O DW tem essas características, por isso, foi o ambiente escolhido para dar suporte à tomada de decisão utilizando grande massa de dados históricas. Ele foi construindo de acordo com o macroprocesso *Construir sistema de apoio à decisão* da metodologia DEFESA.

O primeiro processo a ser realizado ao construir um sistema de apoio à decisão, consiste elaborar um modelo dimensional com expressividade semântica. Sendo assim, este capítulo apresenta sCuD$^2$O, o modelo dimensional de Incidentes de Segurança da Informação. Para o desenvolvimento desse modelo foram utilizadas as atividades do processo *Elaborar modelo dimensional com expressividade semântica* da metodologia DEFESA, conforme ilustrado na Figura 33.

Vale ressaltar ainda que nesse capítulo, com base nos trabalhos Moreira et al.(47) e Amaral e Guizzardi(48), foram definidas regras de transformação das entidades ontológicas tipificadas usando UFO-MLT para conceitos dimensionais, sendo esta também uma das principais contribuições desse trabalho.

## 7.1 Definir o propósito da análise

De acordo com a metodologia DEFESA, *Definir o propósito da análise* consiste na primeira tarefa para elaborar modelo dimensional com expressividade semântica, conforme ilustrado na Figura 33. Com base em sCuDO, foram idealizadas formas de analisar dados de ocorrências de Incidentes de Segurança da Informação em conjunto para identificar tendências e padrões, originando as questões analíticas para embasar a construção do modelo dimensional [1].

**Questões analíticas:**

[1] Inicialmente, ao conceber sCuDO, foram identificadas questões de competência e, ao definir o propósito da análise, foram identificadas novas questões, as questões analíticas. No entanto, como sCuDO é a base de tudo, ele também pode ser dito competente para responder as questões analíticas. Desse ponto de vista, pode-se dizer que as questões analíticas estão incluídas nas questões de competência que sCuDO atende.

QA1 - Quantos incidentes ocorreram?

QA2 - Quantos ataques ocorreram?

QA3 - Quantos ataques levaram a ocorrência de cada incidente?

QA4- Qual o tipo de ataque que ocasionou mais incidentes?

QA5- Qual foi a situação vulnerável mais frequente?

QA6- Qual o resultado não autorizado mais frequente?

QA7- Qual o dano mais frequente?

QA8- Qual o ativo de informação alvo mais atacado?

QA9- Qual o ativo de informação malicioso que mais atacou?

QA10- Qual o atacante que mais causou incidentes?

QA11- Qual a vítima que mais sofreu incidentes?

QA12- O atacante sempre ataca a mesma vítima?

QA13- O atacante sempre comete o mesmo tipo de ataque?

QA14- O alvo sempre sofre o mesmo tipo de ataque?

QA15- A vítima sempre sofre o mesmo tipo de ataque?

QA16- O alvo é atacado em virtude da mesma situação vulnerável?

QA17- A vítima é atacada em virtude da mesma situação vulnerável?

Para responder a essas perguntas foi criado sCuD²O, o modelo dimensional de Incidentes de Segurança da Informação. As entidades de sCuDO foram transformadas em elementos do MD para alcançar os requisitos analíticos necessários.

## 7.2 Identificar os conceitos dimensionais

Esta abordagem define como mapear conceitos de nível superior do domínio, representados utilizando categorias UFO-MLT, para cada conceito de modelo dimensional (MD), representado como fato (*fact*), medida (*measure*), dimensão (*dimension*) ou hierarquia (*hierarchy*). Para tal, foram definidas regras para derivar os elementos do MD a partir dos elementos da ontologia de domínio. E, usando essas regras, paulatinamente sCuDO será transformado em sCuD²O, o modelo dimensional de Incidentes de Segurança da Informação. A representação das regras será feita usando lógica de primeira ordem e os elementos da UFO serão identificados com prefixo u, os elementos da MLT com prefixo m e os elementos do MD com o prefixo d.

A transformação foi iniciada com os eventos. De acordo com Amaral e Guizzardi(48),

fatos são eventos observáveis e, a UFO prevê representação tanto para eventos atômicos como para eventos complexos. Previamente Moreira et al.(47) haviam proposto que eventos complexos fossem representados como fatos. Esse regra foi estendida para todos os tipos de evento (*Event*), o evento atômico (*AtomicEvent*) ou evento complexo (*ComplexEvent*), seja representado como um fato (*fact*) no MD, conforme Regra 1.

*Regra 1* : $\forall e : uEvent \ (uComplexEvent(e) \vee uAtomicEvent(e) \rightarrow dfact(e))$;

sCuDO tem os eventos complexos (*Complex Events*) incidente (*Incident*) e ataque (*Attack*), usando a regra acima sCuD²O tem o fato incidente (*Incident fact*) e o fato ataque (*Attack fact*) conforme ilustrado na Figura 57.

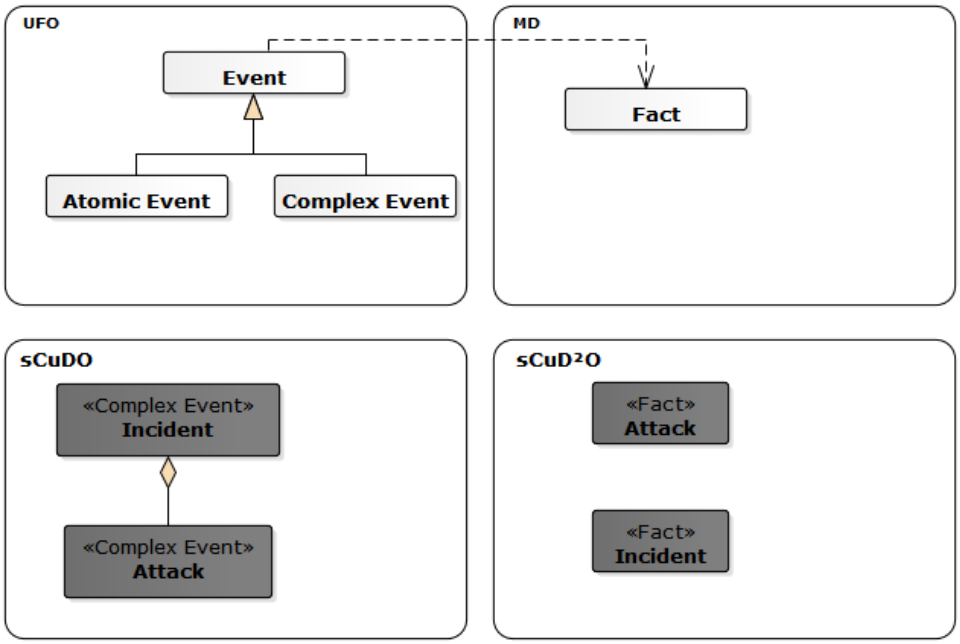



Figura 57 – Transformação de *Event* em *Fact*

Um evento (*Event*) é a manifestação de (*manifestedBy*) disposições (*Disposition*) de objetos (*Objects*) (ver Regra 7 e 8 da Seção 2.1.1). Desta forma, o evento (*Event*) depende exclusivamente deste (*exclusivelyDependsOn*) objeto (*Object*) (ver Regra 11 e 12 da Seção 2.1.1) e o objeto (*Object*) participa do (*participationOf*) evento (*Event*) (Regra 13 e 14 da Seção 2.1.1). Portanto, as participações dos objetos em eventos representados na ontologia de domínio podem ser vistas como possíveis perspectivas de análise para o fatos, ou seja, dimensões (47). Então, todo objeto (*Object*) que participa de (*participationOf*) um evento, representado como fato (*fact*) na MD, é transformado em uma dimensão denominada de *participationObjectDimension*.

*Regra 2* : $\forall o : uObject, \ p : uParticipation, \ e : uEvent \ (uparticipationOf(p, o)$

$\wedge \ hasPart(e, p) \wedge dfact(e) \rightarrow dparticipationObjectDimension(o, e))$;

O Incidente de Segurança da Informação ocorre quando uma pessoa ataca uma

vítima até que ocorra um dano. Sendo assim, o incidente tem a participação de pessoas e, essas pessoas têm ativos de informação sob sua responsabilidade. O atacante usa o ativo de informação sob sua responsabilidade de forma maliciosa para atacar o ativo de informação da vítima, então podemos dizer que o evento ataque (*Attack*) tem a participação de (*participationOf*) ativos de informação (*Information Asset*). Sabendo que o evento complexo incidente de sCuDO está sendo representado como fato (*fact*) em sCuD²O, então o objeto (*Object*), participante do (*participationOf*) do evento (*Event*) incidente (*Incident*), pessoa (*Person*) foi transformado em uma dimensão de objeto participante (*participationObjectDimension*), conforme ilustrado na Figura 58. De forma semelhante, o evento complexo (*Complex Event*) ataque (*Attack*) tem a participação do (*participationof*) ativo de informação (*Information Asset*) que também foi transformado em dimensão de objeto participante (*participationObjectDimension*).

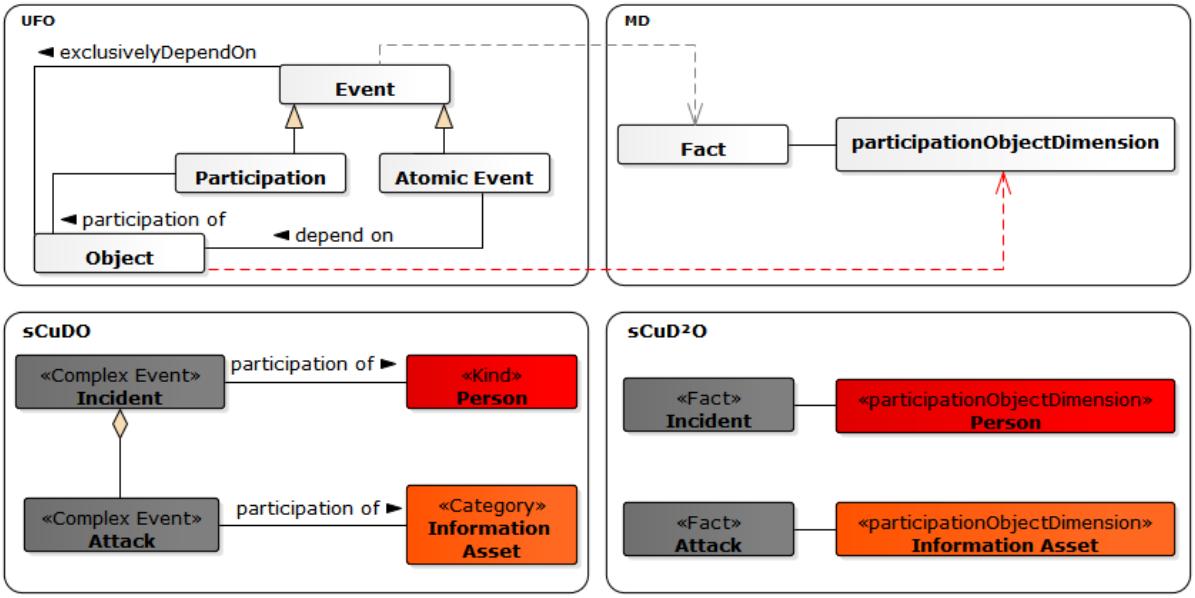



Figura 58 – Transformação de *Participation* em *Dimension*

O objeto participa de um evento desempenhado um papel processual (*processual role*). Um mesmo tipo de objeto pode ter mais de uma participação em um evento e, em cada uma dessas participações o objeto desempenha papéis distintos. Cada um desses papéis devem ser representados explicitamente em sCuD²O. Em sCuD²O os objetos que participam do evento são representados como dimensão de objeto participante (*participationObjectDimension*) e, em MD, quando uma dimensão se relaciona várias vezes com o fato (*fact*), cada um de seus papéis desempenhados dever ser apresentado como uma exibição rotulada separadamente. Tal representação, chama-se *role-playing dimension*. Diante disso, cada objeto (*Object*) que desempenha mais de um papel processual (*processual role*) em sCuDO e que são representados como *participationObjectDimension* em sCuD²O tiveram cada papel processual (*processual role*) seu transformado em *rolePlayingDimension* em sCuD²O, conforme Regra 3.

$\underline{Regra\ 3}: \ \forall o : uObject,\ r : uProcessualRole, f : dfact\ (uplays(o,r) \wedge dparticipationObjectDimension(o,f) \rightarrow drolePlayingDimension(r,o))$

Em sCuDO, o evento incidente tem a participação de dois objetos do mesmo tipo, pessoa. Em uma das participações do objeto pessoa (*Person*) desempenha o papel processual (*processual role*) de atacante (*Attacker*) e na outra participação o objeto pessoa (*Person*) desempenha o papel processual (*processual role*) de vítima (*Victim*). Aplicando a Regra 3, o fato (*fact*) incidente (*Incident*) tem a *participationObjectDimension* pessoa (*Person*) que desempenha o papel (*rolePlayingDimension*) de atacante (*Attacker*) e de vítima (*Victim*).

O evento ataque tem a participação do objeto ativo de informação (*Information Asset*) desempenhando dois papéis processuais (*processual role*), de malicioso (*Malicious*) e de alvo (*Target*). Ao aplicar a Regra 3, o fato (*fact*) ataque (*Attack*) tem a *participationObjectDimension* ativo de informação (*Information Asset*) que desempenha o papel (*rolePlayingDimension*) de malicioso (*Malicious*) e de alvo (*Target*). Detalhes das transformações da representação de cada *processual role* dos objetos em *rolePlayingDimension* das dimensões encontra-se na Figura 59.

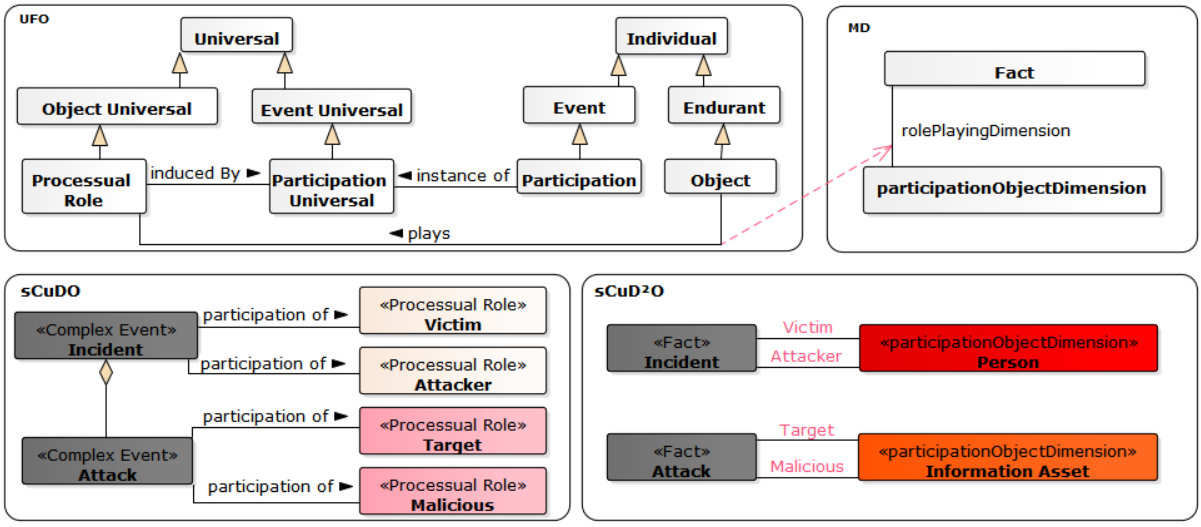



Figura 59 – Representação de *Role* no modelo dimensional

Os eventos são transformações de uma parte da realidade para outra, ou seja, eles podem mudar a realidade alterando o estado de coisas de uma situação para outra (34). Todo evento (*Event*) é acionado (*triggers*) por uma situação (*Situation*) (Regra 17 da Seção da 2.1.1) e cada evento (*Event*) desencadeia (*bringsAbout*) uma situação (*Situation*) (Regra 18 da Seção da 2.1.1). As situações ajudam a fornecer o contexto o qual o evento ocorre, pois, revelam as consequências do evento. E, as situações causadas pelo evento e as condições favoráveis para que o evento ocorra, viabilizam a ocorrência do evento. Esse contexto fornecido pelas situações em sCuDO pode ser transformado em dimensões em sCuD²O,

de forma que a situação (*Situation*) desencadeada (*bringsAbout*) pelo evento (*Event*) seja transformada em uma dimensão de pós-estado (*posStateDimension*). E, situação (*Situation*) que representa a condição necessária para que o evento ocorra (*triggers*) seja transformada em uma dimensão de pré-estado (*preStateDimension*), conforme Regra 4 e 5 respectivamente.

$\underline{Regra\ 4} : \forall e : uEvent,\ \ s : uSituation\ \ (ubringsAbout(e, s) \wedge dfact(e) \rightarrow$

dposStateDimension(s, e))

$\underline{Regra\ 5} : \forall e : uEvent, s : uSituation\ \ (utriggers(s, e) \wedge dfact(e) \rightarrow$

dpreStateDimension(s, e))

O evento incidente causa dano à vítima, então a situação (*Situation*) de dano (*Damage*) de sCuDO foi transformada em uma dimensão de pós-estado (*posStateDimension*) do fato (*fact*) incidente (*Incident*) no sCuD²O. O atacante, para causar o dano, usa um ativo de informação para explorar alguma vulnerabilidade do ativo de informação da vítima para que o ativo de informação da vítima passe a apresentar um resultado não autorizado. Para que um ataque ocorra, o ativo de informação da vítima necessita estar em uma situação vulnerável, logo, a situação (*Situation*) vulnerável (*Vulnerable*) de sCuDO foi transformada na dimensão de pré-estado (*preStateDimension*) do fato (*fact*) ataque (*Attack*). Em virtude do resultado não autorizado (*Unauthorized result*) ser a situação (*Situation*) desencadeada (*bringsAbout*) pelo ataque (*Attack*), esta situação (*Situation*) foi transformada na dimensão de pós-estado (*posStateDimension*). Destalhes das transformações das situações em dimensões estão na Figura 60.

A situação (*Situation*) ocasionada (*bringsAbout*) por um evento (*Event*) ocorre no momento que o evento termina (*endPoint*) (Regra 19 da Seção da 2.1.1). Neste caso, o momento da ocorrência da situação que foi causada pelo evento (*bringsAbout*) coincide com o término do evento (*endPoint*) será representado como uma dimensão temporal (*temporalDimension*) em sCuD²O, conforme regra abaixo.

$\underline{Regra\ 6} : \forall e : uEvent, \exists t : uTimePoint\ \ s : uSituation,\ \ (uobtainsIn(s, t) \wedge$

$ubringsAbout(e, s) \wedge uendPoint(e, t) \wedge dfact(e) \rightarrow dtemporalDimension(t, e))$

Sendo assim, o momento do incidente (*Incident Moment*) que é quando a vítima sofre o dano no evento incidente (*Incident*) de sCuDO foi transformado em uma dimensão temporal (*temporalDimension*) em sCuD²O. E, o momento do ataque (*Attack Moment*), modelado em sCuDO, que o ativo de informação alvo do evento ataque passar a apresentar resultado não autorizado também foi transformado em uma dimensão temporal (*temporalDimension*) em sCuD²O, conforme ilustrado na Figura 60.

Objetos se relacionam com outros objetos mediante (*mediates*) um relacionador (*Relator*) (Regra 1 da Seção 2.1.1). E, nesse relacionamento os objetos desempenham

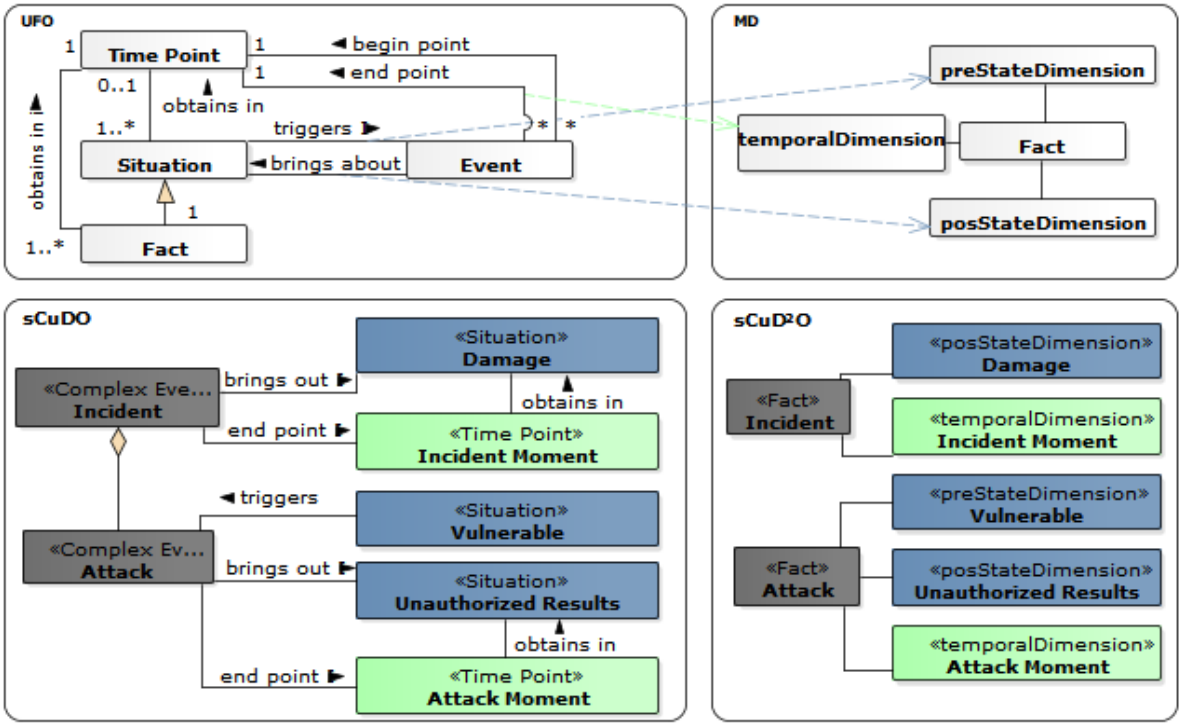



Figura 60 – Transformação de *Situation* em *Dimension*

papéis. Um papel relacional é uma propriedade acidental do objeto manifestada quando um objeto se relaciona com outro. Então, uma descrição completa do objeto e de suas propriedades envolve representar os outros objetos aos quais ele se relaciona.

Em sCuDO há objetos que se relacionam com outros que em sCuD²O são dimensões. Na modelagem dimensional as dimensões podem conter referências para outras dimensões. Uma das formas de modelar este relacionamento é através da tabela de fato. Sendo assim, o objeto (*Object*) que está relacionado com um outro objeto, representado em sCuD²O como uma dimensão de objeto participante (*participationObjectDimension*), o objeto relacionado, mesmo não sendo diretamente relacionado ao fato, também se torna uma dimensão denominada dimensão de objeto externamente dependente (*externallyDependentObjectDimension*) em sCuD²O para que a representação do relacionamento possa ser mantida, conforme Regra 7.

$\underline{Regra\ 7}: \forall x,\ y : uObject,\ r : uRelator\ (umediates(r,x) \wedge umediates(r,y) \wedge$
( ∃! f: dfact $dparticipationObjectDimension(x,f)) \rightarrow$
$dexternallyDependentObjectdimension(y,f))$

Em sCuDo cada uma das pessoas que participam do incidente, seja ela um atacante ou uma vítima, tem responsabilidade sobre um ativo de informação. De forma análoga, ativo de informação malicioso ou um ativo de informação alvo que participam do ataque, tem uma pessoa responsável por ele. A Figura 61 mostra como esses relacionamentos foram expressos em sCuD²O usando a regra de transformação acima.

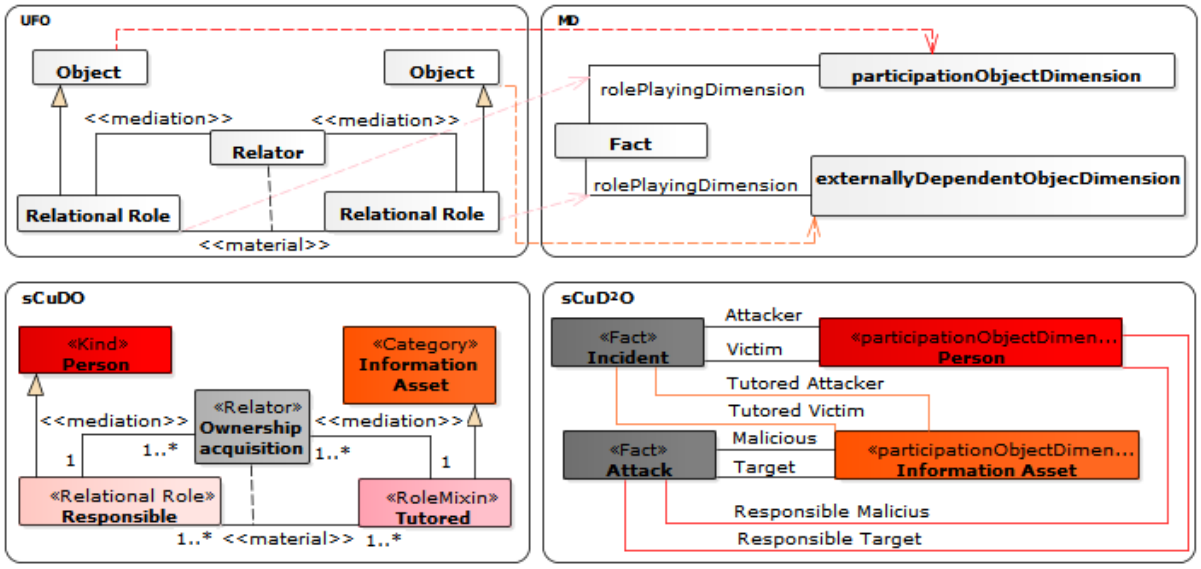



Figura 61 – Transformação de *Objects* em *Dimension*

As regras de transformação até então apresentadas sugeriram a criação de vários tipos de dimensão. Esses nomes foram atribuídos para dar maior expressividade ao sCuD²O porém, há um conjunto de regras que se aplicam a qualquer tipo de dimensão. Então, a partir de agora todos os indivíduos de sCuDO que foram transformados em algum tipo de dimensão no sCuD²O serão referenciados de forma simplificada como dimensão (*dimension*).

A primeira transformação que é válida para todos os tipos de dimensão está relacionada aos tipos de tipos do domínio. Esses tipos são utilizados para descrever entidades do domínio e torná-las mais explícitas. Em sCuDO, as entidades do domínio são representadas, utilizando a MLT, como próprias especializações de *Individual* e os seus respectivos tipos são representados como instâncias de ordem superior que categorizam as entidades do domínio formando uma hierarquia. Uma hierarquia, em modelagem dimensional, é uma estrutura que descreve um padrão transversal de uma dimensão (68). Desta forma, a entidade do domínio representada como uma especialização própria de *Individual* em sCuDO e como uma dimensão em sCuD²O terá o tipo (*Type*) que a categoriza (*categorizes*) representado como uma hierarquia (*hierarchy*) em sCuD²O, conforme Regra 8.

$\underline{Regra\ 8} : \forall t, t' : mType \quad (mcategorizes(t', t) \wedge ddimension(t) \rightarrow$

dhierarchy(t', t))

A entidade do domínio ataque tem tipos, porém em sCuD²O ela é representada como um fato. Para representar os tipos de ataque, uma nova regra foi criada para que a entidade do domínio representada como um fato em sCuD²O tenha seu tipo representado como uma dimensão.

$\underline{Regra\ 9} : \forall t, t' : mType\ (mcategorizes(t', t)\ \wedge\ dfact(t) \rightarrow$

ddimension(t', t))

Tipos podem ser especializado em outros tipos e suas instâncias são especializações próprias de entidades do domínio. Essa estrutura forma uma hierarquia de classificação contendo níveis que podem ser usados para a estruturação de dimensões em modelos dimensionais (48). Um nível de hierarquia contém um conjunto distinto de membros e níveis diferentes correspondem a granularidades de dados diferentes (68).

Regra 10: ∀ t, t':mType, (mcategorizes(t',t) ∧ dhierarchy(t) → dhierarchy(t', t))

Em sCuDO há várias entidades que são categorizadas em tipos. O atacante (*Attacker*), o ativo de informação (*Information Asset*), o dano (*Damage*) e o resultado não autorizado (*Unauthorized Results*) podem ser de vários tipos, como essas entidades são representadas em sCuD²O como dimensões (*Dimension*) então, os seus tipos são representados como hierarquia (*Hierarchy*). O ataque (Attack) e a vulnerabilidade (Vulnerable) tem tipos e seus tipos são especializados em subtipos que em sCuD²O são representados como hierarquias relacionadas. A Figura 62 ilustra sCuD²O completo.

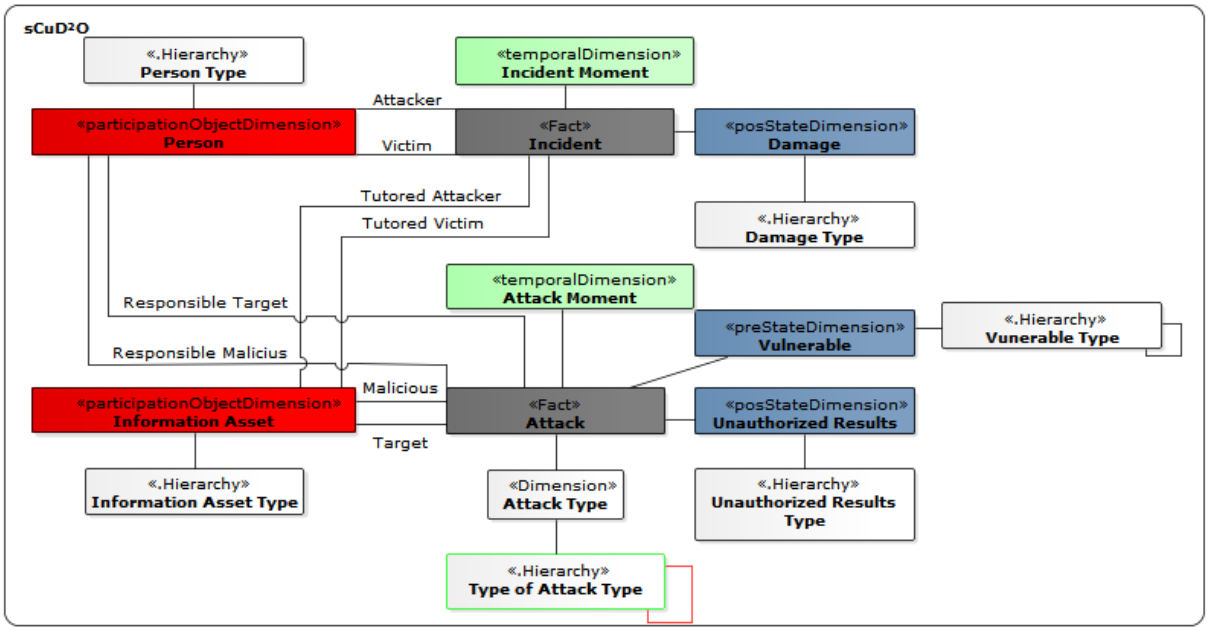



Figura 62 – sCuD²O - Modelo dimensional de Incidentes de Segurança da Informação

# 8 APLICAÇÃO DA METODOLOGIA DEFESA PARA DESENVOLVER AMBIENTE ANALÍTICO DE DADOS

Nos capítulos anteriores os processos da metodologia DEFESA foram aplicados baseados nas necessidades do negócio. Neste capítulo, os dados do negócio também foram considerados para implementação do sistema de apoio à decisão. Visando atingir este objetivo, o processo *Desenvolver ambiente analítico de dados* da metodologia DEFESA foi seguido, conforme ilustrado na Figura 35.

## 8.1 Produzir modelo lógico

Conforme apresentado no Capítulo 7, sCuD$^2$O foi definido levando em consideração as necessidades do negócio (ilustrado na Figura 62). Porém, o modelo lógico considera também os dados do negócio. Para desenvolver um ambiente analítico de dados foi utilizada a base de dados de incidentes de um Grupo de Resposta a Incidentes de Segurança em Computadores (*Computer Security Incident Response Team* - CSIRT), representada usando sCuDO na Seção 6.3, como principal fonte de dados para produzir o modelo lógico.

Os campos da base de incidentes do CSIRT foram associados a entidades usando sCuDO, conforme ilustrado na Figura 56. Nesta seção, os campos da base de incidentes do CSIRT foram usados para definir os atributos das dimensões, medidas e hierarquias dos conceitos dimensionais definidos em sCuD$^2$O.

O campo *Time* como representa o momento do incidente (Incident Moment) em sCuDO então em sCuD$^2$O representa a dimensão temporal momento do incidente (*«temporalDimension» Incident Moment*)(Regra 6 do Capítulo 7). Os campos *Attacker IP Address* e *Attacker Port* em sCuDO representam o ativo de informação malicioso do atacante e, em sCuD$^2$O representam a dimensão de objeto externamente dependente ativo de informação (*«externallyDependentObjectDimension» Information Asset*) desempenhado o papel de tutorado pelo atacante (*«role» Tutored Attacker*) (Regra 3 e 7 do Capítulo 7). De forma semelhante, os campos *Target IP Address* e *Target Port* representam a dimensão de objeto externamente dependente ativo de informação (*«externallyDependentObjectDimension» Information Asset*) desempenhando o papel de tutorado pela vítima (*«role» Tutored Victim*) (Regra 3 e 7 do Capítulo 7). E, os campos *Domain, Device* e *Interface* representam a dimensão objeto participante pessoa (*«participationObjectDimension» Person*) desempenhando o papel de vítima (*«role» Victim*) (Regra 2 do Capítulo 7).

O campo *Name* contém informação do nome do tipo dos ataques que levaram a ocorrência do incidente, logo ele foi usado para compor a dimensão tipo de ataque

(*«Dimension» Attack Type*) (Regra 9 do Capítulo 7). Conforme previamente discutido na Seção 5.4, o tipo de ataque categoriza o incidente e, em sCuDO, os tipos de ataque são tipificados em uma estrutura hierarquizada de acordo com a base de dados de enumeração e classificação comum de padrão de ataque (*Common Attack Pattern Enumeration and Classification* – CAPEC).

A base de dados do CAPEC contém inúmeros campos, porém no escopo desse trabalho somente foram utilizados os campos *CAPEC-ID*, *Name*, *RelatedAttackPatterns* para representar a hierarquia tipo de tipo de ataque (*«Hierarchy» Type of Attack Type*) (Regra 8 do Capítulo 7).

Na base de dados de incidentes do CSIRT não há informação sobre a situação vulnerável do ativo de informação alvo explorada pelo atacante, porém a correlação entre a base de dados do CAPEC e a base de dados *Common Weakness Enumeration* (CWE), representada em sCuDO (ilustrado na Figura 53), foi utilizada para associar o tipo de ataque que ocasionou o incidente aos possíveis tipos de situação vulnerável exploradas. Os campos *CWE-ID*, *Name*, *Related Weaknesses* e *Related Attack Patterns* da base de dados do CWE foram usados para representar a hierarquia tipo de vulnerável (*«Hierarchy» Type Vunerable*) (Regra 8 do Capítulo 7). O quadro 5 resume a relação entre as entidades de sCuD$^2$O e os campos das bases de dados.

Quadro 5 – Relação entre as entidades de sCuD$^2$O e os campos das bases de dados

| Conceito dimensional | Campo(s) | Fonte |
|---|---|---|
| «temporalDimension» Incident Moment | Time | CSIRT |
| «externallyDependentObjectDimension» Information Asset | Attacker IP Address, Attacker Port, Target IP Address, Target Port | CSIRT |
| «participationObjectDimension» Person | Domain, Device, Interface | CSIRT |
| «Dimension» Attack Type | Name | CSIRT |
| «Hierarchy» Type of Attack Type | CAPEC-ID, Name, RelatedAttackPatterns | CAPEC |
| «Hierarchy» Vulnerable Type | CWE-ID, Name, Related Attack Patterns, Related Weaknesses | CWE |

Conforme mostrado no Quadro 5, os dados disponíveis contém informações para popular *«temporalDimension» Incident Moment*, *«externallyDependentObjectDimension» Information Asset*, *«participationObjectDimension» Person*, *«Dimension» Attack Type*, *«Hierarchy» Type of Attack Type* e *«Hierarchy» Vulnerable Type*, conforme ilustrado na Figura 63.

Na base de dados de incidentes do CSIRT cada incidente é composto por ataques do mesmo tipo causados por um mesmo ativo de informação malicioso contra o mesmo

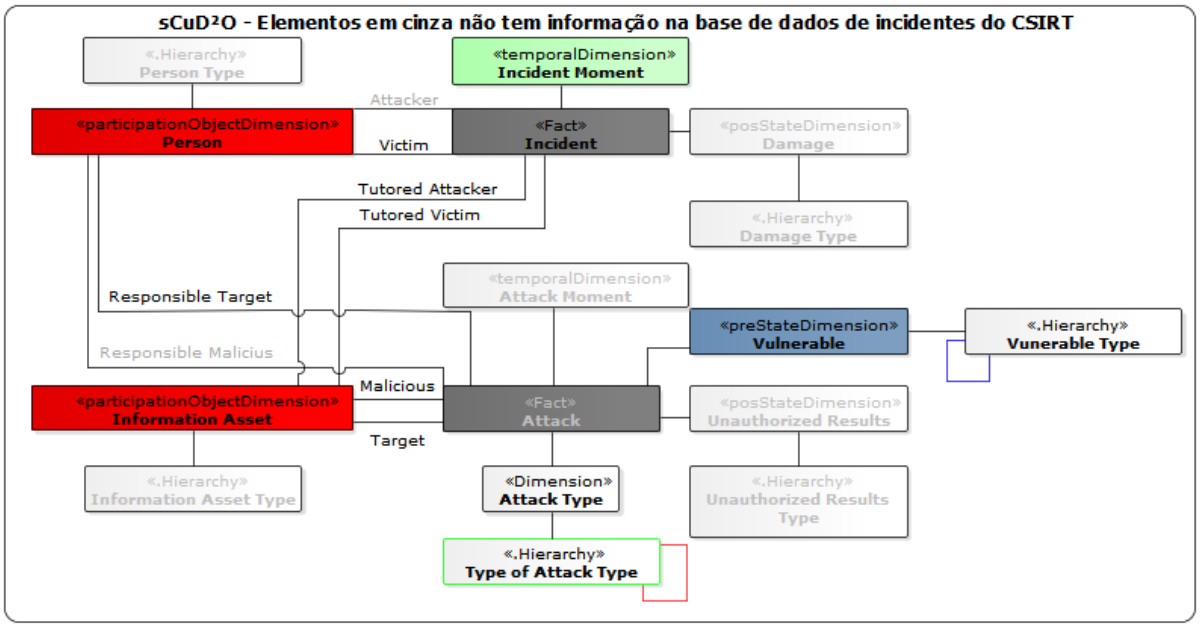



Figura 63 – Identificação dos conceitos dimensionais de sCuD²O com informação na base de dados de incidentes do CSIRT

ativo alvo e não há informação de quando cada um desses ataques ocorreram, da situação vulnerável em que o ativo de informação alvo estava e do resultado não autorizado causado por eles. Tais características denotam que esta base de dados está estruturada no grão do incidente, isto é, as informações estão agrupadas por incidente e, consequentemente, não há detalhes do ataque que viabilize analisá-lo como um fato. Sendo assim, este sistema de apoio à decisão somente analisa o fato incidente e a quantidade de ataques do incidente, disponibilizada através do campo *Attack Count*, foi utilizada como medida (*Measure*) do fato incidente.

Em virtude de cada incidente ser composto por ataques do mesmo tipo, a dimensão tipo de ataque (*« Dimension» Attack Type*), originalmente modelada em sCuD²O como dimensão do fato ataque, foi usada para obter informações sobre o tipo de ataque do incidente. Desta forma, neste sistema de apoio à decisão o fato incidente tem a dimensão tipo de ataque (*« Dimension» Attack Type*) e a relação com a hierarquia tipo de tipo de ataque (*«Hierarchy» Type of Attack Type*) foi mantida. E, a hierarquia tipo de vulnerável (*«Hierarchy» Type Vunerable*) foi relacionada à hierarquia de tipo de tipo de ataque (*«Hierarchy» Type of Attack Type*).

Dada a natureza dos dados da base de dados de incidentes do CSIRT, somente há informações para analisar o fato incidente pelo sistema de apoio à decisão e algumas adaptações foram feitas em sCuD²O para usar as informações disponíveis. Desta forma, a base de incidentes do CSIRT, a base de dados do CAPEC e a base de dados do CWE foram usadas para compor os atributos das dimensões, hierarquias e medidas, conforme

ilustrado na Figura 64.

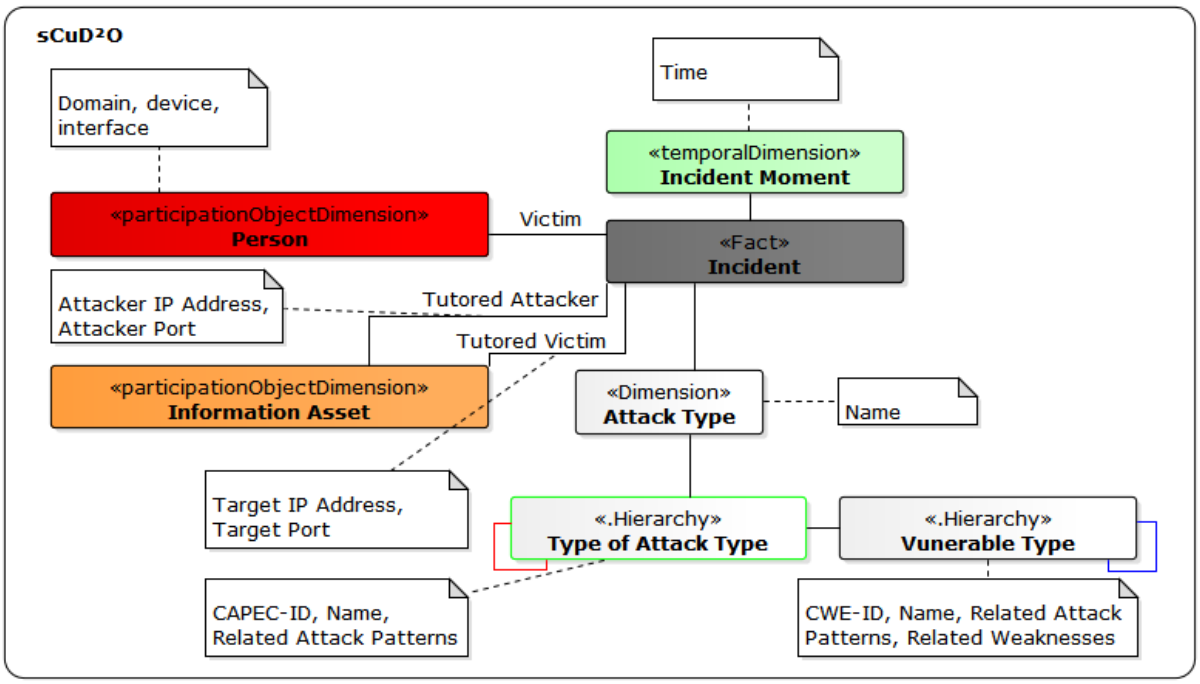



Figura 64 – sCuD$^2$O adaptado para as fontes de dados disponíveis

A partir dessas informações, o modelo lógico, apresentado na Figura 65, foi produzido contendo os dados da vítima através da dimensão pessoa que por questão de simplificação de nome para tabela será nomeada como *DimPerson*. Os dados do ativo de informação tutoreado pela vítima e pelo atacante ficam armazenados na dimensão ativo de informação (*DimInformationAsset*), a dimensão temporal (*DimTemporal*) tem informação do momento em que o incidente ocorreu e a dimensão tipo de ataque (*DimAttackType*) com informação do tipo dos ataques que ocasionaram o incidente.

O tipo de ataque da base de incidentes do CSIRT se relaciona com os tipos de ataque de sCuDO, isto é, com os tipos de ataque da base de dados do CAPEC. Essa relação é representada através de hierarquia tipo de tipo de ataque (*HTypeAttackType*). Os tipos de ataques podem ser especializado em outro tipo e este tipo em outro em no máximo três níveis tanto na categorização de tipo de ataque por mecanismo quanto por domínio. Para representar esses três níveis de especialização na hierarquia tipo de tipo de ataque foi criado o campo categoria (*category*) para representar o tipo mais genérico, o segundo nível de especialização será representado pelo campo meta padrão (*metaPattern*) e o terceiro nível pelo campo supertipo (*superType*) tanto para a estrutura de tipo de domínio de ataque quanto para estrutura de tipos de mecanismo de ataque.

Os tipos de ataque tem tipos de situação vulnerável associadas a eles, representadas através da hierarquia de tipo de vulnerável (*HVunerableType*). E, os tipos de vulnerável podem ser especializados em vários tipos tanto na categorização de tipo de vulnerável por

conceitos de pesquisa, de desenvolvimento e arquiteturais e, essas relações estão sendo representadas através das hierarquias supertipo de tipo de vulnerável por conceito de pesquisa (*HSuperVulnerableTypebyResarch*), supertipo de tipo de vulnerável por conceito de desenvolvimento (*HSuperVulnerableTypebyDevelopment*) e supertipo de tipo de vulnerável por conceito arquitetural (*HSuperVulnerableTypebyArchitectural*).

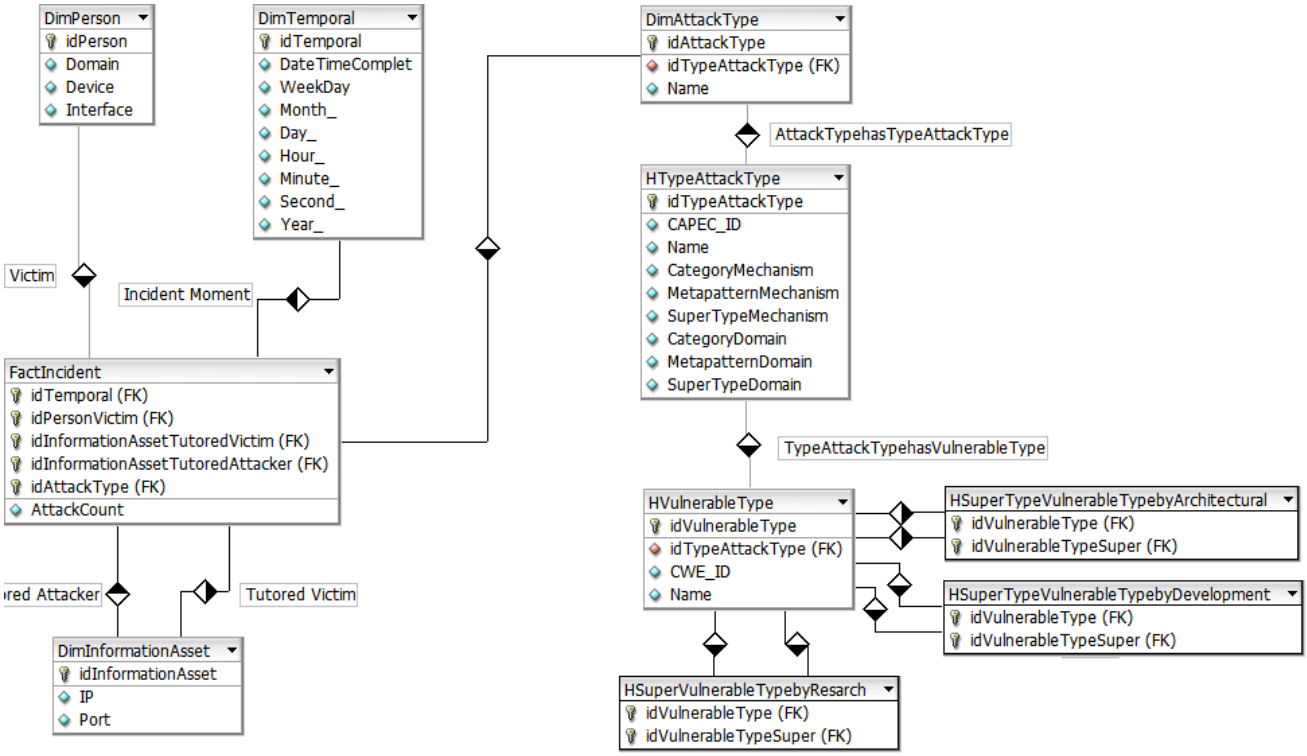



Figura 65 – Modelo lógico de Incidentes de Segurança da Informação

## 8.2 Extrair dados

Após definição do modelo lógico, os dados devem ser coletados e colocados em uma área intermediária de armazenamento para serem tratados antes de serem carregados no DW. Para tal, esta tarefa de *Extrair dados* pegará os dados coletados de suas fontes e os carregará em um banco de dados no PostgreSQL com o auxílio da ferramenta de integração de dados Pentaho Data Integration.

A base de dados de incidentes do CSIRT foi disponibilizada através dos seguintes arquivos: janeiro.csv, fevereiro.csv, marco.csv, abril.csv e maio.csv. Os dados desses arquivos serão reunidos na tabela *Incident*, conforme ilustrado na Figura 66 [1].

[1] O comando SQL para criação da tabela intermediária *Incident* encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

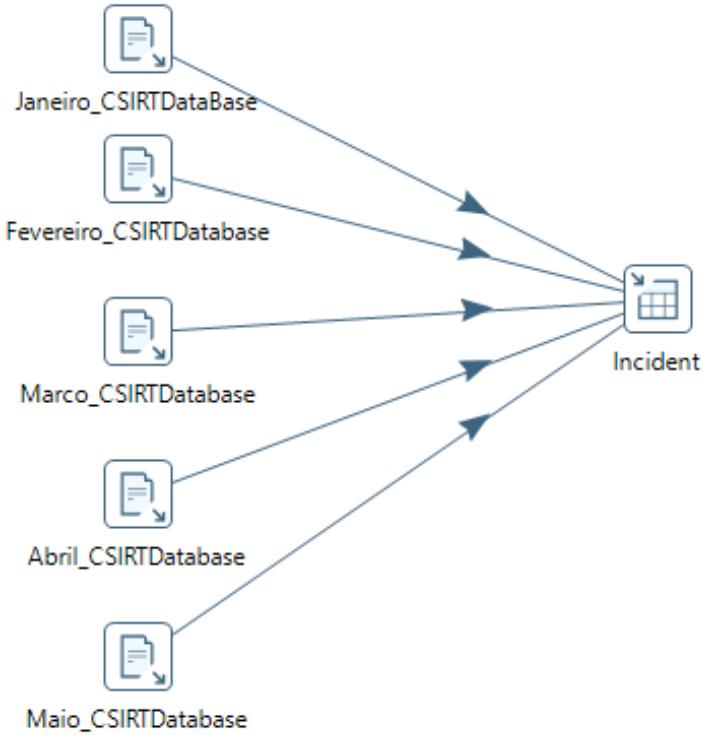



Figura 66 – Carga da base de dados do CSIRT na área de armazenamento intermediária

A base de dados do CAPEC disponibiliza 516 padrões de ataque categorizados por mecanismo de ataque e por domínio de ataque através dos arquivos 1000.csv e 3000.csv, respectivamente. Os dois arquivos contém a mesma estrutura e a única distinção entre eles é relação de especialização entre os tipos de ataques expressas no campo *Related Attack Patterns*. Esses arquivos foram carregados respectivamente nas tabelas *TypeAttackMechanism* e *TypeAttackDomain*, conforme a Figura 67 [2].

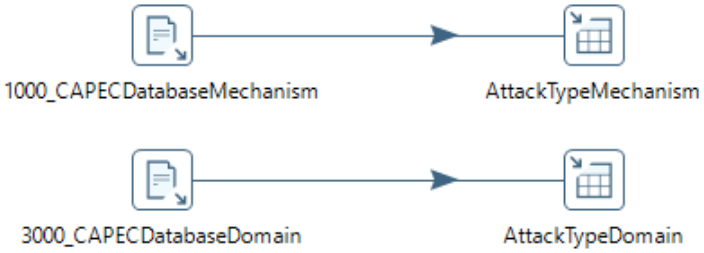



Figura 67 – Carga da base de dados do CAPEC na área de armazenamento intermediária

A base de dados do CWE disponibiliza 808 tipos de vulnerabilidades categorizadas de acordo com conceitos de pesquisa, de desenvolvimento e arquiteturais através dos arquivos 1000.csv, 699.csv e 1008.csv, respectivamente. Esses arquivos foram carrega-

[2] O comando SQL para criação das tabelas intermediárias de tipo de ataque encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

dos respectivamente nas tabelas *TypeVulnerableResarch*, *TypeVulnerableDevelopment* e *TypeVulnerableArchitectural*, conforme a Figura 68 [3].

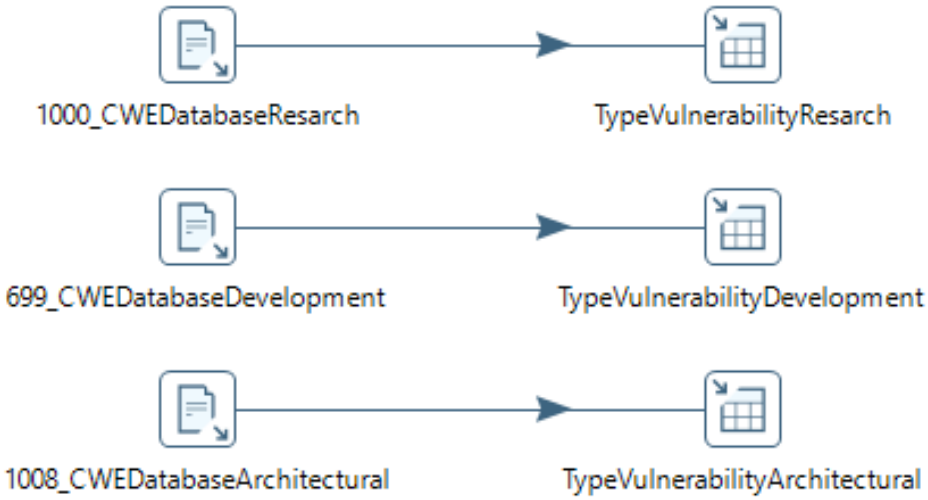



Figura 68 – Carga da base de dados do CWE na área de armazenamento intermediária

A extração de dados resultou em uma área intermediária de armazenamento de dados composta pela tabela *Incident* com os dados da base de incidentes do CSIRT, pelas tabelas *AttackTypeMechanism* e *AttackTypeDomain* com os dados da base do CAPEC e pelas tabelas *VulnerableTypeResarch*, *VulnerableTypeDevelopment* e *VulnerableTypeArchitectural* com os dados da base do CWE.

## 8.3 Realizar limpeza, transformação e carga nos dados

Os dados da área intermediária de armazenamento foram limpos e transformados para serem carregados no DW de acordo com o modelo lógico apresentado na Figura 65. Iniciando pela tabela *Incident*, os seus campos foram usados para carregar as tabelas DimTemporal, DimInformationAsset e DimPerson do DW, conforme Quadro 6 [4].

As tabelas *TypeAttackMechanism* e *TypeAttackDomain* possuem a mesma estrutura e, os seus campos *CAPEC-ID*, *Name* e *Related Attack Patterns* foram usados para carregar a hierarquia de tipo de tipo de ataque (*HTypeAttackType*) [5].

Na tabela intermediária *Incident*, o campo *Name* armazena o nome do tipo de ataque da base de dados de incidentes do CSIRT e os valores não utilizam o padrão de

[3] O comando SQL para criação das tabelas intermediárias de tipo de situação vulnerável encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

[4] O comando SQL para criação dessas tabelas encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

[5] O comando SQL para criação da hierarquia de tipo de tipo de ataque encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

Quadro 6 – Transformações realizadas usando a tabela intermediária *Incident*

| Campo da Tabela Incident | Tabela DW | Campo(s) DW |
|---|---|---|
| Time | DimTemporal | DateTimeComplet, Weekday, Month, Day, Hour, Minute, Second, Year |
| Attacker IP Address | DimInformationAsset | IP |
| Attacker Port | DimInformationAsset | Port |
| Target IP Address | DimInformationAsset | IP |
| Target Port | DimInformationAsset | Port |
| Domain | DimPerson | Domain |
| Device | DimPerson | Device |
| Interface | DimPerson | Interface |

nomenclatura de sCuDO. Porém, ao observar os valores deste campo foi constatado que há 487 tipos de ataque distintos e, dentre eles, 67 tipos de ataque têm informação do *Common Vulnerabilities and Exposures* (CVE), por exemplo, "HTTP: Microsoft Office Memory Corruption Vulnerability (CVE-2015-2477)". O CVE, sigla inglesa para vulnerabilidades e exposições comuns, é uma lista pública de falhas de segurança específicas de um produto ou sistema amplamente utilizada pela comunidade de segurança. No CVE, administrado pela MITRE Corporation, alguns registros estão associados as vulnerabilidades da base do CWE, a base que sCuDO usa para categorizar os tipos de situação vulnerável. E a base do CWE está associada aos padrões de ataque da base de dados do CAPEC, a base que sCuDO usa para categorizar os tipos de ataque. Essa associação entre CVE e CWE e entre CWE e CAPEC ofereceu uma oportunidade de associar os nomes dados aos tipos de ataque com informação do CVE na base de dados de incidentes do CSIRT a um tipo de ataque da base de dados do CAPEC. Os 67 tipos de ataques com informação do CVE foram analisados manualmente para verificar se havia informação sobre CWE associado para posteriormente verificar o padrão de ataque CAPEC associado ao CWE.

Ao pesquisar os CVE dos tipos de ataque foi verificado que 7 deles não tem CWE associados e outros 11 o CWE não estava associado a nenhum tipo de ataque da base de dados do CAPEC. Os outros 49 tipos de ataque da base de dados de incidentes do CSIRT foram associados ao tipo de ataque de sCuDO através das seguintes regras ordenadas por prioridade:

1 - Único tipo de ataque CAPEC associado a um único tipo de vulnerável do CWE;

2 - O tipo de ataque CAPEC que tem o mesmo nome do tipo de vulnerável do CWE;

3 - O tipo de ataque CAPEC que é supertipo de todos os outros tipos associados ao tipo de vulnerável do CWE;

4 - O tipo de ataque CAPEC que é supertipo com maior número de tipos de ataque associados ao tipo de vulnerável do CWE;

5 - Nome do tipo do CAPEC associado ao CWE mais semelhante ao nome do tipo de ataque da base de incidentes do CSIRT.

A lista completa dos 49 tipos de ataque da base de incidentes do CSIRT que foram associados aos tipos de ataque de sCuDO encontra-se em Apêndice A.

Para associar os outros tipos de ataque, foi criado um algoritmo em java que usa o método Jaro Distance (69), método que indica o percentual de semelhança entre *strings*, para descobrir o tipo de ataque do base de dados do CAPEC que mais se assemelha ao tipo de ataque da base de dados de incidentes do CSIRT. O algoritmo funciona da seguinte forma: ele recebe dois arquivos, um contendo a lista de tipos de ataque da base de dados de incidentes do CSIRT e outro a lista de tipos de ataque da base de dados do CAPEC. Cada tipo de ataque da base de dados de incidentes do CSIRT é comparado *string* a *string* com todos os tipos de ataque da base de dados do CAPEC. A comparação que obtive maior percentual de equivalência foi o tipo de ataque do CAPEC escolhido para associar ao tipo de ataque da base de incidentes do CSIRT [6].

Como resultado, cada tipo de ataque da base de incidentes do CSIRT foi associado ao tipo de ataque da base de dados do CAPEC mais semelhante, a Figura 69 mostra extrato dessa associação. Essa correspondência foi realizada para compatibilizar os tipos de ataque da base de dados de incidentes do CSIRT com os tipos de ataque de sCuDO, possibilitando que tipos possam ser especializados em subtipos ou agrupados em categorias de acordo com o proposto em sCuDO. E, o arquivo gerado pelo algoritmo e a lista manual de associação de tipos de ataques contendo o tipo de ataque da base do CSIRT e o tipo de ataque da base do CAPEC (disponível no Apêndice A) foram reunidos na tabela intermediária *AttackType* e esses dados foram usados para compor a dimensão tipo de ataque do DW (*DimAttackType*) [7].

Os tipos de ataque tem situações vulneráveis associadas. Os tipos de vulnerável estão armazenados nas tabelas intermediárias *TypeVulnerableResarch*, *TypeVulnerableDevelopment* e *TypeVulnerableArchitectural* de acordo com sua hierarquia de tipos. Para representar os tipos de vulnerável no ambiente de análise de dados os campos *CWE-ID*, *Name*, *Related Attack Patterns* dessas tabelas foram usados para carregar a hierarquia de tipo de vulnerável (*HVulnerableType*). E, o campo *Related Weaknesses* foi usado para carregar as hierarquias de supertipo de tipo de vulnerável por conceito de pesquisa (*HSuperVulnerableTypebyResarch*), supertipo de tipo de vulnerável por conceito de desenvolvimento

[6] Algoritmo para associar tipo de ataque encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

[7] O comando SQL para criação da dimensão tipo de ataque encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

| Name do tipo de ataque da base de incidente | Tipo de ataque do CAPEC |
|---|---|
| UDP: Too Many Inbound UDP Packets | UDP Flood |
| TCP: Full-Connect Port Scan | TCP Connect Scan |
| TCP Control Segment Anomaly | TCP Connect Scan |
| TCP: SYN Port Scan | TCP SYN Scan |
| HTTP: Blind SQL Injection - Exploit | Blind SQL Injection |
| ICMP: Timestamp Request Host Sweep | Timestamp Request |
| ICMP: Timestamp Probe | TCP Timestamp Probe |
| TCP: RST Resource Exhaustion DoS | TCP RST Injection |
| MALWARE: Malicious File Detected by GTI | Add Malicious File to Shared Webroot |
| TCP: Fingerprinting NMAP | Fingerprinting |
| HTTP: SQL Injection - Exploit III | SQL Injection |
| HTTP: WebDAV Large Body DoS | HTTP DoS |

Figura 69 – Associação do tipo de ataque do CAPEC ao tipo de ataque da base de incidentes do CSIRT usando semelhança entre *strings*

(*HSuperVulnerableTypebyDevelopment*) e supertipo de tipo de vulnerável por conceito arquiteturais (*HSuperVulnerableTypebyArchitectural*) [8].

Após construção de todas as dimensões e hierarquias, foi construída a tabela fato composta pelas chaves primárias das dimensões pessoa (*DimPerson*), ativo de informação (*DimInformationAsset*), temporal (*DimTemporal*) e tipo de ataque (*DimAttackType*). E, a medida foi obtida do campo *Attack count* da tabela *Incident* [9].

## 8.4 Disponibilizar dados para serem consultados

Com a construção do DW os dados já estão no formato favorável para serem consultados. Para facilitar a manipulação desses dados pelos usuários finais eles foram carregados em um ferramenta de processamento analítico *online* (*Online Analytical Processing* – OLAP) chamada Power BI. O Power BI, como toda ferramenta OLAP, facilita a navegabilidade, o mapeamento da fonte de dados e a construção de consultas.

O Power BI não oferece suporte a herança múltiplas. Porém, os incidentes são de um tipo de ataque que pode ter mais de um supertipo e este supertipo ser especialização de um tipo mais genérico. Essa relação de especialização de tipos é representada no DW através da tabela HTypeAttackType. Então, para os tipos que possuem mais de um supertipo foi eleito o supertipo mais recorrente para representá-lo. A Figura 70 mostra esses tipos e destaca em amarelo o supertipo que foi usado ao carregar os dados no Power BI.

[8] O comando SQL para criação da hierarquia de tipo de vulnerável encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

[9] O comando SQL para criação do fato *Incident* encontra-se em <https://github.com/MartaRigaud/DefesaAmbienteAnalitico>

| AttackType | Supertype AttackType |
|---|---|
| CAPEC-563: Add Malicious File to Shared Webroot | CAPEC-165: File Manipulation |
| | CAPEC-233: Privilege Escalation |
| | CAPEC-122: Privilege Abuse |
| CAPEC-634: Probe Audio and Video Peripherals | CAPEC-116: Excavation |
| | CAPEC-117: Interception |
| CAPEC-407: Pretexting | CAPEC-410: Information Elicitation |
| | CAPEC-416: Manipulate Human Behavior |
| CAPEC-58: Restful Privilege Elevation | CAPEC-122: Privilege Abuse |
| | CAPEC-233: Privilege Escalation |
| CAPEC-41: Using Meta-characters in E-mail Headers to Inject Malicious Payloads | CAPEC-242: Code Injection |
| | CAPEC-137: Parameter Injection |
| CAPEC-177: Create files with the same name as files protected with a higher classification | CAPEC-165: File Manipulation |
| | CAPEC-233: Privilege Escalation |
| | CAPEC-122: Privilege Abuse |
| CAPEC-562: Modify Shared File | CAPEC-165: File Manipulation |
| | CAPEC-233: Privilege Escalation |
| CAPEC-221: XML External Entities Blowup | CAPEC-122: Privilege Abuse |
| | CAPEC-272: Protocol Manipulation |

Figura 70 – Lista de tipos de ataque que tem mais de um supertipo no mesmo nível de especialização

Após o acerto na hierarquia de tipos de tipo de ataque os dados do DW foram carregados na ferramenta Power BI para serem consultados. As questões analíticas definidas no Capítulo 7 serão usadas para consultar os dados.

Por questão de confidencialidade as informações sobre ativo de informação e pessoa não serão exibidas. Ao mostrar os resultados das consultas tais informações foram substituídas por números. E, em virtude das fontes de dados usadas para alimentar o sistema de apoio à decisão não ter informação sobre o atacante, o resultado não autorizado causado pelo ataque e o dano causado pelo incidente, as seguintes questões de análise não serão respondidas: QA6- Qual o resultado não autorizado mais frequente? e QA7- Qual o dano mais frequente?.

Além disso, as seguintes questões analíticas serão adaptadas aos dados disponíveis:

QA10 - De: Qual o atacante que mais causou incidentes?

QA10 - Para: Qual foi o ativo de informação tutorado pelo atacante que mais causou incidentes?

QA12- De: O atacante sempre ataca a mesma vítima?

QA12- Para: Os ativos de informação tutorados pelo atacante sempre atacam a mesma vítima?

QA13- De: O atacante sempre comete o mesmo tipo de ataque?

QA13- Para: Os ativos de informação tutorados pelos atacantes cometem sempre o mesmo tipo de ataque?

A primeira consulta realizada foi para verificar quantos incidentes ocorreram (QA1 - Quantos incidentes ocorreram?). Conforme ilustrado na Figura 71, ocorrem 1.254.248

incidentes, esses incidentes foram causados por 1.718.148 ataques (QA2 - Quantos ataques ocorreram?), dando uma média de 1,37 ataques por incidente (QA3 - Quantos ataques levaram a ocorrência de cada incidente?).

| IncidentCount | AttakCount | Média de AttackCount |
|---|---|---|
| 1254248 | 1718148 | 1,37 |

Figura 71 – Total de incidentes e ataques e média de ataques por incidente

Ao observar os tipos de ataque por mecanismo de ataque (*Attack mechanism*), o tipo de ataque *Subvert Acess Control* foi responsável por maior número de incidentes (QA4 - Qual o tipo de ataque que ocasionou mais incidentes?), totalizando 946.436 ocorrências. Dessas 946.436 ocorrências, 931.783 delas foram causadas pelo subtipo *Exploration of Trusted Credentials*. E, a sua especialização *Remote Services with Stolen Credentials* somou 925.783 ocorrências, conforme ilustrado na Figura 72.

O tipo de ataque *Remote Services with Stolen Credentials* é associado a situação vulnerável *Insufficiently Protected Credentials*. E esta situação vulnerável está ligada 930.691 incidentes, sendo a mais frequente (QA5 - Qual foi a situação vulnerável mais frequente?), conforme ilustrado na Figura 73.

O ativo de informação alvo, isto é, o ativo de informação tutorado pela vítima mais atacado foi o ativo de informação com a identificação 358. Ele sofreu 98.863 ataques, causando 7.965 incidentes (QA8 - Qual o ativo de informação alvo mais atacado?). Todos esses incidentes foram causados pelo tipo de ataque *WiFi Mac Address Tracking*, conforme ilustrado na Figura 74.

O ativo de informação malicioso 259 foi o que mais atacou (QA9 - Qual o ativo de informação malicioso que mais atacou?), totalizando 160.546 ataques todos do tipo *WiFi Mac Address Tracking* para atingir dezenas de ativo de informação alvo da vítima, causando 19.156 incidentes (QA10 - Qual foi o ativo de informação tutorado pelo atacante que mais causou incidentes?), conforme ilustrado na Figura 75.

Entre as vítimas dos incidentes, a mais atingida foi a vítima 33 com 198.967 incidentes (QA11 - Qual a vítima que mais sofreu incidentes?), conforme ilustrado na Figura 76.

Os ativos maliciosos, isto é, os ativos de informação tutorados pelos atacantes geralmente atacam várias vítimas (QA12- Os Ativos de informação do atacante sempre atacam a mesma vítima?), conforme ilustrado na Figura 77. E, a maioria dos ataques causados por ativo de informação tutorados pelos atacantes são do mesmo tipo de ataque

| Attack Type | IncidentCount |
|---|---|
| ⊟ **Subvert Access Control** | **946436** |
| ⊟ **Exploitation of Trusted Credentials** | **931783** |
| Remote Services with Stolen Credentials | 925783 |
| Use of Known Domain Credentials | 4904 |
| Session Credential Falsification through Manipulation | 973 |
| Session Credential Falsification through Prediction | 75 |
| Kerberoasting | 33 |
| SaaS User Request Forgery | 12 |
| Windows Admin Shares with Stolen Credentials | 2 |
| Session Hijacking | 1 |
| ⊞ **Exploiting Trust in Client** | **14100** |
| ⊞ **Privilege Abuse** | **244** |
| ⊞ **Privilege Escalation** | **148** |
| ⊞ **Bypassing Physical Security** | **128** |
| ⊞ **Man in the Middle Attack** | **18** |
| ⊞ **Authentication Bypass** | **15** |
| ⊞ **Collect and Analyze Information** | **149383** |
| ⊞ **Engage in Deceptive Interactions** | **58936** |
| ⊞ **Manipulate System Resources** | **50729** |
| ⊞ **Abuse Existing Functionality** | **31110** |
| ⊞ **Inject Unexpected Items** | **12152** |
| ⊞ **Manipulate Data Structures** | **4894** |
| ⊞ **Employ Probabilistic Techniques** | **592** |
| ⊞ **Manipulate Timing and State** | **16** |
| **Total** | **1254248** |

Figura 72 – Total de incidentes por tipo de ataque (*Attack Type*)

(QA13- Os ativos de informação tutorados pelos atacantes cometem sempre o mesmo tipo de ataque?), conforme ilustrado na Figura 78.

| Vulnerable Type | IncidentCount |
|---|---|
| ⊟ **Insufficiently Protected Credentials** | **930691** |
| Remote Services with Stolen Credentials | 925783 |
| Use of Known Domain Credentials | 4904 |
| Password Recovery Exploitation | 2 |
| Windows Admin Shares with Stolen Credentials | 2 |
| ⊞ **Channel Accessible by Non-Endpoint ('Man-in-the-Middle')** | **71297** |
| ⊞ **Information Exposure** | **68475** |
| ⊞ **Information Exposure Through Sent Data** | **58656** |
| ⊞ **Allocation of Resources Without Limits or Throttling** | **20466** |
| ⊞ **Missing Release of Resource after Effective Lifetime** | **19035** |
| ⊞ **Hidden Functionality** | **18464** |
| ⊞ **Protection Mechanism Failure** | **12277** |
| ⊞ **Improper Authentication** | **12187** |
| ⊞ **Use of Insufficiently Random Values** | **11503** |
| ⊞ **Improper Neutralization of Special Elements in Output Used by a Downstream Component ('Injection')** | **10744** |
| ⊞ **Interpretation Conflict** | **8563** |
| ⊞ **Incorrect Comparison** | **8386** |
| ⊞ **Improper Input Validation** | **7680** |
| ⊞ **Download of Code Without Integrity Check** | **7402** |
| ⊞ **Improper Enforcement of Message or Data Structure** | **6207** |
| ⊞ **Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling')** | **5907** |
| ⊞ **Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')** | **5749** |
| ⊞ **Improper Neutralization of Argument Delimiters in a Command ('Argument Injection')** | **4850** |
| ⊞ **Improper Handling of Extra Parameters** | **4507** |
| ⊞ **Improper Neutralization of Input Terminators** | **4507** |
| ⊞ **Reliance on Reverse DNS Resolution for a Security-Critical Action** | **3729** |
| ⊞ **Improper Restriction of Operations within the Bounds of a Memory Buffer** | **2977** |
| ⊞ **Improper Neutralization of Trailing Special Elements** | **2202** |
| ⊞ **Missing Encryption of Sensitive Data** | **1971** |
| ⊞ **Information Exposure Through an Error Message** | **1918** |
| ⊞ **External Control of File Name or Path** | **1896** |
| **Total** | **1254248** |

Figura 73 – Total de incidentes por tipo vulnerável (*Vulnerable Type*) e tipo de ataque (*Attack Type*)

| Information Asset tutored by the Victim | Attack Count | Incident Count |
|---|---|---|
| ⊟ 358 | 98863 | 7965 |
| WiFi MAC Address Tracking | 98863 | 7965 |
| ⊞ 883012 | 80374 | 1390 |
| ⊞ 852116 | 78428 | 12375 |
| ⊞ 755919 | 31571 | 537 |
| ⊞ 228877 | 23318 | 811 |
| ⊞ 229077 | 17422 | 2269 |
| ⊞ 149147 | 14453 | 255 |
| ⊞ 213 | 12363 | 6260 |
| ⊞ 166 | 11919 | 10520 |
| ⊞ 592340 | 10828 | 10824 |
| ⊞ 333 | 7275 | 812 |
| ⊞ 314 | 7041 | 854 |
| ⊞ 309 | 6613 | 4995 |
| ⊞ 823475 | 6559 | 6559 |
| ⊞ 169 | 6538 | 686 |
| ⊞ 733742 | 6421 | 1665 |
| ⊞ 325 | 6210 | 739 |
| ⊞ 861785 | 6209 | 1939 |
| ⊞ 614837 | 6018 | 6018 |
| ⊞ 614836 | 5776 | 3288 |
| ⊞ 168 | 5775 | 1128 |
| ⊞ 352357 | 4695 | 27 |
| ⊞ 326785 | 4669 | 181 |
| ⊞ 529542 | 3710 | 1237 |
| ⊞ 261 | 3450 | 1845 |
| ⊞ 189 | 3017 | 803 |
| ⊞ 861786 | 2899 | 2899 |
| ⊞ 229079 | 2858 | 2858 |
| Total | 1718148 | 1254248 |

Figura 74 – Quantidade de ataques (*Attack Count*) e incidentes (*Incident Count*) por ativo de informação tutorado pela vítima (*Information Asset Tutored by the Victim*)

| Information Asset tutored by the attacker | Attack Count | Incident Count |
|---|---|---|
| ⊟ 259 | 160546 | 19156 |
| WiFi MAC Address Tracking | 160546 | 19156 |
| ⊞ 883013 | 80376 | 1391 |
| ⊞ 852126 | 49050 | 4482 |
| ⊞ 755920 | 31580 | 546 |
| ⊞ 228870 | 27035 | 1800 |
| ⊞ 852125 | 14076 | 3586 |
| ⊞ 213 | 12327 | 6252 |
| ⊞ 852124 | 12271 | 2870 |
| ⊞ 330412 | 11816 | 211 |
| ⊞ 166 | 11198 | 8012 |
| ⊞ 252 | 11036 | 9923 |
| ⊞ 720324 | 8449 | 1330 |
| ⊞ 113024 | 7230 | 7221 |
| ⊞ 383780 | 6401 | 1652 |
| ⊞ 729946 | 5764 | 5764 |
| ⊞ 329895 | 5382 | 523 |
| ⊞ 611024 | 5222 | 1741 |
| ⊞ 856049 | 4726 | 25 |
| ⊞ 326786 | 4668 | 180 |
| ⊞ 756748 | 4536 | 2479 |
| ⊞ 720325 | 3534 | 3534 |
| ⊞ 824913 | 3410 | 3410 |
| ⊞ 529542 | 3363 | 463 |
| ⊞ 852127 | 3030 | 1436 |
| ⊞ 529583 | 2832 | 2832 |
| ⊞ 327325 | 2547 | 2547 |
| ⊞ 824887 | 2386 | 2386 |
| ⊞ 824901 | 2281 | 1042 |
| ⊞ 905729 | 2278 | 280 |
| Total | 1718148 | 1254248 |

Figura 75 – Quantidade de ataques (*Attack Count*) e de incidentes (*Incident Count*) por Ativo de informação tutorado pela atacante (*Information Asset Tutored by the Attacker*)

| Victim | Incident Count |
|---|---|
| 33 | 198967 |
| 8 | 176946 |
| 3 | 144907 |
| 24 | 133610 |
| 26 | 96429 |
| 18 | 91998 |
| 6 | 73338 |
| 38 | 55757 |
| 39 | 42352 |
| 13 | 32271 |
| 19 | 32176 |
| 32 | 30548 |
| 10 | 22536 |
| 7 | 21877 |
| 23 | 20260 |
| 35 | 14728 |
| 2 | 10396 |
| 30 | 8281 |
| 31 | 7960 |
| 11 | 7682 |
| 16 | 6183 |
| 14 | 4377 |
| 36 | 2687 |
| Total | 1254248 |

Figura 76 – Quantidade de incidentes (*Incident Count*) por Vítima (*Victim*)

| Information Asset Attacker / Victim | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 259 | | | | | | 160546 | | | | | | | | | | | | | | | | | | |
| 883013 | | | | | | 80376 | | | | | | | | | | | | | | | | | | |
| 852126 | | | | | | 49050 | | | | | | | | | | | | | | | | | | |
| 755920 | | | | | | 31580 | | | | | | | | | | | | | | | | | | |
| 228870 | | | | 27035 | | | | | | | | | | | | | | | | | | | | |
| 852125 | | | | | | 14076 | | | | | | | | | | | | | | | | | | |
| 213 | | | | | | 12327 | | | | | | | | | | | | | | | | | | |
| 852124 | | | | | | 12271 | | | | | | | | | | | | | | | | | | |
| 330412 | | | | | | | | | | | | | | | | | 1160 | | | | | | | |
| 166 | | | | | | 11198 | | | | | | | | | | | | | | | | | | |
| 252 | | | | | | 11036 | | | | | | | | | | | | | | | | | | |
| 720324 | | 178 | 88 | 14 | | | 1960 | 1088 | | 7 | | | 8 | | | | 4 | 11 | 1 | | | | | |
| 113024 | | | | | | | | | | | | | | | | | | | | | | | | |
| 383780 | | | | | | | | | | | | | | | | | | | | | | | | |
| 729946 | | | | | | | | | | | | | | | | | | | | | | | | |
| 329895 | | | | | | | | | | | | | | | | | 4520 | | | | | | | 80 |
| 611024 | | | | | | | | | | | | | | | | | | | | | | | | |
| 856049 | | | | | | | | | | | | | | | | | | | | | | | | |
| 326786 | | | | | | | | | | | | | | | | | | | | | | | | 46 |
| 756748 | | | | | | | | | | 305 | 4222 | | | | | | | | | | | | | |
| 720325 | | 50 | 52 | 14 | | | 379 | 258 | | 67 | | | 86 | | 2 | | 37 | 120 | 19 | | | | 3 | |
| 824913 | | | | | | 3410 | | | | | | | | | | | | | | | | | | |
| 529542 | 32 | 18 | | | | | | 275 | | | | | | | | | | 129 | | | | | 1 | |
| 852127 | | | | | | 3030 | | | | | | | | | | | | | | | | | | |
| 529583 | 17 | 68 | | | | | | 349 | | | | | | | | | | 724 | | | | | | |
| 327325 | | | | | | | | | | | | | | | | | | | | | | | | |
| 824887 | | | | | | 2386 | | | | | | | | | | | | | | | | | | |
| 824901 | | | | | | 2281 | | | | | | | | | | | | | | | | | | |
| 905729 | | | | | | | | | | | | | | | | | | | | | | | | |
| **Total** | **2201** | **10814** | **146442** | **28413** | **2787** | **412654** | **24403** | **178865** | **9** | **22988** | **9816** | **2005** | **32532** | **5522** | **31** | **7941** | **7805** | **92538** | **326** | **142** | **126** | **2149** | **20546** | **13** |
| | | | | | | | | | | | | | | | | | | | **77** | | | | | **91** |

Figura 77 – Relação entre quantidade de ataque por ativo de informação tutorado pelo atacante (*Information Asset Tutored Attacker*) e vítimas (*Victim*)

| Information Asset Attacker / Attack Type | Abuse Existing Functionality | Collect and Analyze Information | Employ Probabilistic Techniques | Engage in Deceptive Interactions | Inject Unexpected Items | Manipulate Data Structures | Manipulate System Resources | Manipulate Timing and State | Subvert Access Control | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| 259 | | 160546 | | | | | | | | 160546 |
| 883013 | | 80376 | | | | | | | | 80376 |
| 852126 | | 49050 | | | | | | | | 49050 |
| 755920 | | 31580 | | | | | | | | 31580 |
| 228870 | | 26985 | | 50 | | | | | | 27035 |
| 852125 | | 14076 | | | | | | | | 14076 |
| 213 | | | | 12327 | | | | | | 12327 |
| 852124 | | 12271 | | | | | | | | 12271 |
| 330412 | 8480 | 1032 | | | 2248 | 10 | 45 | | 1 | 11816 |
| 166 | | 11171 | | 27 | | | | | | 11198 |
| 252 | | 11036 | | | | | | | | 11036 |
| 720324 | 8127 | 322 | | | | | | | | 8449 |
| 113024 | 3 | 7151 | | | | | 1 | | 75 | 7230 |
| 383780 | | | | 6401 | | | | | | 6401 |
| 729946 | | 5764 | | | | | | | | 5764 |
| 329895 | 4606 | 589 | | | 91 | 3 | 22 | 71 | | 5382 |
| 611024 | | | | | | | 5222 | | | 5222 |
| 856049 | | | | | 4686 | | 40 | | | 4726 |
| 326786 | | 4668 | | | | | | | | 4668 |
| 756748 | 201 | 5 | | 4079 | 56 | 2 | 181 | | 12 | 4536 |
| 720325 | 1131 | 2402 | | | 1 | | | | | 3534 |
| 824913 | | 3410 | | | | | | | | 3410 |
| 529542 | 2930 | 21 | | | 12 | 64 | 278 | | 58 | 3363 |
| 852127 | | 3030 | | | | | | | | 3030 |
| 529583 | 923 | 249 | | 15 | 42 | 176 | 839 | | 588 | 2832 |
| 327325 | | | | 2547 | | | | | | 2547 |
| **Total** | **65297** | **517080** | **592** | **74050** | **23813** | **5210** | **80876** | **107** | **951123** | **1718148** |

Figura 78 – Relação entre quantidade de ataque por ativo de informação tutorado pelo atacante (*Information Asset Tutored Attacker*) e tipo de ataque (*Attack Type*)

Os ativos de informação alvo, ou seja, os ativos de informação tutorados pelas vítimas geralmente sofrem o mesmo tipo de ataque (QA14- O alvo sempre sofre o mesmo tipo de ataque?), conforme ilustrado na Figura 79. E, consequentemente, os alvos são atacados em virtude da mesma situação vulnerável (QA16- O alvo é atacado em virtude da mesma situação vulnerável?).

| Information Asset Victim / Attack Type | Abuse Existing Functionality | Collect and Analyze Information | Employ Probabilistic Techniques | Engage in Deceptive Interactions | Inject Unexpected Items | Manipulate Data Structures | Manipulate System Resources | Manipulate Timing and State | Subvert Access Control | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| 358 | | 98863 | | | | | | | | 98863 |
| 883012 | | 80374 | | | | | | | | 80374 |
| 852116 | | 78428 | | | | | | | | 78428 |
| 755919 | | 31571 | | | | | | | | 31571 |
| 228877 | | 23318 | | | | | | | | 23318 |
| 229077 | | | | | | | 17422 | | | 17422 |
| 149147 | 11062 | 1029 | | | 2274 | 10 | 51 | 26 | 1 | 14453 |
| 213 | | 36 | | 12327 | | | | | | 12363 |
| 166 | | 11919 | | | | | | | | 11919 |
| 592340 | | 10823 | | 5 | | | | | | 10828 |
| 333 | | 7275 | | | | | | | | 7275 |
| 314 | | 7041 | | | | | | | | 7041 |
| 309 | | 6613 | | | | | | | | 6613 |
| 823475 | | 6557 | | | | | 1 | | 1 | 6559 |
| 169 | | 6538 | | | | | | | | 6538 |
| 733742 | | 12 | | 6401 | 8 | | | | | 6421 |
| 325 | | 6210 | | | | | | | | 6210 |
| 861785 | 20 | | | | | | | | 6189 | 6209 |
| 614837 | | | | 6018 | | | | | | 6018 |
| 614836 | | | | 5712 | | | 64 | | | 5776 |
| 168 | | 5775 | | | | | | | | 5775 |
| 352357 | | | | | 4687 | | 8 | | | 4695 |
| 326785 | | 4668 | | 1 | | | | | | 4669 |
| 529542 | 1669 | 711 | | 33 | 341 | 2 | 954 | | | 3710 |
| 261 | | 3450 | | | | | | | | 3450 |
| 189 | | 3017 | | | | | | | | 3017 |
| 861786 | 75 | | | 1 | | 1 | | | 2822 | 2899 |
| 229079 | | | | | | | 2858 | | | 2858 |
| 865059 | 2638 | 7 | | | | | | | | 2645 |
| 811634 | | 979 | | | | | 1533 | | | 2512 |
| **Total** | **65297** | **517080** | **592** | **74050** | **23813** | **5210** | **80876** | **107** | **951123** | **1718148** |

Figura 79 – Relação entre quantidade de ataque por ativo de informação tutorado pela vítima (*Information Asset Tutored Attacker*) e tipo de ataque (*Attack Type*)

Porém, o mesmo não ocorre com a vítima. A Figura 80 mostra que as vítimas sofreram tipos de ataque distintos (QA15- A vítima sempre sofre o mesmo tipo de ataque?). Tais tipos de ataque geralmente são de natureza distinta, sendo causados por diversos tipos de situação vulnerável (QA17- A vítima é atacada em virtude da mesma situação vulnerável?). Para exemplificar essa variedade, a Figura 81 ilustra os tipos de situação vulnerável da vítima 33, a vítima com maior número de ocorrências de incidentes.

| Victim / Attack Type | Abuse Existing Functionality | Collect and Analyze Information | Employ Probabilistic Techniques | Engage in Deceptive Interactions | Inject Unexpected Items | Manipulate Data Structures | Manipulate System Resources | Manipulate Timing and State | Subvert Access Control | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| 33 | 4363 | 22318 | 127 | 8765 | 2389 | 1209 | 9242 | 5 | 150549 | **198967** |
| 8 | 697 | 1136 | 7 | 31 | 402 | 69 | 4981 | | 169623 | **176946** |
| 3 | 830 | 2630 | 321 | 477 | 1244 | 87 | 5053 | | 134265 | **144907** |
| 24 | 302 | 606 | | 15 | 61 | 12 | 1828 | 8 | 130778 | **133610** |
| 26 | 683 | 788 | | 14 | 37 | 41 | 1936 | | 92930 | **96429** |
| 18 | 705 | 644 | 45 | 81 | 205 | 44 | 1261 | | 89013 | **91998** |
| 6 | | 64820 | | 7224 | | | 980 | | 314 | **73338** |
| 38 | 6289 | 19962 | | 13417 | 2710 | 1925 | 5451 | | 6003 | **55757** |
| 39 | 3402 | 9514 | 5 | 1652 | 483 | 200 | 6830 | | 20266 | **42352** |
| 13 | 131 | 283 | 9 | 35 | 136 | 46 | 57 | | 31574 | **32271** |
| 19 | 884 | 419 | | 6 | 106 | 13 | 313 | | 30435 | **32176** |
| 32 | 610 | 10080 | | 10662 | 176 | 109 | 4369 | | 4542 | **30548** |
| 10 | 622 | 198 | | 7 | 177 | 117 | 889 | | 20526 | **22536** |
| 7 | 767 | 252 | 70 | 11 | 1445 | 39 | 172 | 1 | 19120 | **21877** |
| 23 | 244 | 154 | 1 | 1 | 32 | 58 | 317 | | 19453 | **20260** |
| 35 | 320 | 316 | 4 | 997 | 982 | 250 | 638 | | 11221 | **14728** |
| 2 | 216 | 40 | 1 | 307 | 40 | 16 | 80 | | 9696 | **10396** |
| 30 | 1755 | 1495 | | 671 | 463 | 132 | 2858 | | 907 | **8281** |
| 31 | 3580 | 2693 | | 31 | 419 | 222 | 66 | | 949 | **7960** |
| 11 | 681 | 13 | | 6647 | 129 | 15 | 194 | | 3 | **7682** |
| 16 | 467 | 3001 | | 544 | 33 | 122 | 1352 | | 664 | **6183** |
| 14 | 991 | 1 | | 3252 | 28 | 1 | 97 | | 7 | **4377** |
| 36 | 2 | 1586 | | 795 | 10 | 8 | 56 | | 230 | **2687** |
| 4 | 298 | 1843 | | 52 | 18 | 57 | 280 | | 30 | **2578** |
| 20 | 4 | 5 | | 2277 | 10 | 1 | 13 | | 8 | 2327 |
| **Total** | **31110** | **149383** | **592** | **58936** | **12152** | **4894** | **50729** | **16** | **946436** | **1254248** |

Figura 80 – Relação entre quantidade de incidentes por vítima (*Victim*) e tipo de ataque (*Attack Type*)

| Person Victim | IncidentCount |
|---|---|
| ⊟ 33 | 198967 |
| Insufficiently Protected Credentials | 149938 |
| Information Exposure | 20117 |
| Allocation of Resources Without Limits or Throttling | 4135 |
| Missing Release of Resource after Effective Lifetime | 4014 |
| Reliance on Reverse DNS Resolution for a Security-Critical Action | 3541 |
| Improper Neutralization of Trailing Special Elements | 2190 |
| Hidden Functionality | 1781 |
| Incorrect Comparison | 1359 |
| Improper Input Validation | 1329 |
| Improper Neutralization of Special Elements in Output Used by a Downstream Component ('Injection') | 1308 |
| Improper Neutralization of Argument Delimiters in a Command ('Argument Injection') | 1261 |
| Improper Handling of Extra Parameters | 1206 |
| Improper Neutralization of Input Terminators | 1206 |
| Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') | 872 |
| Improper Enforcement of Message or Data Structure | 863 |
| Improper Restriction of Operations within the Bounds of a Memory Buffer | 750 |
| External Control of File Name or Path | 455 |
| Improper Handling of Alternate Encoding | 452 |
| Incorrect Behavior Order: Validate Before Canonicalize | 452 |
| Incorrect Behavior Order: Validate Before Filter | 452 |
| Encoding Error | 450 |
| Incomplete Blacklist to Cross-Site Scripting | 450 |
| External Control of Assumed-Immutable Web Parameter | 315 |
| Reliance on Cookies without Validation and Integrity Checking | 315 |
| Channel Accessible by Non-Endpoint ('Man-in-the-Middle') | 294 |
| Missing Encryption of Sensitive Data | 270 |
| Weak Password Requirements | 127 |
| Authentication Bypass by Assumed-Immutable Data | 125 |
| Reliance on a Single Factor in a Security Decision | 125 |
| Inadequate Encryption Strength | 124 |
| Use of Insufficiently Random Values | 124 |
| Insecure Temporary File | 78 |
| Improper Neutralization of Script in Attributes in a Web Page | 70 |
| Interpretation Conflict | 61 |
| **Total** | **1254248** |

Figura 81 – Relação entre quantidade de incidentes da vítima (*Victim*) 33 e tipo de vulnerável (*Vulnerable Type*)

# 9 CONCLUSÕES E CONSIDERAÇÕES FINAIS

As organizações têm usado como estratégia de defesa cibernética reunir informações de Incidentes de Segurança de Informação em um único ambiente para analisá-las e compará-las com o objetivo de oferecer suporte à tomada de decisão que leve a uma redução significativa do número de ocorrências de incidentes. Porém, enfrentam dificuldades inerentes à heterogeneidade semântica dos conceitos e as diferentes formas de categorização dos incidentes.

Há várias ontologias de Incidente de Segurança de Informação. Todas essas diferentes formas de representação e, até mesmo a falta de uma representação formal, tornam a troca de informações sobre Incidentes de Segurança da Informação bastante complexa mesmo para organizações renomadas na área de segurança. Para prover um apoio metodológico conceitual para a troca de informações sobre Incidentes de Segurança da Informação visando oferecer suporte à tomada de decisões na área de defesa cibernética este trabalho desenvolve e implementa a metodologia DEFESA.

A metodologia DEFESA define um macroprocesso para modelar ontologia de domínio bem fundamentada na UFO-MLT com o objetivo de fornecer atividades para representação dos conceitos do domínio de forma mais explícita, das relações entre eles e dos critérios de classificação dos tipos. Desta forma, a ontologia de domínio construída utilizando DEFESA pode ser usada em consenso para representar incidentes e ajudar na troca de informações. As informações compartilhadas devem ser reunidas em um ambiente apropriado para analisá-las, para tal DEFESA define um macroprocesso para desenvolver ambiente analítico de dados.

A aplicação da metodologia DEFESA para análise de Incidentes de Segurança da Informação foi uma das principais contribuições deste trabalho. Utilizando DEFESA, foi modelado sCuDO, a ontologia de domínio baseada na UFO-MLT de Incidente de Segurança da Informação, para representação dos conceitos, suas relações e seus tipos do domínio. Com sCuDo, foram representados dois incidentes e foi feita uma associação entre as entidades do domínio e os campos de uma base de dados de incidentes de um Grupo de Resposta a Incidentes de Segurança em Computadores.

A definição de regras de transformação de entidades ontológicas tipificadas usando UFO-MLT em conceitos dimensionais, se caracteriza em uma outra contribuição que também merece destaque. Tais regras foram utilizadas para gerar sCuD$^2$O, o modelo dimensional de Incidentes de Segurança da Informação. Seguindo a arquitetura em camadas proposta pela metodologia DEFESA, baseado em sCuD$^2$O foi desenvolvido o ambiente analítico de dados, utilizando os dados da base de dados de incidentes de um Grupo

de Resposta a Incidentes de Segurança em Computadores, da base de dados de padrões de ataque do CAPEC e da base de dados de vulnerabilidade do CWE. E, como última contribuição, esses dados foram analisados usando como referencial para consulta as questões analíticas definidas durante a implementação da metodologia DEFESA.

Durante o desenvolvimento deste trabalho foram encontradas algumas dificuldades. Enquanto estava sendo elaborada à ontologia de domínio baseada na UFO-MLT muitos trabalhos novos foram publicados reformulando alguns fundamentos da UFO, como por exemplo, os esteriótipos *pre-state* e *pos-state* foram substituídos por *triggers* e *bringsAbout*, respectivamente. Tais mudanças levaram a retomada do estudo sobre a UFO e atualização de sCuDO. Por outro lado, esses trabalhos também esclareceram muitas dúvidas, principalmente, com relação ao emprego dos tipos da UFO-B. Tais fundamentos foram de grande importância para definição das regras de transformação das entidades ontológicas em conceitos dimensionais. Além disso, há poucos trabalhos que desenvolvem ontologias utilizando a MLT e trabalhos com UFO-MLT são ainda mais escassos, tornando o processo de desenvolvimento mais demorado.

Outra dificuldade, em virtude da confidencialidade característica desse domínio, foi a obtenção de fontes de dados de Incidentes de Segurança da Informação para a composição do ambiente analítico. O objetivo inicial era integração de várias fontes de dados de ocorrências de incidentes heterogêneas para validar a aplicabilidade da metodologia em sua totalidade. Porém, só foi obtida uma base de dados de incidentes. A Rede Nacional de Ensino e Pesquisa (RNP) tem envidado esforços para institucionalizar a disponibilização de dados de pesquisa [1], sabendo disso foi feita uma solicitação de dados, porém eles não foram disponibilizados a tempo para serem usados neste trabalho.

Essas dificuldades foram administradas e o trabalho foi concluído. O uso da UFO-MLT para elaborar sCuDO melhorou a expressividade semântica do domínio quando comparado com outras as ontologias de Incidente de Segurança da Informação pesquisadas. Porém, ainda há alguns pontos para serem melhorados. Ao ler a descrição textual de sCuDO, está descrito que o atacante é responsável pelo ativo de informação malicioso e a vítima é responsável pelo ativo de informação alvo. Porém, no diagrama este relacionamento foi modelado de forma genérica, a pessoa (*Person*) desempenha o papel relacional (*Relational Role*) de responsável (*Responsible*) por um ativo de informação (*Information Asset*) devido a impossibilidade de especialização entre (*Relational Role*) e papel processual (*Processual Role*) pois, esses papéis são de natureza distintas. O ideal seria o papel processual de atacante e de vítima poderem ser uma especialização do papel relacional responsável. Desta forma, poderiam ser modeladas as especializações da relação material (relator) de aquisição de posse (*OWnership acquisition*), a aquisição de posse do atacante e da vítima e, por fim, representar diretamente as relações entre atacante e ativo de informação malicioso

[1] https://dadosabertos.rnp.br/

e entre vítima e ativo de informação alvo.

Como trabalho futuro, as ontologias que representam parte do domínio de Incidente de Segurança de Informação poderiam ser correlacionadas utilizando sCuDO como referência, formando uma rede de ontologias de Incidente de Segurança da Informação integradas.

Outra abordagem interessante seria ampliar sCuDO para representar o risco eminente do incidente e as disposições do ativo de informação alvo que os levam a situação vulnerável ao ataque. Com relação ao risco, há uma ontologia de risco representando a experiência de risco como um evento indesejado com potencial de causar perdas. Essa experiência de risco é composta por eventos de risco que podem ser um evento de ameaça ou um evento de perda. O evento de ameaça é aquele com o potencial de causar uma perda, que pode ser intencional, como um ataque de hackers, ou não intencional. E, o evento de perda necessariamente afeta as intenções de maneira negativa (70). Essa estrutura se assemelha muito com o Incidente modelado em sCuDO. O Incidente pode ser considerado uma experiência de risco e o ataque um evento de perda e, a partir dessa associação as outras entidades da ontologia de risco poderiam ser incorporadas a sCuDO.

A ontologia de software proposta por Duarte et al.(60) detalha as disposições manifestadas quando ocorre um ataque devido a falha de software. Tal representação poderia ser correlacionada a sCuDo e, ainda poderia servir de base para representação de disposições de outros tipos de ataque, aumentando a abrangência da representação do domínio.

Além disso, a metodologia DEFESA poderia ser aplicada em outros domínios ou atendender a outras necessidades para que ela seja testada, amadurecida e validada. E, poderia ser desenvolvido um ambiente de análise com dados heterogêneos oriundos de diversas fontes de ocorrências de incidentes.

# REFERÊNCIAS

1 GUARINO, N. Formal ontology in information systems. In: *first international conference (FOIS'98)*. [S.l.]: FOIS'98, 1998. p. 3 – 15. IOS press v. 46. 7, 25, 26

2 RUY, F. B.; FALBO, R. de A.; BARCELLOS, M. P.; COSTA, S. D.; GUIZZARDI, G. Seon: A software engineering ontology network. In: *EKAW*. [S.l.: s.n.], 2016. 7, 25, 26

3 GUIZZARDI, G. Ontological foundations for structural conceptual models. 2005. 7, 27, 28, 29, 30, 31

4 GUIZZARDI, G.; FONSECA, C. M.; BENEVIDES, A. B.; ALMEIDA, J. P. A.; PORELLO, D.; SALES, T. P. Endurant types in ontology-driven conceptual modeling: Towards ontouml 2.0. In: SPRINGER. *International Conference on Conceptual Modeling*. [S.l.], 2018. p. 136–150. 7, 29, 30, 31

5 BENEVIDES, A. B.; BOURGUET, J.-R.; GUIZZARDI, G.; PEÑALOZA, R.; ALMEIDA, J. P. A. Representing a reference foundational ontology of events in sroiq. IOS Press, n. Preprint, p. 1–42, 2019. 7, 26, 32, 33, 34, 35

6 CARVALHO, V. A.; ALMEIDA, J. P. A.; FONSECA, C. M.; GUIZZARDI, G. Multi-level ontology-based conceptual modeling. *Data & Knowledge Engineering*, Elsevier, v. 109, p. 3–24, 2017. 7, 36, 37, 38, 40

7 FARIA, M.; FIGUEIREDO, G.; CORDEIRO, K.; CAVALCANTI, M.; CAMPOS, M. Applying multi-level theory to an information security incident domain ontology. In: ONTOBRAS. *Ontobras*. [S.l.], 2019. p. XXX. 7, 18, 40, 41

8 FONSECA, F. d. C. S.; BELLOZE, K. T. Estudo e implementação de um modelo de ontologia tendo como domínio os atendimentos realizados pelo samu. *Relatórios Técnicos do DCC/UFJF*, 2010. 7, 43

9 FALBO, R. d. A. Sabio: Systematic approach for building ontologies. In: *ONTO. COM/ODISE@ FOIS*. [S.l.: s.n.], 2014. 7, 43, 44

10 MOREIRA, G. B. *Uma Ontologia para Tratamento de Incidentes de Segurança da Informação*. Dissertação (Mestrado) — Instituto Militar de Engenharia (IME), Rio de Janeiro, 2018. 7, 22, 51, 52, 54, 88

11 PING, L.; HAIFENG, Y.; GUOQING, M. An incident response decision support system based on cbr and ontology. In: *ICCASM*. [S.l.: s.n.], 2010. p. 337 – 340. 7, 22, 51, 52, 54, 88, 89

12 LI, W.; TIAN, S. An ontology-based intrusion alerts correlation system. In: *CIbSE*. Beijing 100044, China: Expert Systems with Applications 37, 2010. (X, 37), p. 7138–7146. 7, 53, 54, 88

13 SWIMMER, M. *Towards an Ontology of Malware Classes*. 2008. January 27, 2008. Disponível em: <http://www.scribd.com/doc/24058261/Towards-an-Ontology-of-Malware-Classes>. 7, 53, 54

14 ANSARINIA, M.; ASGHARI, S. A.; SOUZANI, A.; GHAZNAVI, A. Ontology-based modeling of ddos attacks for attack plan detection. *6th International Symposium on Telecommunications*, IST 2012, v. 6th, p. 1–5, 2012. 7, 53, 54, 88, 89

15 FUERTES, W.; REYES, F.; VALLADARES, P.; TAPIA, F.; TOULKERIDIS, T.; PéREZ, E. An integral model to provide reactive and proactive services in an academic csirt based on business intelligence. In: *Systems*. [S.l.]: doi:10.3390/systems5040052, 2017. p. 52 – 71. 7, 55, 56

16 BOUCHRA, A.; WAKRIME, A. A.; SEKKAKI, A.; LARBI, K. Automating data warehouse design using ontology. In: . [S.l.: s.n.], 2016. 7, 57, 58

17 REN, S.; WANG, T.; LU, X. Dimensional modeling of medical data warehouse based on ontology. In: IEEE. *2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA)*. [S.l.], 2018. p. 144–149. 7, 59

18 PADAYACHEE, K.; WORKU, E. Shared situational awareness in information security incident management. In: *International Conference for Internet Technology and Secured Transactions*. Porto Alegre: ICITST-2017, 2017. p. 1. 2017. 18

19 CERT.BR. *Estatísticas do CERT.br*. 2017. Dezembro de 2019. Disponível em: <https://www.cert.br/stats/incidentes/>. 18, 74, 75, 76

20 CERT.BR. *Criando um Grupo de Respostas a Incidentes de Segurança em Computadores: Um Processo para Iniciar a Implantação*. 2004. Dezembro de 2019. Disponível em: <https://www.cert.br/certcc/csirts/Creating-A-CSIRT-br.html>. 18

21 LINE, M. B.; TØNDEL, I. A.; JAATUN, M. G. Information security incident management: Planning for failure. In: IEEE. *2014 Eighth International Conference on IT Security Incident Management & IT Forensics*. [S.l.], 2014. p. 47–61. 18

22 GLOBO, O. *Olimpíada registra 4,2 milhões de eventos de segurança cibernética*. 2016. Dezembro de 2019. Disponível em: <https://oglobo.globo.com/economia/olimpiada-registra-42-milhoes-de-eventos-de-seguranca-cibernetica-20061352>. 21

23 CERT.BR. *A segurança e defesa cibernética nos Jogos Olímpicos e Paralímpico 2016*. 2016. Dezembro de 2019. Disponível em: <https://www.cert.br/forum2016/slides/ForumCSIRTs2016-CDCiber-Coord-Rio2016.pdf>. 21

24 FAB. *FAB reforça atividades de defesa cibernética durante a Rio 2016*. 2016. Dezembro de 2019. Disponível em: <http://www.fab.mil.br/noticias/mostra/27517/>. 21

25 MD. *Defesa Cibernética no Brasil: Análise da Atuação do Ministério da Defesa na Copa do Mundo de 2014 e nas Olimpíadas de 2016*. 2019. Dezembro de 2019. Disponível em: <https://www.defesa.gov.br/arquivos/ensino_e_pesquisa/defesa_academia/cadn/artigos/xvi_cadn/defesa_cibernetica_no_brasil_analise_da_atuacao_do_ministerio_da_defesa_na_copa_do_mundo_de_2014_e_nas_olimpiadas_de_2016.pdf>. 21, 60

26 DEFESA, M. da. *Ministério da Defesa - Estratégia Nacional de Defesa*. 1. ed. [S.l.: s.n.], 2012. 21

27 ENGENHARIA, I. M. de. *Laboratório de Defesa Cibernética*. 2009. Março de 2020. Disponível em: <http://defesacibernetica.ime.eb.br/>. 21

28 GRUBER, T. R. Toward principles for the design of ontologies used for knowledge sharing. In: *International Journal of Human Computer Studies*. [S.l.: s.n.], 1995. p. 907 – 928. 25, 26

29 JASPER, R.; USCHOLD, M. A framework for understanding and classifying ontology applications. In: *X*. x: X, 1995. p. 907 – 928. Dezembro de 2018. Disponível em: <ttp://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.39.6456>. 25

30 FALBO, R. d. A.; GUIZZARDI, G.; DUARTE, K. C. An ontological approach to domain engineering. In: *International conference on Software engineering and knowledge engineering*. [S.l.]: ACM, 2002. p. 351–358. 25

31 SUAREZ-FIGUEROA, M. C.; GOMEZ-PEREZ, A.; MOTTA, E.; GANGEMI, A. Introduction: Ontology engineering in a networked world. In: *Ontology Engineering in a Networked World*. [S.l.]: Springer, 2012. p. 1–6. 25

32 GUIZZARDI, G.; WAGNER, G.; ALMEIDA, J. P. A.; GUIZZARDI, R. S. Towards ontological foundations for conceptual modeling: The unified foundational ontology (ufo) story. *Applied ontology*, IOS Press, v. 10, n. 3-4, p. 259–271, 2015. 26, 28

33 GUIZZARDI, G.; WAGNER, G. A unified foundational ontology and some applications of it in business modeling. In: *CAiSE Workshops (3)*. [S.l.: s.n.], 2004. p. 129 – 143. 27, 28

34 GUIZZARDI, G.; WAGNER, G.; FALBO, R. de A.; GUIZZARDI, R. S.; ALMEIDA, J. P. A. Towards ontological foundations for the conceptual modeling of events. In: SPRINGER. *International Conference on Conceptual Modeling*. [S.l.], 2013. p. 327–341. 28, 32, 108

35 GUIZZARDI, G.; WAGNER, G.; GUARINO, N.; SINDEREN, M. van. An ontologically well-founded profile for uml conceptual models. In: SPRINGER. *International Conference on Advanced Information Systems Engineering*. [S.l.], 2004. p. 112–126. 28

36 GUIZZARDI, R. S.; GUIZZARDI, G.; PERINI, A.; MYLOPOULOS, J. Towards an ontological account of agent-oriented goals. In: SPRINGER. *International Workshop on Software Engineering for Large-Scale Multi-agent Systems*. [S.l.], 2006. p. 148–164. 28

37 GUIZZARDI, G.; MASOLO, C.; BORGO, S. In defense of a trope-based ontology for conceptual modeling: an example with the foundations of attributes, weak entities and datatypes. In: SPRINGER. *International Conference on Conceptual Modeling*. [S.l.], 2006. p. 112–125. 28

38 ARMSTRONG, D. M. Universals: An opinionated introduction (boulder. *CO: Westview*, 1989. 28

39 JONATHAN, L. E. The possibility of metaphysics: substance, identity and time. New York: Oxford University Press, 1998. 28

40 GUIZZARDI, G.; FALBO, R. de A.; GUIZZARDI, R. S. Grounding software domain ontologies in the unified foundational ontology (ufo): The case of the ode software process ontology. In: *CIbSE*. [S.l.: s.n.], 2008. p. 127–140. 29

41 GUIZZARDI, G.; GUARINO, N.; ALMEIDA, J. P. A. Ontological considerations about the representation of events and endurants in business models. In: SPRINGER. *International Conference on Business Process Management*. [S.l.], 2016. p. 20–36. 30, 32

42 CARVALHO, V. A. *Foundations for Ontology-based Multi-level Conceptual Modeling*. Tese (Doutorado) — Universidade Federal do Espírito Santo, Brasil, 2016. 36

43 CARVALHO, V. A.; ALMEIDA, J. P. A.; FONSECA, C. M.; GUIZZARDI, G. Extending the foundations of ontology-based conceptual modeling with a multi-level theory. In: SPRINGER. *International Conference on Conceptual Modeling*. [S.l.], 2015. p. 119–133. 40

44 FERNÁNDEZ-LÓPEZ, M.; GÓMEZ-PÉREZ, A.; JURISTO, N. Methontology: from ontological art towards ontological engineering. American Asociation for Artificial Intelligence, 1997. 42

45 FETTKE, P.; LOOS, P. Ontological evaluation of reference models using the bunge-wand-weber model. *AMCIS 2003 Proceedings*, p. 384, 2003. 44

46 KIMBALL, R.; ROSS, M. *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. 3. ed. [S.l.: s.n.], 2013. 46, 49

47 MOREIRA, J.; CORDEIRO, K.; CAMPOS, M. L.; BORGES, M. Ontowarehousing–multidimensional design supported by a foundational ontology: a temporal perspective. In: SPRINGER. *International Conference on Data Warehousing and Knowledge Discovery*. [S.l.], 2014. p. 35–44. 49, 104, 106

48 AMARAL, G.; GUIZZARDI, G. On the application of ontological patterns for conceptual modeling in multidimensional models. In: *European Conference on Advances in Databases and Information Systems*. [S.l.: s.n.], 2019. p. 215–231. 49, 50, 104, 105, 112

49 ROMERO, O.; ABELLÓ, A. A framework for multidimensional design of data warehouses from ontologies. *Data & Knowledge Engineering*, Elsevier, v. 69, n. 11, p. 1138–1157, 2010. 55

50 THENMOZHI, M.; VIVEKANANDAN, K. A tool for data warehouse multidimensional schema design using ontology. *International Journal of Computer Science Issues (IJCSI)*, International Journal of Computer Science Issues (IJCSI), v. 10, n. 2, p. 161, 2013. 56

51 SILVA, S. A. D. *Modelo de Capacidades e Maturidade para Defesa Cibernética*. São Paulo: [s.n.], 2011. (Dissertação de Mestrado). 60

52 ISO/IEC, A. N. *Norma Brasileira de Tecnologia de Informação Técnicas de Segurança - Técnicas de segurança - Sistemas de gestão de segurança da informação - Requisitos (ABNT NBR ISO IEC 27001:2013)*. [S.l.]: ABNT, 2013. v. 1. 73, 74, 76

53 MD. *Ministério da Defesa - Doutrina Militar de Defesa Cibernética*. 1. ed. [S.l.: s.n.], 2014. 73, 75, 76

54 ISO/IEC, A. N. *Norma Brasileira de Tecnologia de Informação Técnicas de Segurança Código de prática para a gestão da segurança de informação (ABNT NBR ISO IEC 27002:2005)*. [S.l.]: ABNT, 2015. v. 1. 73

55 CORPORATION, T. M. *Common Attack Pattern Enumeration and Classification*. 1999. Dezembro de 2019. Disponível em: <https://cwe.mitre.org/index.html>. 73, 76

56 ISO/IEC. *Information technology — Security techniques — Information security management systems — Overview and vocabulary (ISO/IEC 27000:2018-02)*. [S.l.]: ISO copyright office, 2018. v. 2. 74, 75

57 HOWARD, J. D.; LONGSTAFF, T. A. A common language for computer security incidents. *Sandia National Laboratories*, v. 1, p. 1–19, 1998. 74, 75, 76, 82, 84, 85

58 LABORATORIES, S. N.; ENERGY, U. S. D. of; SCIENTIFIC, U. S. D. of Energy. Office of; INFORMATION, T. *A Common Language for Computer Security Incidents*. United States. Department of Energy, 1998. Disponível em: <https://books.google.com.br/books?id=Br-3nQAACAAJ>. 74

59 CAPEC. *ECommon Attack Pattern Enumeration and Classification*. 2007. Dezembro de 2019. Disponível em: <http://capec.mitre.org/index.html>. 75

60 DUARTE, B. B.; FALBO, R. A.; GUIZZARDI, G.; GUIZZARDI, R. S.; SOUZA, V. E. Towards an ontology of software defects, errors and failures. In: SPRINGER. *International Conference on Conceptual Modeling*. [S.l.], 2018. p. 349–362. 80, 139

61 KRISHNA, C. R.; DUTTA, M.; KUMAR, R. *Proceedings of 2nd International Conference on Communication, Computing and Networking: ICCCN 2018, NITTTR Chandigarh, India*. [S.l.]: Springer, 2018. v. 46. 92

62 TRIPATHI, S.; GUPTA, B.; ALMOMANI, A.; MISHRA, A.; VELURU, S. Hadoop based defense solution to handle distributed denial of service (ddos) attacks. *Journal of Information Security*, Scientific Research Publishing, v. 4, n. 03, p. 150, 2013. 92

63 GARCIA, L. *Um pouco sobre o WannaCry*. 2019. 19 Março de 2019. Disponível em: <https://wltech.com.br/category/noticias/page/2/>. 97

64 ROHR, A. *Hackers de grupo ligado à Coreia do Norte atacaram bancos na América Latina, diz Trend Micro*. 2018. Dezembro de 2019. Disponível em: <https://g1.globo.com/economia/tecnologia/blog/altieres-rohr/post/2018/11/26/hackers-de-grupo-ligado-a-coreia-do-norte-atacaram-bancos-na-america-latina-diz-trend-micro.ghtml>. 97

65 MICROSOFT. *Worm de ransomware WannaCrypt direcionado a sistemas desatualizados*. 2017. Dezembro de 2019. Disponível em: <https://docs.microsoft.com/pt-br/windows/security/threat-protection/wannacrypt-ransomware-worm-targets-out-of-date-systems-wdsi>. 97

66 PATURI, A. *The Financial Impact of Cyber Threats*. 2017. Dezembro de 2019. Disponível em: <https://www.darkreading.com/risk/the-financial-impact-of-cyber-threats/a/d-id/1330668>. 97

67 MOREIRA, G.; CALEGARIO, V.; DUARTE, J.; SANTOS, A. dos. A era dos crypto ransomwares: um estudo de caso sobre o wannacry. In: *Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*. [S.l.]: Sociedade Brasileira de Computação, 2017. p. 509–516. 98

68 GROUP, O. M. et al. Common warehouse metamodel (cwm) specification. *An Adopted Specification of the Object Management Group, Inc.*, 2000. 111, 112

69 WIKIPéDIA. *Método Jaro Distance*. 2007. Março de 2020. Disponível em: <https://pt.wikipedia.org/wiki/Dist%C3%A2ncia_de_Jaro-Winkler>. 121

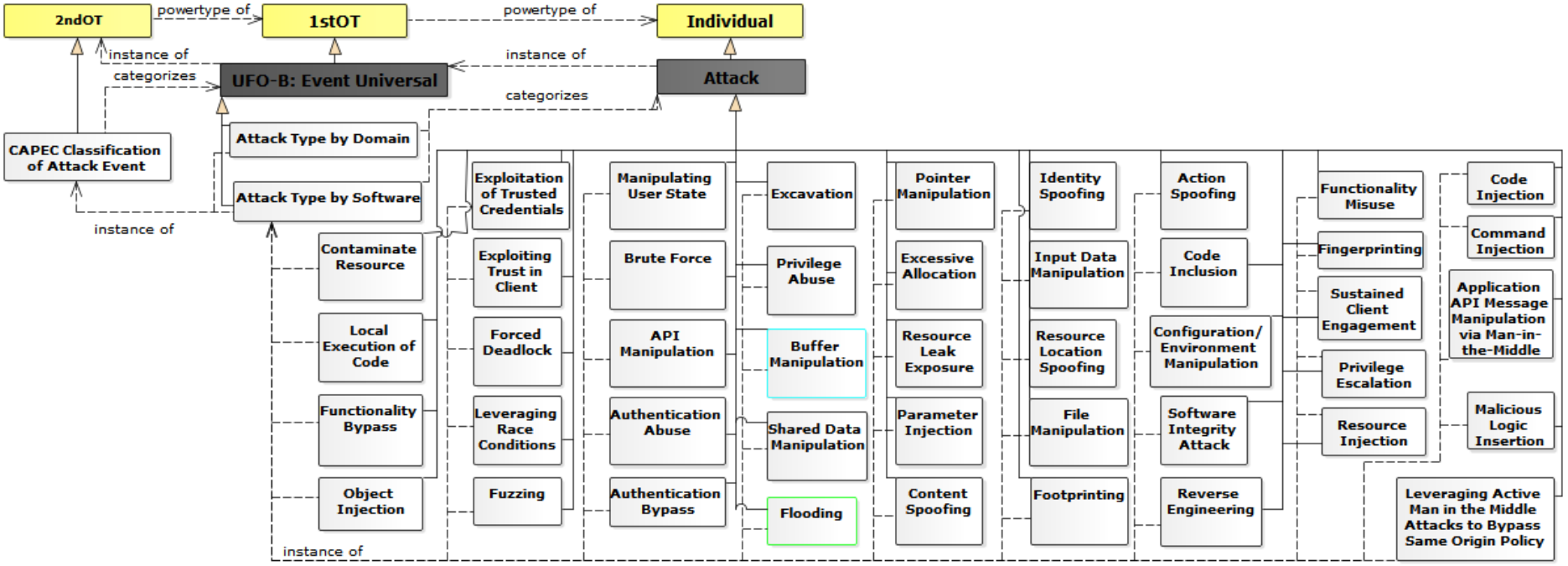
70 SALES, T. P.; BAIÃO, F.; GUIZZARDI, G.; ALMEIDA, J. P. A.; GUARINO, N.; MYLOPOULOS, J. The common ontology of value and risk. In: SPRINGER. *International Conference on Conceptual Modeling*. [S.l.], 2018. p. 121–135. 139

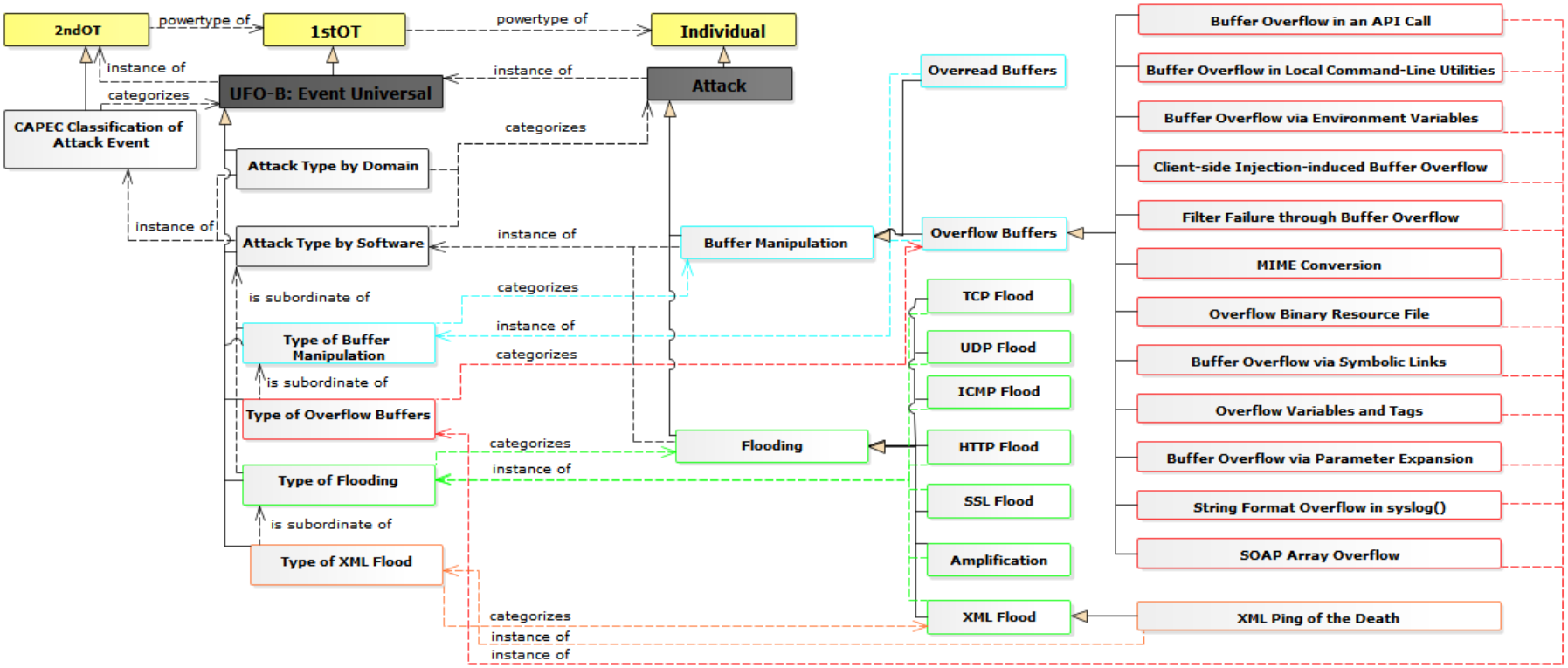
# APÊNDICE A – ASSOCIAÇÃO DO PADRÃO DE ATAQUE DO CAPEC AO TIPO DE ATAQUE DA BASE DE INCIDENTE ATRAVÉS DO CVE

| Nome do tipo de ataque na base de incidente do CSIRT | CVE | CWE associado | CAPEC | Regra |
|---|---|---|---|---|
| HTTP: Apache Tomcat PUT JSP File Upload (CVE-2017-12615 and CVE-2017-12617) | CVE-2017-12615 CVE-2017-12617 | Unrestricted Upload of File with Dangerous Type | Accessing Functionality Not Properly Constrained by ACLs | 1 |
| HTTP: JBoss Application Server Remote Code Execution Vulnerability (CVE-2017-12149) | CVE-2017-12149 | Deserialization of Untrusted Data | Object Injection | 1 |
| HTTP: Internet Explorer XSS Filter Vulnerability (CVE-2016-3212) | CVE-2016-3212 | Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') | Cross-Site Scripting (XSS) | 2 |
| HTTP: Joomla SQL injection vulnerability (CVE-2017-8917) | CVE-2017-8917 | Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') | SQL Injection | 2 |
| HTTP: WP GDPR Compliance Plugin Privilege Escalation Vulnerability (CVE-2018-19207) | CVE-2018-19207 | Direct Request ('Forced Browsing') | Forceful Browsing | 2 |
| HTTP: GNU Bash Environment Variable Handling Command Execution Exploit (CVE-2014-6271) | CVE-2014-6271 | used in an OS Command ('OS Command Injection') | OS Command Injection | 2 |
| HTTP: Acrobat Reader Memory Corruption Vulnerability(CVE-2016-6955) | CVE-2016-6955 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Adobe Acrobat Reader Memory Corruption Vulnerability (CVE-2018-12847) | CVE-2018-12847 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Adobe Acrobat Reader Memory Corruption Vulnerability (CVE-2018-12855) | CVE-2018-12855 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Adobe Graphics Manager Memory Corruption Vulnerability (CVE-2017-11252) | CVE-2017-11252 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: IIS 6.0 WebDAV Service ScStoragePathFromUrl Function Buffer Overflow (CVE-2017-7269) | CVE-2017-7269 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Internet Explorer Memory Corruption Vulnerability (CVE-2014-6366) | CVE-2014-6366 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Internet Explorer Memory Corruption Vulnerability (CVE-2017-8594) | CVE-2017-8594 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft ATMFD Kernel Pool Code Execution Vulnerability (CVE-2015-2426) | CVE-2015-2426 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Edge DoLoopBodyStart Out-of-Bound Vulnerability (CVE-2017-11811) | CVE-2017-11811 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft IE type confusion vulnerability (CVE-2017-0202) | CVE-2017-0202 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Internet Explorer CDomRange Use After Free Vulnerability (CVE-2014-0274) | CVE-2014-0274 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Internet Explorer DirectWrite use after free vulnerability (CVE-2014-0263) | CVE-2014-0263 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Internet Explorer Onpropertychange Use After Free Vulnerability (CVE-2014-0312) | CVE-2014-0312 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Internet Explorer Use-After-Free Vulnerability (CVE-2017-11903) | CVE-2017-11903 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Office EQNEDT32 Stack Buffer Overflow (CVE-2018-0802) | CVE-2018-0802 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Office Memory Corruption Vulnerability (CVE-2015-2477) | CVE-2015-2477 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Office Memory Corruption Vulnerability (CVE-2017-11826) | CVE-2017-11826 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Office Memory Corruption Vulnerability CVE-2016-0139 | CVE-2016-0139 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft PowerPoint 365 Use-After-Free Vulnerability (CVE-2019-0822) | CVE-2019-0822 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| PKTSEARCH: HP OpenView Storage Data Protector Denial of Service Vulnerability (CVE-2011-1866) | CVE-2011-1866 | Improper Restriction of Operations within the Bounds of a Memory Buffer | Buffer Manipulation | 3 |
| HTTP: Microsoft Win32k Elevation of Privilege Vulnerability (CVE-2018-8124) | CVE-2018-8124 | Improper Resource Shutdown or Release | Excessive Allocation | 4 |
| HTTP: Microsoft Windows Kernel Elevation of Privilege Vulnerability (CVE-2018-8611) | CVE-2018-8611 | Improper Resource Shutdown or Release | Excessive Allocation | 4 |
| HTTP: Ruby on Rails Web Application Framework DoS Vulnerability (CVE-2019-5419) | CVE-2019-5419 | Uncontrolled Resource Consumption | Excessive Allocation | 4 |
| HTTP: Kibana Local File Inclusion Vulnerability (CVE-2018-17246) | CVE-2018-17246 | Inclusion of Functionality from Untrusted Control Sphere | Code Inclusion | 4 |
| HTTP: Microsoft Office Bad Index Remote Code Execution Vulnerability (CVE-2014-6334) | CVE-2014-6334 | Improper Control of Generation of Code ('Code Injection') | Code Injection | 4 |

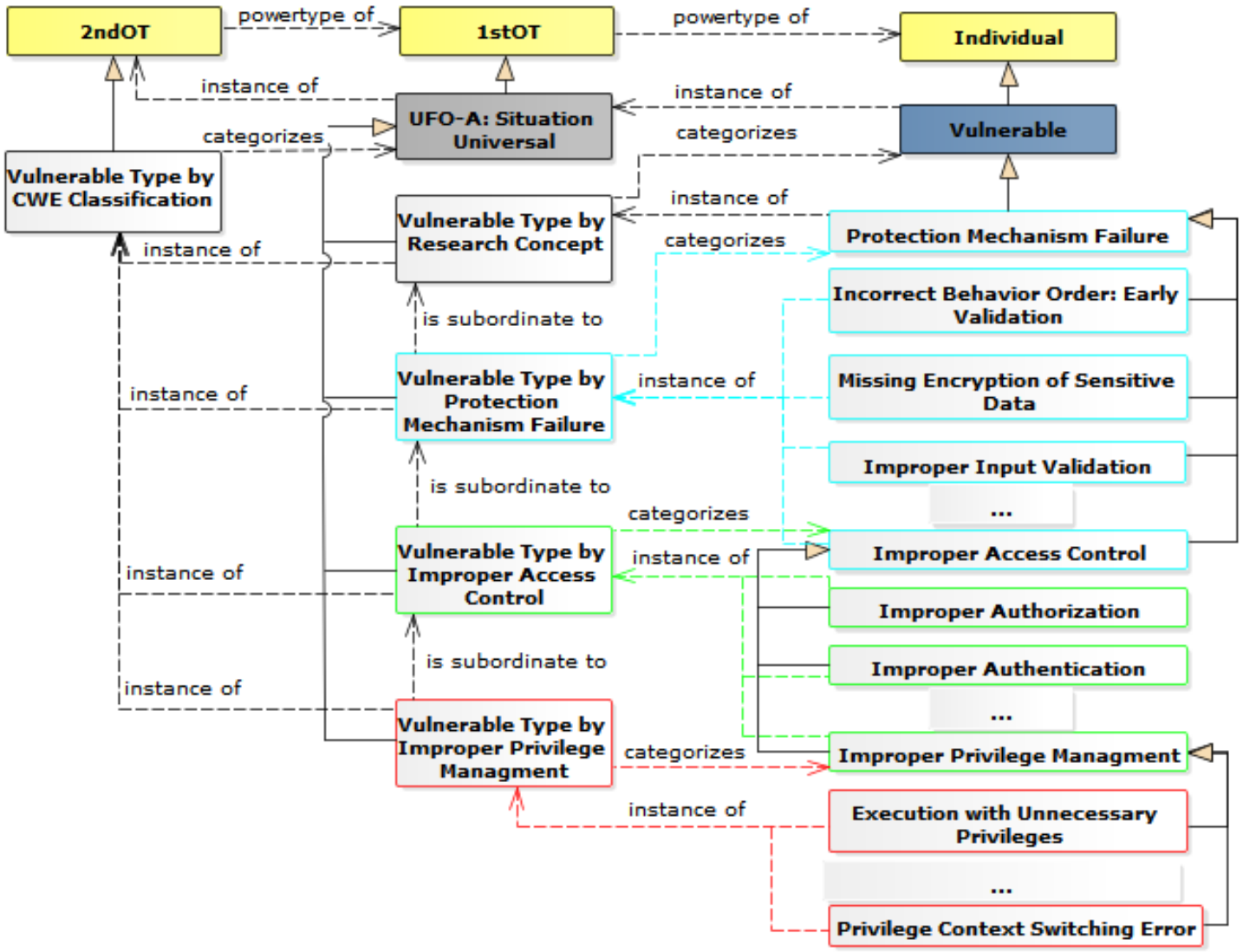
| Nome do tipo de ataque na base de incidente do CSIRT | CVE | CWE associado | CAPEC | Regra |
|---|---|---|---|---|
| HTTP: Microsoft Windows HTTP.sys Remote Code Execution (CVE-2015-1635) | CVE-2015-1635 | Improper Control of Generation of Code ('Code Injection') | Code Injection | 4 |
| HTTP: Apache HTTPD Cookie Handling Denial Of Service (CVE-2012-0021) | CVE-2012-0021 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: Apache Server Multiple Vulnerabilities CVE-2014-0098 | CVE-2014-0098 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: Apache Struts 2 Remote Code Execution (CVE-2017-5638) | CVE-2017-5638 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: Apache Struts Remote Code Execution Vulnerability (CVE-2018-11776) | CVE-2018-11776 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: Drupal Remote Code Execution (CVE-2018-7600) | CVE-2018-7600 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: PHP Remote Code Execution Vulnerability (CVE-2018-20062) | CVE-2018-20062 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: Spring Data Commons Remote Code Execution Vulnerability (CVE-2018-1273) | CVE-2018-1273 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| TELNET: Cisco Catalyst Remote Code Execution Vulnerability (CVE-2017-3881) | CVE-2017-3881 | Improper Input Validation | Leverage Alternate Encoding | 4 |
| HTTP: Microsoft Browser Information Disclosure Vulnerability (CVE-2017-0068) | CVE-2017-0068 | Information Exposure | Host Discovery | 4 |
| HTTP: Microsoft Windows GDI Component Information Disclosure Vulnerability (CVE-2017-0287) | CVE-2017-0287 | Information Exposure | Host Discovery | 4 |
| HTTP: Microsoft Windows GDI Component Information Disclosure Vulnerability (CVE-2017-0289) | CVE-2017-0289 | Information Exposure | Host Discovery | 4 |
| HTTP: Microsoft Browser Spoofing Vulnerability (CVE-2016-3276) | CVE-2016-3276 | Improper Access Control | Targeted Malware | 4 |
| HTTP: Microsoft Edge PDF Parsing Out of Bounds Write Vulnerability (CVE-2016-3319) | CVE-2016-3319 | Improper Access Control | Targeted Malware | 4 |
| HTTP: Windows Image File Handling Information Disclosure Vulnerability (CVE-2016-7212) | CVE-2016-7212 | Improper Access Control | Targeted Malware | 4 |
| HTTP: Dasan GPON Home Routers Authentication Bypass (CVE-2018-10561) | CVE-2018-10561 | Improper Authentication | Authentication Bypass | 5 |
| SMTP: Dovecot rfc822_parse_domain Out of Bounds Read Vulnerability (CVE-2017-14461) | CVE-2017-14461 | Out-of-bounds Read Information Exposure | Overread Buffers | 5 |

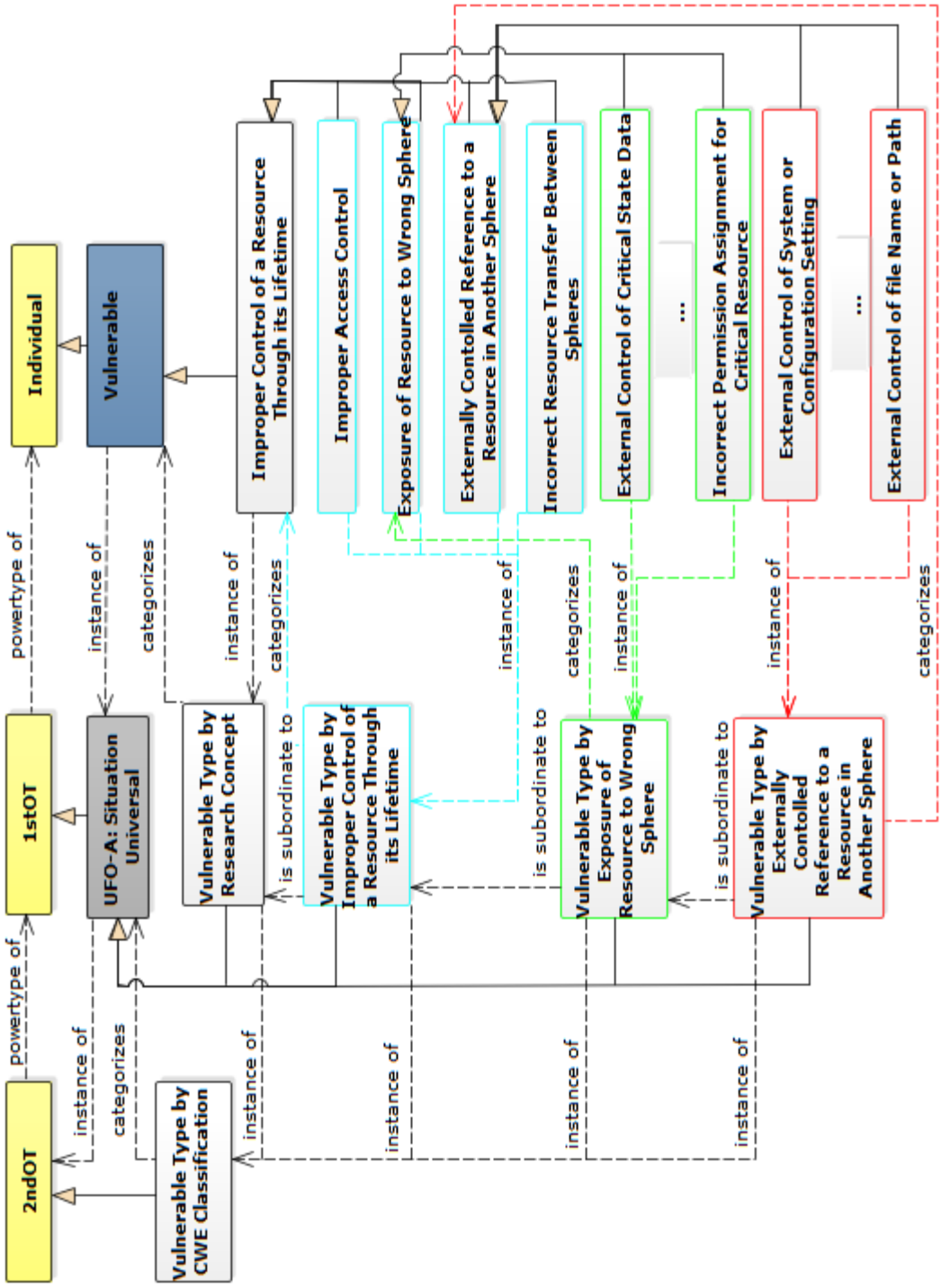
# APÊNDICE B – MODELO BASEADO NA UFO-MLT DE TIPO DE ATAQUE

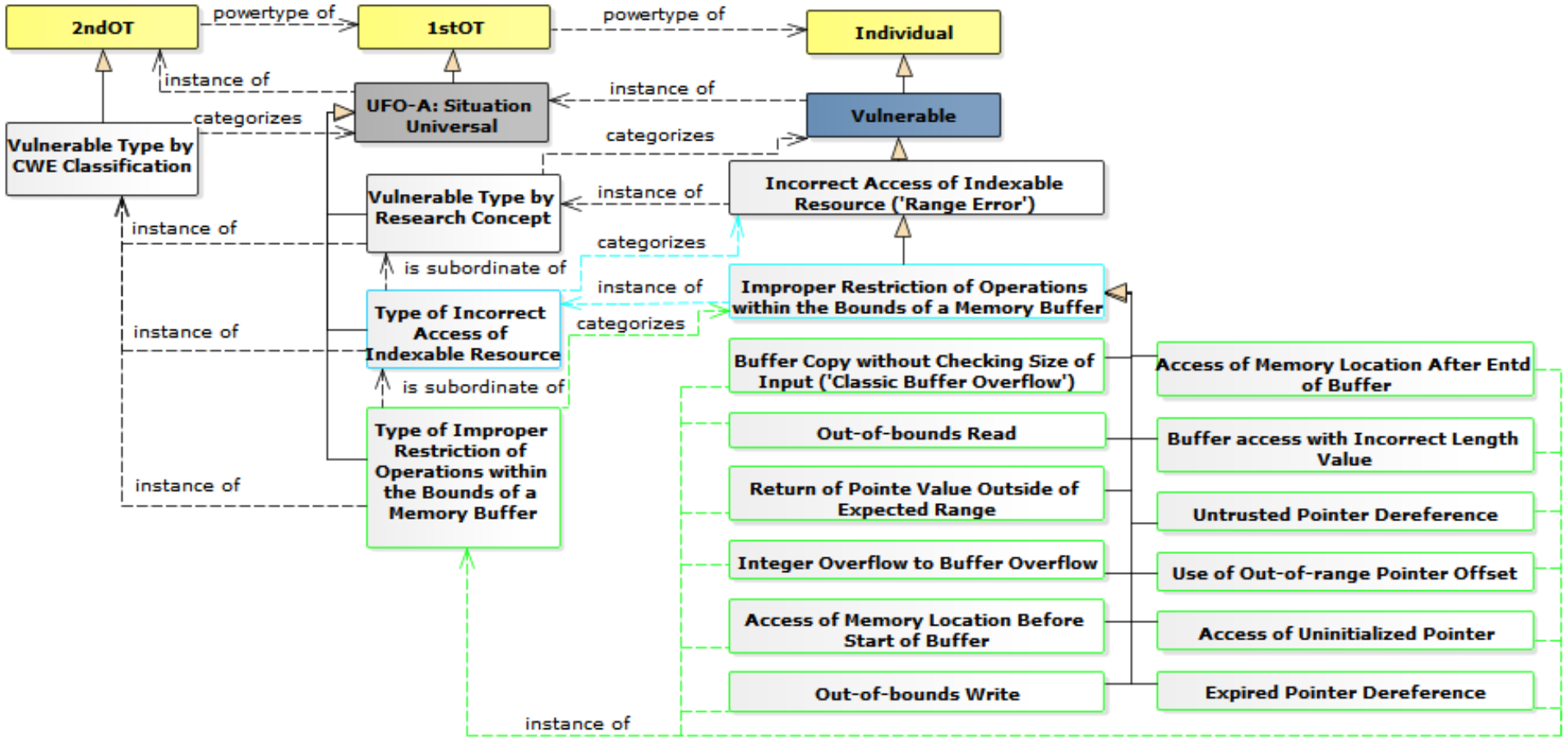



2ndOT
powertype of
1stOT
powertype of
Individual
instance of
instance of
categorizes
UFO-B: Event Universal
Attack
CAPEC Classification of Attack Event
categorizes
Attack Type by Domain
instance of
Attack Type by Software
instance of
Buffer Manipulation
is subordinate of
categorizes
Type of Buffer Manipulation
instance of
categorizes
is subordinate of
Type of Overflow Buffers
categorizes
Flooding
Type of Flooding
instance of
is subordinate of
Type of XML Flood
categorizes
instance of
instance of
Overread Buffers
Overflow Buffers
TCP Flood
UDP Flood
ICMP Flood
HTTP Flood
SSL Flood
Amplification
XML Flood
Buffer Overflow in an API Call
Buffer Overflow in Local Command-Line Utilities
Buffer Overflow via Environment Variables
Client-side Injection-induced Buffer Overflow
Filter Failure through Buffer Overflow
MIME Conversion
Overflow Binary Resource File
Buffer Overflow via Symbolic Links
Overflow Variables and Tags
Buffer Overflow via Parameter Expansion
String Format Overflow in syslog()
SOAP Array Overflow
XML Ping of the Death

# APÊNDICE C – MODELO BASEADO NA UFO-MLT DE TIPO DE SITUAÇÃO VULNERÁVEL




Individual
Vulnerable
Improper Control of a Resource Through its Lifetime
Improper Access Control
Exposure of Resource to Wrong Sphere
Externally Contolled Reference to a Resource in Another Sphere
Incorrect Resource Transfer Between Spheres
External Control of Critical State Data
...
Incorrect Permission Assignment for Critical Resource
External Control of System or Configuration Setting
...
External Control of file Name or Path
powertype of
instance of
categorizes
1stOT
UFO-A: Situation Universal
Vulnerable Type by Research Concept
Vulnerable Type by Improper Control of a Resource Through its Lifetime
Vulnerable Type by Exposure of Resource to Wrong Sphere
Vulnerable Type by Externally Contolled Reference to a Resource in Another Sphere
is subordinate to
2ndOT
Vulnerable Type by CWE Classification


2ndOT
powertype of
1stOT
powertype of
Individual
instance of
UFO-A: Situation Universal
instance of
Vulnerable
categorizes
Vulnerable Type by CWE Classification
categorizes
Vulnerable Type by Research Concept
instance of
Incorrect Access of Indexable Resource ('Range Error')
instance of
categorizes
is subordinate of
instance of
Improper Restriction of Operations within the Bounds of a Memory Buffer
Type of Incorrect Access of Indexable Resource
categorizes
instance of
Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
Access of Memory Location After Entd of Buffer
is subordinate of
Out-of-bounds Read
Buffer access with Incorrect Length Value
Type of Improper Restriction of Operations within the Bounds of a Memory Buffer
Return of Pointe Value Outside of Expected Range
Untrusted Pointer Dereference
instance of
Integer Overflow to Buffer Overflow
Use of Out-of-range Pointer Offset
Access of Memory Location Before Start of Buffer
Access of Uninitialized Pointer
Out-of-bounds Write
Expired Pointer Dereference
instance of